



WINDOWS 10 AND WINDOWS SERVER 2016 SECURITY AUDITING AND MONITORING REFERENCE

June 16, 2016

Abstract

This document contains:

- Detailed technical descriptions for most of the advanced security audit policies that are included with Windows 10 and Windows Server 2016.
- Monitoring recommendations for security events to include in advanced security audit policies.
- Recommendations for Group Policy settings for advanced security audit policy for domain controllers, workstations, and member servers.

Andrei Miroshnikov
Microsoft ISRM ACE Team

Copyright

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2016 Microsoft. All rights reserved.

Please refer to [Microsoft Trademarks](#) for a list of trademarked products.

Portions of this software may be based on NCSA Mosaic. NCSA Mosaic was developed by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign. Distributed under a licensing agreement with Spyglass, Inc.

May contain security software licensed from RSA Data Security, Inc.

UPnP™ is a certification mark of the UPnP™ Implementers Corporation.

Bluetooth® is a trademark owned by Bluetooth SIG, Inc., USA and licensed to Microsoft Corporation.

Intel is a registered trademark of Intel Corporation.

Itanium is a registered trademark of Intel Corporation.

All other trademarks are property of their respective owners.

Table of publishing updates

Version	Date	Type of publishing
1	05/13/2016	First publication
1.1	06/01/2016	“Schema Value” column added to File Access Codes table
1.2	06/16/2016	Updates to “Audit Authorization Policy Change” and “4703(S): A user right was adjusted.”

Contents

Account Logon.....	1
Audit Credential Validation.....	1
4774(S): An account was mapped for logon.....	2
4775(F): An account could not be mapped for logon.....	2
4776(S, F): The computer attempted to validate the credentials for an account.....	3
4777(F): The domain controller failed to validate the credentials for an account.....	6
Audit Kerberos Authentication Service.....	6
4768(S, F): A Kerberos authentication ticket (TGT) was requested.....	7
4771(F): Kerberos pre-authentication failed.....	20
4772(F): A Kerberos authentication ticket request failed.....	25
Audit Kerberos Service Ticket Operations.....	26
4769(S, F): A Kerberos service ticket was requested.....	27
4770(S): A Kerberos service ticket was renewed.....	38
4773(F): A Kerberos service ticket request failed.....	41
Audit Other Account Logon Events.....	42
Account Management.....	43
Audit Application Group Management.....	43
4783(S): A basic application group was created.....	43
4784(S): A basic application group was changed.....	43
4785(S): A member was added to a basic application group.....	43
4786(S): A member was removed from a basic application group.....	43
4787(S): A non-member was added to a basic application group.....	43
4788(S): A non-member was removed from a basic application group.....	43
4789(S): A basic application group was deleted.....	43

4790(S): An LDAP query group was created.....	43
4791(S): An LDAP query group was changed.....	43
4792(S): An LDAP query group was deleted.....	43
Audit Computer Account Management.....	44
4741(S): A computer account was created.....	45
4742(S): A computer account was changed.....	57
4743(S): A computer account was deleted.....	65
Audit Distribution Group Management.....	67
4749(S): A security-disabled global group was created.....	68
4750(S): A security-disabled global group was changed.....	70
4751(S): A member was added to a security-disabled global group.....	73
4752(S): A member was removed from a security-disabled global group.....	77
4753(S): A security-disabled global group was deleted.....	80
4759(S): A security-disabled universal group was created.....	82
4760(S): A security-disabled universal group was changed.....	82
4761(S): A member was added to a security-disabled universal group.....	82
4762(S): A member was removed from a security-disabled universal group.....	82
4763(S): A security-disabled universal group was deleted.....	82
4744(S): A security-disabled local group was created.....	83
4745(S): A security-disabled local group was changed.....	83
4746(S): A member was added to a security-disabled local group.....	83
4747(S): A member was removed from a security-disabled local group.....	83
4748(S): A security-disabled local group was deleted.....	83
Audit Other Account Management Events.....	84
4782(S): The password hash an account was accessed.....	85
4793(S): The Password Policy Checking API was called.....	87
Audit Security Group Management.....	89

4727(S): A security-enabled global group was created.....	90
4737(S): A security-enabled global group was changed.....	90
4728(S): A member was added to a security-enabled global group.....	90
4729(S): A member was removed from a security-enabled global group.....	90
4730(S): A security-enabled global group was deleted.....	90
4731(S): A security-enabled local group was created.....	91
4732(S): A member was added to a security-enabled local group.....	93
4733(S): A member was removed from a security-enabled local group.....	97
4734(S): A security-enabled local group was deleted.....	101
4735(S): A security-enabled local group was changed.....	103
4754(S): A security-enabled universal group was created.....	106
4755(S): A security-enabled universal group was changed.....	106
4756(S): A member was added to a security-enabled universal group.....	106
4757(S): A member was removed from a security-enabled universal group.....	106
4758(S): A security-enabled universal group was deleted.....	106
4764(S): A group's type was changed.....	107
4799(S): A security-enabled local group membership was enumerated.....	109
Audit User Account Management.....	113
4720(S): A user account was created.....	115
4722(S): A user account was enabled.....	123
4723(S, F): An attempt was made to change an account's password.....	125
4724(S, F): An attempt was made to reset an account's password.....	127
4725(S): A user account was disabled.....	130
4726(S): A user account was deleted.....	132
4738(S): A user account was changed.....	135
4740(S): A user account was locked out.....	142
4765(S): SID History was added to an account.....	144

4766(F): An attempt to add SID History to an account failed.....	145
4767(S): A user account was unlocked.....	146
4780(S): The ACL was set on accounts which are members of administrators groups.....	147
4781(S): The name of an account was changed.....	149
4794(S, F): An attempt was made to set the Directory Services Restore Mode administrator password.....	151
4798(S): A user's local group membership was enumerated.....	153
5376(S): Credential Manager credentials were backed up.....	156
5377(S): Credential Manager credentials were restored from a backup.....	158
Detailed Tracking.....	160
Audit DPAPI Activity.....	160
4692(S, F): Backup of data protection master key was attempted.....	161
4693(S, F): Recovery of data protection master key was attempted.....	164
4694(S, F): Protection of auditable protected data was attempted.....	166
4695(S, F): Unprotection of auditable protected data was attempted.....	167
Audit PNP Activity.....	168
6416(S): A new external device was recognized by the System.....	169
6419(S): A request was made to disable a device.....	174
6420(S): A device was disabled.....	179
6421(S): A request was made to enable a device.....	184
6422(S): A device was enabled.....	189
6423(S): The installation of this device is forbidden by system policy.....	194
6424(S): The installation of this device was allowed, after having previously been forbidden by policy.....	198
Audit Process Creation.....	199
4688(S): A new process has been created.....	200
4696(S): A primary token was assigned to process.....	206
Audit Process Termination.....	211
4689(S): A process has exited.....	212

Audit RPC Events.....	215
5712(S): A Remote Procedure Call (RPC) was attempted.....	215
DS Access.....	217
Audit Detailed Directory Service Replication.....	217
4928(S, F): An Active Directory replica source naming context was established.....	218
4929(S, F): An Active Directory replica source naming context was removed.....	220
4930(S, F): An Active Directory replica source naming context was modified.....	222
4931(S, F): An Active Directory replica destination naming context was modified.....	224
4934(S): Attributes of an Active Directory object were replicated.....	225
4935(F): Replication failure begins.....	226
4936(S): Replication failure ends.....	227
4937(S): A lingering object was removed from a replica.....	227
Audit Directory Service Access.....	229
4662(S, F): An operation was performed on an object.....	230
4661(S, F): A handle to an object was requested.....	235
Audit Directory Service Changes.....	243
5136(S): A directory service object was modified.....	244
5137(S): A directory service object was created.....	248
5138(S): A directory service object was undeleted.....	252
5139(S): A directory service object was moved.....	255
5141(S): A directory service object was deleted.....	259
Audit Directory Service Replication.....	263
4932(S): Synchronization of a replica of an Active Directory naming context has begun.....	263
4933(S, F): Synchronization of a replica of an Active Directory naming context has ended.....	265
Logon and Logoff.....	268
Audit Account Lockout.....	268
4625(F): An account failed to log on.....	269

Audit User/Device Claims.....	276
4626(S): User/Device claims information.....	277
Audit Group Membership.....	281
4627(S): Group membership information.....	282
Audit IPsec Extended Mode.....	285
4978: During Extended Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.....	285
4979: IPsec Main Mode and Extended Mode security associations were established.....	285
4980: IPsec Main Mode and Extended Mode security associations were established.....	285
4981: IPsec Main Mode and Extended Mode security associations were established.....	285
4982: IPsec Main Mode and Extended Mode security associations were established.....	285
4983: An IPsec Extended Mode negotiation failed. The corresponding Main Mode security association has been deleted.....	285
4984: An IPsec Extended Mode negotiation failed. The corresponding Main Mode security association has been deleted.....	285
Audit IPsec Main Mode.....	286
4646: Security ID: %1.....	286
4650: An IPsec Main Mode security association was established. Extended Mode was not enabled. Certificate authentication was not used.....	286
4651: An IPsec Main Mode security association was established. Extended Mode was not enabled. A certificate was used for authentication.....	286
4652: An IPsec Main Mode negotiation failed.....	286
4653: An IPsec Main Mode negotiation failed.....	286
4655: An IPsec Main Mode security association ended.....	286
4976: During Main Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.....	286
5049: An IPsec Security Association was deleted.....	286
5453: An IPsec negotiation with a remote computer failed because the IKE and AuthIP IPsec Keying Modules (IKEEXT) service is not started.....	286
Audit IPsec Quick Mode.....	287
4977: During Quick Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.....	287
5451: An IPsec Quick Mode security association was established.....	287

5452: An IPsec Quick Mode security association ended.....	287
Audit Logoff.....	288
4634(S): An account was logged off.....	289
4647(S): User initiated logoff.....	291
Audit Logon.....	293
4624(S): An account was successfully logged on.....	294
4625(F): An account failed to log on.....	301
4648(S): A logon was attempted using explicit credentials.....	301
4675(S): SIDs were filtered.....	305
Audit Network Policy Server.....	307
6272: Network Policy Server granted access to a user.....	307
6273: Network Policy Server denied access to a user.....	307
6274: Network Policy Server discarded the request for a user.....	307
6275: Network Policy Server discarded the accounting request for a user.....	307
6276: Network Policy Server quarantined a user.....	307
6277: Network Policy Server granted access to a user but put it on probation because the host did not meet the defined health policy.....	307
6278: Network Policy Server granted full access to a user because the host met the defined health policy.....	307
6279: Network Policy Server locked the user account due to repeated failed authentication attempts.....	307
6280: Network Policy Server unlocked the user account.....	307
Audit Other Logon/Logoff Events.....	308
4649(S): A replay attack was detected.....	309
4778(S): A session was reconnected to a Window Station.....	310
4779(S): A session was disconnected from a Window Station.....	313
4800(S): The workstation was locked.....	316
4801(S): The workstation was unlocked.....	318
4802(S): The screen saver was invoked.....	320
4803(S): The screen saver was dismissed.....	322

5378(F): The requested credentials delegation was disallowed by policy.....	324
5632(S, F): A request was made to authenticate to a wireless network.....	326
5633(S, F): A request was made to authenticate to a wired network.....	329
Audit Special Logon.....	332
4964(S): Special groups have been assigned to a new logon.....	333
4672(S): Special privileges assigned to new logon.....	336
Object Access.....	340
Audit Application Generated.....	340
4665: An attempt was made to create an application client context.....	340
4666: An application attempted an operation.....	340
4667: An application client context was deleted.....	340
4668: An application was initialized.....	340
Audit Certification Services.....	341
4868: The certificate manager denied a pending certificate request.....	342
4869: Certificate Services received a resubmitted certificate request.....	342
4870: Certificate Services revoked a certificate.....	342
4871: Certificate Services received a request to publish the certificate revocation list (CRL).....	342
4872: Certificate Services published the certificate revocation list (CRL).....	342
4873: A certificate request extension changed.....	342
4874: One or more certificate request attributes changed.....	342
4875: Certificate Services received a request to shut down.....	342
4876: Certificate Services backup started.....	342
4877: Certificate Services backup completed.....	342
4878: Certificate Services restore started.....	342
4879: Certificate Services restore completed.....	342
4880: Certificate Services started.....	342
4881: Certificate Services stopped.....	342

4882: The security permissions for Certificate Services changed.....	342
4883: Certificate Services retrieved an archived key.....	342
4884: Certificate Services imported a certificate into its database.....	342
4885: The audit filter for Certificate Services changed.....	342
4886: Certificate Services received a certificate request.....	342
4887: Certificate Services approved a certificate request and issued a certificate.....	342
4888: Certificate Services denied a certificate request.....	342
4889: Certificate Services set the status of a certificate request to pending.....	343
4890: The certificate manager settings for Certificate Services changed.....	343
4891: A configuration entry changed in Certificate Services.....	343
4892: A property of Certificate Services changed.....	343
4893: Certificate Services archived a key.....	343
4894: Certificate Services imported and archived a key.....	343
4895: Certificate Services published the CA certificate to Active Directory Domain Services.....	343
4896: One or more rows have been deleted from the certificate database.....	343
4897: Role separation enabled.....	343
4898: Certificate Services loaded a template.....	343
Audit Detailed File Share.....	344
5145(S, F): A network share object was checked to see whether client can be granted desired access.....	345
Audit File Share.....	352
5140(S, F): A network share object was accessed.....	353
5142(S): A network share object was added.....	356
5143(S): A network share object was modified.....	358
5144(S): A network share object was deleted.....	364
5168(F): SPN check for SMB/SMB2 failed.....	366
Audit File System.....	369
4656(S, F): A handle to an object was requested.....	370

4658(S): The handle to an object was closed.....	380
4660(S): An object was deleted.....	383
4663(S): An attempt was made to access an object.....	386
4664(S): An attempt was made to create a hard link.....	392
4985(S): The state of a transaction has changed.....	394
5051(-): A file was virtualized.....	396
4670(S): Permissions on an object were changed.....	398
Audit Filtering Platform Connection.....	404
5031(F): The Windows Firewall Service blocked an application from accepting incoming connections on the network.....	405
5150(-): The Windows Filtering Platform blocked a packet.....	406
5151(-): A more restrictive Windows Filtering Platform filter has blocked a packet.....	407
5154(S): The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.....	408
5155(F): The Windows Filtering Platform has blocked an application or service from listening on a port for incoming connections.....	413
5156(S): The Windows Filtering Platform has permitted a connection.....	414
5157(F): The Windows Filtering Platform has blocked a connection.....	420
5158(S): The Windows Filtering Platform has permitted a bind to a local port.....	426
5159(F): The Windows Filtering Platform has blocked a bind to a local port.....	431
Audit Filtering Platform Packet Drop.....	432
5152(F): The Windows Filtering Platform blocked a packet.....	433
5153(S): A more restrictive Windows Filtering Platform filter has blocked a packet.....	439
Audit Handle Manipulation.....	440
4658(S): The handle to an object was closed.....	440
4690(S): An attempt was made to duplicate a handle to an object.....	441
Audit Kernel Object.....	444
4656(S, F): A handle to an object was requested.....	444
4658(S): The handle to an object was closed.....	444
4660(S): An object was deleted.....	444

4663(S): An attempt was made to access an object.....	445
Audit Other Object Access Events.....	446
4671(-): An application attempted to access a blocked ordinal through the TBS.....	446
4691(S): Indirect access to an object was requested.....	447
5148(F): The Windows Filtering Platform has detected a DoS attack and entered a defensive mode; packets associated with this attack will be discarded.....	449
5149(F): The DoS attack has subsided and normal processing is being resumed.....	450
4698(S): A scheduled task was created.....	451
4699(S): A scheduled task was deleted.....	454
4700(S): A scheduled task was enabled.....	457
4701(S): A scheduled task was disabled.....	460
4702(S): A scheduled task was updated.....	463
5888(S): An object in the COM+ Catalog was modified.....	465
5889(S): An object was deleted from the COM+ Catalog.....	468
5890(S): An object was added to the COM+ Catalog.....	471
Audit Registry.....	475
4663(S): An attempt was made to access an object.....	475
4656(S, F): A handle to an object was requested.....	475
4658(S): The handle to an object was closed.....	475
4660(S): An object was deleted.....	475
4657(S): A registry value was modified.....	476
5039(-): A registry key was virtualized.....	479
4670(S): Permissions on an object were changed.....	480
Audit Removable Storage.....	481
4656(S, F): A handle to an object was requested.....	481
4658(S): The handle to an object was closed.....	481
4663(S): An attempt was made to access an object.....	481
Audit SAM.....	482

4661(S, F): A handle to an object was requested.....	482
Audit Central Policy Staging.....	483
4818(S): Proposed Central Access Policy does not grant the same access permissions as the current Central Access Policy.....	484
Policy Change.....	490
Audit Policy Change.....	490
4715(S): The audit policy (SACL) on an object was changed.....	491
4719(S): System audit policy was changed.....	495
4817(S): Auditing settings on object were changed.....	499
4902(S): The Per-user audit policy table was created.....	503
4906(S): The CrashOnAuditFail value has changed.....	505
4907(S): Auditing settings on object were changed.....	506
4908(S): Special Groups Logon table modified.....	512
4912(S): Per User Audit Policy was changed.....	514
4904(S): An attempt was made to register a security event source.....	517
4905(S): An attempt was made to unregister a security event source.....	520
Audit Authentication Policy Change.....	524
4670(S): Permissions on an object were changed.....	525
4706(S): A new trust was created to a domain.....	525
4707(S): A trust to a domain was removed.....	529
4716(S): Trusted domain information was modified.....	531
4713(S): Kerberos policy was changed.....	535
4717(S): System security access was granted to an account.....	537
4718(S): System security access was removed from an account.....	541
4739(S): Domain Policy was changed.....	545
4864(S): A namespace collision was detected.....	552
4865(S): A trusted forest information entry was added.....	553
4866(S): A trusted forest information entry was removed.....	556

4867(S): A trusted forest information entry was modified.....	559
Audit Authorization Policy Change.....	562
4703(S): A user right was adjusted.....	563
4704(S): A user right was assigned.....	571
4705(S): A user right was removed.....	578
4670(S): Permissions on an object were changed.....	584
4911(S): Resource attributes of the object were changed.....	585
4913(S): Central Access Policy on the object was changed.....	591
Audit Filtering Platform Policy Change.....	598
4709(S): IPsec Services was started.....	599
4710(S): IPsec Services was disabled.....	599
4711(S): May contain any one of the following:.....	599
4712(F): IPsec Services encountered a potentially serious failure.....	599
5040(S): A change has been made to IPsec settings. An Authentication Set was added.....	599
5041(S): A change has been made to IPsec settings. An Authentication Set was modified.....	599
5042(S): A change has been made to IPsec settings. An Authentication Set was deleted.....	599
5043(S): A change has been made to IPsec settings. A Connection Security Rule was added.....	599
5044(S): A change has been made to IPsec settings. A Connection Security Rule was modified.....	599
5045(S): A change has been made to IPsec settings. A Connection Security Rule was deleted.....	599
5046(S): A change has been made to IPsec settings. A Crypto Set was added.....	599
5047(S): A change has been made to IPsec settings. A Crypto Set was modified.....	599
5048(S): A change has been made to IPsec settings. A Crypto Set was deleted.....	599
5440(S): The following callout was present when the Windows Filtering Platform Base Filtering Engine started.....	599
5441(S): The following filter was present when the Windows Filtering Platform Base Filtering Engine started.....	599
5442(S): The following provider was present when the Windows Filtering Platform Base Filtering Engine started.....	599
5443(S): The following provider context was present when the Windows Filtering Platform Base Filtering Engine started.....	599
5444(S): The following sub-layer was present when the Windows Filtering Platform Base Filtering Engine started.....	599

5446(S): A Windows Filtering Platform callout has been changed.....	599
5448(S): A Windows Filtering Platform provider has been changed.....	599
5449(S): A Windows Filtering Platform provider context has been changed.....	599
5450(S): A Windows Filtering Platform sub-layer has been changed.....	600
5456(S): PASTore Engine applied Active Directory storage IPsec policy on the computer.....	600
5457(F): PASTore Engine failed to apply Active Directory storage IPsec policy on the computer.....	600
5458(S): PASTore Engine applied locally cached copy of Active Directory storage IPsec policy on the computer.....	600
5459(F): PASTore Engine failed to apply locally cached copy of Active Directory storage IPsec policy on the computer.....	600
5460(S): PASTore Engine applied local registry storage IPsec policy on the computer.....	600
5461(F): PASTore Engine failed to apply local registry storage IPsec policy on the computer.....	600
5462(F): PASTore Engine failed to apply some rules of the active IPsec policy on the computer. Use the IP Security Monitor snap-in to diagnose the problem.....	600
5463(S): PASTore Engine polled for changes to the active IPsec policy and detected no changes.....	600
5464(S): PASTore Engine polled for changes to the active IPsec policy, detected changes, and applied them to IPsec Services.....	600
5465(S): PASTore Engine received a control for forced reloading of IPsec policy and processed the control successfully.....	600
5466(F): PASTore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory cannot be reached, and will use the cached copy of the Active Directory IPsec policy instead. Any changes made to the Active Directory IPsec policy since the last poll could not be applied.....	600
5467(F): PASTore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory can be reached, and found no changes to the policy. The cached copy of the Active Directory IPsec policy is no longer being used.....	600
5468(S): PASTore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory can be reached, found changes to the policy, and applied those changes. The cached copy of the Active Directory IPsec policy is no longer being used.....	600
5471(S): PASTore Engine loaded local storage IPsec policy on the computer.....	600
5472(F): PASTore Engine failed to load local storage IPsec policy on the computer.....	600
5473(S): PASTore Engine loaded directory storage IPsec policy on the computer.....	600
5474(F): PASTore Engine failed to load directory storage IPsec policy on the computer.....	600
5477(F): PASTore Engine failed to add quick mode filter.....	600
Audit MPSSVC Rule-Level Policy Change.....	602
4944(S): The following policy was active when the Windows Firewall started.....	603
4945(S): A rule was listed when the Windows Firewall started.....	606

4946(S): A change has been made to Windows Firewall exception list. A rule was added.....	609
4947(S): A change has been made to Windows Firewall exception list. A rule was modified.....	611
4948(S): A change has been made to Windows Firewall exception list. A rule was deleted.....	614
4949(S): Windows Firewall settings were restored to the default values.....	616
4950(S): A Windows Firewall setting has changed.....	617
4951(F): A rule has been ignored because its major version number was not recognized by Windows Firewall.....	620
4952(F): Parts of a rule have been ignored because its minor version number was not recognized by Windows Firewall. The other parts of the rule will be enforced.....	622
4953(F): Windows Firewall ignored a rule because it could not be parsed.....	623
4954(S): Windows Firewall Group Policy settings have changed. The new settings have been applied.....	625
4956(S): Windows Firewall has changed the active profile.....	626
4957(F): Windows Firewall did not apply the following rule.....	628
4958(F): Windows Firewall did not apply the following rule because the rule referred to items not configured on this computer.....	630
Audit Other Policy Change Events.....	631
4714(S): Encrypted data recovery policy was changed.....	632
4819(S): Central Access Policies on the machine have been changed.....	633
4826(S): Boot Configuration Data loaded.....	636
4909(-): The local policy settings for the TBS were changed.....	638
4910(-): The group policy settings for the TBS were changed.....	638
5063(S, F): A cryptographic provider operation was attempted.....	638
5064(S, F): A cryptographic context operation was attempted.....	639
5065(S, F): A cryptographic context modification was attempted.....	640
5066(S, F): A cryptographic function operation was attempted.....	641
5067(S, F): A cryptographic function modification was attempted.....	642
5068(S, F): A cryptographic function provider operation was attempted.....	643
5069(S, F): A cryptographic function property operation was attempted.....	644
5070(S, F): A cryptographic function property modification was attempted.....	645
5447(S): A Windows Filtering Platform filter has been changed.....	647

6144(S): Security policy in the group policy objects has been applied successfully.....	648
6145(F): One or more errors occurred while processing security policy in the group policy objects.....	650
Privilege Use.....	652
Audit Non Sensitive Privilege Use.....	652
4673(S, F): A privileged service was called.....	653
4674(S, F): An operation was attempted on a privileged object.....	653
4985(S): The state of a transaction has changed.....	653
Audit Other Privilege Use Events.....	654
4985(S): The state of a transaction has changed.....	654
Audit Sensitive Privilege Use.....	655
4673(S, F): A privileged service was called.....	656
4674(S, F): An operation was attempted on a privileged object.....	662
4985(S): The state of a transaction has changed.....	669
System.....	670
Audit IPsec Driver.....	670
4960(S): IPsec dropped an inbound packet that failed an integrity check. If this problem persists, it could indicate a network issue or that packets are being modified in transit to this computer. Verify that the packets sent from the remote computer are the same as those received by this computer. This error might also indicate interoperability problems with other IPsec implementations.....	671
4961(S): IPsec dropped an inbound packet that failed a replay check. If this problem persists, it could indicate a replay attack against this computer.....	671
4962(S): IPsec dropped an inbound packet that failed a replay check. The inbound packet had too low a sequence number to ensure it was not a replay.....	671
4963(S): IPsec dropped an inbound clear text packet that should have been secured. This is usually due to the remote computer changing its IPsec policy without informing this computer. This could also be a spoofing attack attempt.....	671
4965(S): IPsec received a packet from a remote computer with an incorrect Security Parameter Index (SPI). This is usually caused by malfunctioning hardware that is corrupting packets. If these errors persist, verify that the packets sent from the remote computer are the same as those received by this computer. This error may also indicate interoperability problems with other IPsec implementations. In that case, if connectivity is not impeded, then these events can be ignored.....	671
5478(S): IPsec Services has started successfully.....	671
5479(): IPsec Services has been shut down successfully. The shutdown of IPsec Services can put the computer at greater risk of network attack or expose the computer to potential security risks.....	671

5480(F): IPsec Services failed to get the complete list of network interfaces on the computer. This poses a potential security risk because some of the network interfaces may not get the protection provided by the applied IPsec filters. Use the IP Security Monitor snap-in to diagnose the problem.....	671
5483(F): IPsec Services failed to initialize RPC server. IPsec Services could not be started.....	671
5484(F): IPsec Services has experienced a critical failure and has been shut down. The shutdown of IPsec Services can put the computer at greater risk of network attack or expose the computer to potential security risks.....	671
5485(F): IPsec Services failed to process some IPsec filters on a plug-and-play event for network interfaces. This poses a potential security risk because some of the network interfaces may not get the protection provided by the applied IPsec filters. Use the IP Security Monitor snap-in to diagnose the problem.....	671
Audit Other System Events.....	671
5024(S): The Windows Firewall Service has started successfully.....	673
5025(S): The Windows Firewall Service has been stopped.....	674
5027(F): The Windows Firewall Service was unable to retrieve the security policy from the local storage. The service will continue enforcing the current policy.....	675
5028(F): The Windows Firewall Service was unable to parse the new security policy. The service will continue with currently enforced policy.....	676
5029(F): The Windows Firewall Service failed to initialize the driver. The service will continue to enforce the current policy.....	677
5030(F): The Windows Firewall Service failed to start.....	677
5032(F): Windows Firewall was unable to notify the user that it blocked an application from accepting incoming connections on the network.....	678
5033(S): The Windows Firewall Driver has started successfully.....	678
5034(S): The Windows Firewall Driver was stopped.....	679
5035(F): The Windows Firewall Driver failed to start.....	680
5037(F): The Windows Firewall Driver detected critical runtime error. Terminating.....	680
5058(S, F): Key file operation.....	681
5059(S, F): Key migration operation.....	685
6400(-): BranchCache: Received an incorrectly formatted response while discovering availability of content.....	688
6401(-): BranchCache: Received invalid data from a peer. Data discarded.....	688
6402(-): BranchCache: The message to the hosted cache offering it data is incorrectly formatted.....	688
6403(-): BranchCache: The hosted cache sent an incorrectly formatted response to the client.....	689
6404(-): BranchCache: Hosted cache could not be authenticated using the provisioned SSL certificate.....	689
6405(-): BranchCache: %2 instance(s) of event id %1 occurred.....	690
6406(-): %1 registered to Windows Firewall to control filtering for the following: %2.....	690

6407(-): 1%.....	690
6408(-): Registered product %1 failed and Windows Firewall is now controlling the filtering for %2.....	691
6409(-): BranchCache: A service connection point object could not be parsed.....	691
Audit Security State Change.....	692
4608(S): Windows is starting up.....	693
4609(S): Windows is shutting down.....	693
4616(S): The system time was changed.....	694
4621(S): Administrator recovered system from CrashOnAuditFail.....	697
Audit Security System Extension.....	698
4610(S): An authentication package has been loaded by the Local Security Authority.....	699
4611(S): A trusted logon process has been registered with the Local Security Authority.....	700
4614(S): A notification package has been loaded by the Security Account Manager.....	702
4622(S): A security package has been loaded by the Local Security Authority.....	703
4697(S): A service was installed in the system.....	705
Audit System Integrity.....	709
4612(S): Internal resources allocated for the queuing of audit messages have been exhausted, leading to the loss of some audits.....	710
4615(S): Invalid use of LPC port.....	710
4618(S): A monitored security event pattern has occurred.....	711
4816(S): RPC detected an integrity violation while decrypting an incoming message.....	713
5038(F): Code integrity determined that the image hash of a file is not valid. The file could be corrupt due to unauthorized modification or the invalid hash could indicate a potential disk device error.....	713
5056(S): A cryptographic self-test was performed.....	714
5062(S): A kernel-mode cryptographic self-test was performed.....	714
5057(F): A cryptographic primitive operation failed.....	715
5060(F): Verification operation failed.....	716
5061(S, F): Cryptographic operation.....	717
6281(F): Code Integrity determined that the page hashes of an image file are not valid. The file could be improperly signed without page hashes or corrupt due to unauthorized modification. The invalid hashes could indicate a potential disk device error.....	720

6410(F): Code integrity determined that a file does not meet the security requirements to load into a process.....	720
Other Events.....	722
1100(S): The event logging service has shut down.....	722
1102(S): The audit log was cleared.....	723
1104(S): The security log is now full.....	725
1105(S): Event log automatic backup.....	726
1108(S): The event logging service encountered an error while processing an incoming event published from %1.....	727
Appendix A: Security monitoring recommendations for many audit events.....	730
Appendix B: List of Tables.....	731

Account Logon

Audit Credential Validation

Audit Credential Validation determines whether the operating system generates audit events on credentials that are submitted for a user account logon request. These events occur on the computer that is authoritative for the credentials as follows:

- For domain accounts, the domain controller is authoritative.
- For local accounts, the local computer is authoritative.

Event volume:

- High on domain controllers.
- Low on member servers and workstations.

Because domain accounts are used much more frequently than local accounts in enterprise environments, most of the Account Logon events in a domain environment occur on the domain controllers that are authoritative for the domain accounts. However, these events can occur on any computer, and they may occur in conjunction with or on separate computers from Logon and Logoff events.

The main reason to enable this auditing subcategory is to handle local accounts authentication attempts and, for domain accounts, NTLM authentication in the domain. It is especially useful for monitoring unsuccessful attempts, to find brute-force attacks, account enumeration, and potential account compromise events on domain controllers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	Yes	Yes	Yes	<p>Expected volume of events is high for domain controllers, because this subcategory will generate events when an authentication attempt is made using any domain account and NTLM authentication.</p> <p>IF – We recommend Success auditing to keep track of domain-account authentication events using the NTLM protocol. Expect a high volume of events. For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. Just collecting Success auditing events in this subcategory for future use in case of a security incident is not very useful, because events in this subcategory are not always informative.</p> <p>We recommend Failure auditing, to collect information about failed authentication attempts using domain accounts and the NTLM authentication protocol.</p>
Member Server	Yes	Yes	Yes	Yes	<p>Expected volume of events is low for member servers, because this subcategory will generate events when an authentication attempt is made using a local account, which should not happen too often.</p> <p>We recommend Success auditing, to keep track of authentication events by local accounts.</p> <p>We recommend Failure auditing, to collect information about failed authentication attempts by local accounts.</p>
Workstation	Yes	Yes	Yes	Yes	<p>Expected volume of events is low for workstations, because this subcategory will generate events when an authentication attempt is made using a local account, which should not happen too often.</p> <p>We recommend Success auditing, to keep track of authentication events by local accounts.</p> <p>We recommend Failure auditing, to collect information about failed authentication attempts by local accounts.</p>

Events List:

- [4774\(S\)](#): An account was mapped for logon.
- [4775\(F\)](#): An account could not be mapped for logon.
- [4776\(S, F\)](#): The computer attempted to validate the credentials for an account.
- [4777\(F\)](#): The domain controller failed to validate the credentials for an account.

4774(S): An account was mapped for logon.

It appears that this event never occurs.

Event Schema:

An account was mapped for logon.

Authentication Package:%1

Account UPN:%2

Mapped Name:%3

Required Server Roles: no information.

Minimum OS Version: no information.

Event Versions: 0.

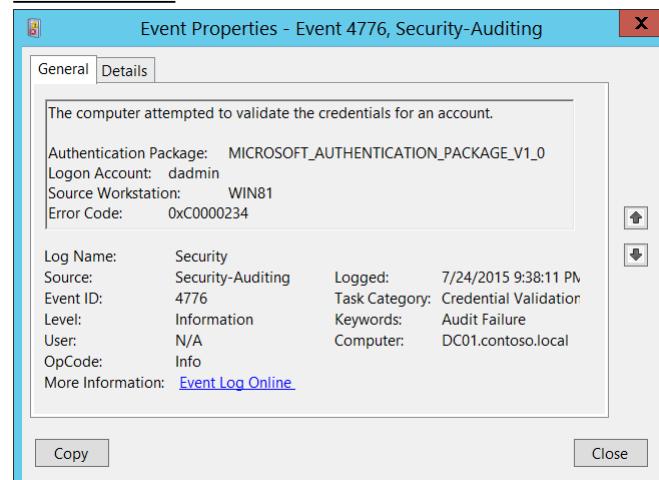
Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4775(F): An account could not be mapped for logon.

It appears that this event never occurs.

Event Schema:



The computer attempted to validate the credentials for an account.

Authentication Package: MICROSOFT_AUTHENTICATION_PACKAGE_V1_0
Logon Account: dadmin
Source Workstation: WIN81
Error Code: 0xC0000234

Log Name: Security
Source: Security-Auditing
Event ID: 4776
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy Close

An account could not be mapped for logon.

Authentication Package:%1

Account Name:%2

Required Server Roles: no information.

Minimum OS Version: no information.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4776(S, F): The computer attempted to validate the credentials for an account.

Event Description:

This event generates every time that a credential validation occurs using NTLM authentication.

This event occurs only on the computer that is authoritative for the provided credentials. For domain accounts, the domain controller is authoritative. For local accounts, the local computer is authoritative.

It shows successful and unsuccessful credential validation attempts.

It shows only the computer name (**Source Workstation**) from which the authentication attempt was performed (authentication source). For example, if you authenticate from CLIENT-1 to SERVER-1 using a domain account you will see CLIENT-1 in the **Source Workstation** field. Information about the destination computer (SERVER-1) is not presented in this event.

If a credential validation attempt fails, you will see a Failure event with **Error Code** parameter value not equal to “**0x0**”.

The main advantage of this event is that on domain controllers you can see all authentication attempts for domain accounts when NTLM authentication was used.

For monitoring local account logon attempts, it is better to use event “[4624](#): An account was successfully logged on” because it contains more details and is more informative.

This event also generates when a workstation unlock event occurs.

This event does not generate when a domain account logs on locally to a domain controller.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4776</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14336</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-07-25T04:38:11.003163100Z" />
<EventRecordID>165437</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="532" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="PackageName">MICROSOFT_AUTHENTICATION_PACKAGE_V1_0</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="Workstation">WIN81</Data>
<Data Name="Status">0xc0000234</Data>
</EventData>
</Event>
```

Required Server Roles: no specific requirements.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

- **Authentication Package** [Type = UnicodeString]: the name of [Authentication Package](#) which was used for credential validation. It is always "MICROSOFT_AUTHENTICATION_PACKAGE_V1_0" for [4776](#) event.

Authentication package is a DLL that encapsulates the authentication logic used to determine whether to permit a user to log on. [Local Security Authority](#) (LSA) authenticates a user logon by sending the request to an authentication package. The authentication package then examines the logon information and either authenticates or rejects the user logon attempt.

- **Logon Account** [Type = UnicodeString]: the name of the account that had its credentials validated by the **Authentication Package**. Can be user name, computer account name or [well-known security principal](#) account name. Examples:
 - User example: dadmin
 - Computer account example: WIN81\$
 - Local System account example: Local
 - Local Service account example: Local Service
- **Source Workstation** [Type = UnicodeString]: the name of the computer from which the logon attempt originated.
- **Error Code** [Type = HexInt32]: contains error code for Failure events. For Success events this parameter has "0x0" value. The table below contains most common error codes for this event:

Error Code	Description
0xC0000064	The username you typed does not exist. Bad username.
0xC000006A	Account logon with misspelled or bad password.
0xC000006D	Generic logon failure. Some of the potential causes for this: <ul style="list-style-type: none"> • An invalid username and/or password was used • LAN Manager Authentication Level mismatch between the source and target computers.
0xC000006F	Account logon outside authorized hours.
0xC0000070	Account logon from unauthorized workstation.
0xC0000071	Account logon with expired password.
0xC0000072	Account logon to account disabled by administrator.
0xC0000193	Account logon with expired account.
0xC0000224	Account logon with "Change Password at Next Logon" flagged.
0xC0000234	Account logon with account locked.
0xc0000371	The local account store does not contain secret material for the specified account.
0x0	No errors.

Table 1. Winlogon Error Codes.

Security Monitoring Recommendations:

For 4776(S, F): The computer attempted to validate the credentials for an account.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Logon Account ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Logon Account ” value (with other information) to monitor how or when a particular account is being used. To monitor activity of specific user accounts outside of working hours, monitor the appropriate Logon Account + Source Workstation pairs.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Logon Account ” that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Logon Account ” for accounts that are outside the whitelist.
Restricted-use computers: You might have certain computers from which certain people (accounts) should not log on.	Monitor the target Source Workstation for credential validation requests from the “ Logon Account ” that you are concerned about.
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Logon Account ” for names that don’t comply with naming conventions.

- If NTLM authentication should not be used for a specific account, monitor for that account. Don’t forget that local logon will always use NTLM authentication if an account logs on to a device where its user account is stored.
- You can use this event to collect all NTLM authentication attempts in the domain, if needed. Don’t forget that local logon will always use NTLM authentication if the account logs on to a device where its user account is stored.
- If a local account should be used only locally (for example, network logon or terminal services logon is not allowed), you need to monitor for all events where **Source Workstation** and **Computer** (where the event was generated and where the credentials are stored) have different values.
- Consider tracking the following errors for the reasons listed:

Error to track	What the error might indicate
User logon with misspelled or bad user account	For example, N events in the last N minutes can be an indicator of an account enumeration attack, especially relevant for highly critical accounts.

User logon with misspelled or bad password	For example, N events in the last N minutes can be an indicator of a brute-force password attack, especially relevant for highly critical accounts.
User logon outside authorized hours	Can indicate a compromised account; especially relevant for highly critical accounts.
User logon from unauthorized workstation	Can indicate a compromised account; especially relevant for highly critical accounts.
User logon to account disabled by administrator	For example, N events in last N minutes can be an indicator of an account compromise attempt, especially relevant for highly critical accounts.
User logon with expired account	Can indicate an account compromise attempt; especially relevant for highly critical accounts.
User logon with account locked	Can indicate a brute-force password attack; especially relevant for highly critical accounts.

4777(F): The domain controller failed to validate the credentials for an account.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system. [4776](#) failure event is generated instead.

Audit Kerberos Authentication Service

Audit Kerberos Authentication Service determines whether to generate audit events for Kerberos authentication ticket-granting ticket (TGT) requests.

If you configure this policy setting, an audit event is generated after a Kerberos authentication TGT request. Success audits record successful attempts and Failure audits record unsuccessful attempts.

Event volume: High on Kerberos Key Distribution Center servers.

This subcategory contains events about issued TGTs and failed TGT requests. It also contains events about failed Pre-Authentications, due to wrong user password or when the user's password has expired.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	We recommend Success auditing, because you will see all Kerberos Authentication requests (TGT requests), which are a part of domain account logons. Also, you can see the IP address from which this account requested a TGT, when TGT was requested, which encryption type was used and so on. We recommend Failure auditing, because you will see all failed requests with wrong password, username, revoked certificate, and so on. You will also be able to detect Kerberos issues or possible attack attempts. Expected volume is high on domain controllers.
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Events List:

- [4768](#)(S, F): A Kerberos authentication ticket (TGT) was requested.
- [4771](#)(F): Kerberos pre-authentication failed.
- [4772](#)(F): A Kerberos authentication ticket request failed.

4768(S, F): A Kerberos authentication ticket (TGT) was requested.

Event Properties - Event 4768, Microsoft Windows security auditing.

General Details

A Kerberos authentication ticket (TGT) was requested.

Account Information:
 Account Name: dadmin
 Supplied Realm Name: CONTOSO.LOCAL

Service Name: krbtgt
 Service ID: CONTOSO\krbtgt

Network Information:
 Client Address: ::ffff:10.0.0.12
 Client Port: 49273

Additional Information:
 Ticket Options: 0x40810010
 Result Code: 0x0
 Ticket Encryption Type: 0x12
 Pre-Authentication Type: 15

Certificate Information:
 Certificate Issuer Name: contoso-DC01-CA-1
 Certificate Serial Number: 1D000000D292FBE3C6CDDAFA20002000000D
 Certificate Thumbprint: 564DFAEE99C71D62ABC553E695BD8DBC4669413

Certificate information is only provided if a certificate was used for pre-authentication.

Pre-authentication types, ticket options, encryption types and result codes are defined in RFC 4120.

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 4768
 Level: Information
 User: N/A
 OpCode: Info
 Task Category: Kerberos Authentication Service
 Keywords: Audit Success
 Computer: DC01.contoso.local

More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time Key Distribution Center issues a Kerberos Ticket Granting Ticket (TGT). This event generates only on domain controllers.

If TGT issue fails then you will see Failure event with **Result Code** field not equal to “**0x0**”.

This event doesn't generate for **Result Codes**: 0x10, 0x17 and 0x18. Event “[4771](#): Kerberos pre-authentication failed.” generates instead.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4768</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14339</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-07T18:13:46.074535600Z" />
<EventRecordID>166747</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1496" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
```

```
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO.LOCAL</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="ServiceName">krbtgt</Data>
<Data Name="ServiceSid">S-1-5-21-3457937927-2839227994-823803824-502</Data>
<Data Name="TicketOptions">0x40810010</Data>
<Data Name="Status">0x0</Data>
<Data Name="TicketEncryptionType">0x12</Data>
```

```
<Data Name="PreAuthType">15</Data>
<Data Name="IpAddress">::ffff:10.0.0.12</Data>
<Data Name="IpPort">49273</Data>
<Data Name="CertIssuerName">contoso-DC01-CA-1</Data>
<Data Name="CertSerialNumber">1D0000000D292FBE3C6CDDAFA200020000000D</Data>
<Data Name="CertThumbprint">564DFAEE99C71D62ABC553E695BD8DBC46669413</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Account Information:

- **Account Name** [Type = UnicodeString]: the name of account, for which (TGT) ticket was requested. Computer account name ends with \$ character.
 - User account example: dadmin
 - Computer account example: WIN81\$
- **Supplied Realm Name** [Type = UnicodeString]: the name of the Kerberos Realm that **Account Name** belongs to. This can appear in a variety of formats, including the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

A **Kerberos Realm** is a set of managed nodes that share the same Kerberos database. The Kerberos database resides on the Kerberos master computer system, which should be kept in a physically secure room. Active Directory domain is the example of Kerberos Realm in the Microsoft Windows Active Directory world.

- **User ID** [Type = SID]: SID of account for which (TGT) ticket was requested. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

For example: CONTOSO\dadmin or CONTOSO\WIN81\$.

- **NULL SID** – this value shows in [4768](#) Failure events.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

Service Information:

- **Service Name** [Type = UnicodeString]: the name of the service in the Kerberos Realm to which TGT request was sent. Typically has value “**krbtgt**” for TGT requests, which means Ticket Granting Ticket issuing service.
 - For Failure events **Service Name** typically has the following format: **krbtgt/REALM_NAME**. For example: krbtgt/CONTOSO.
- **Service ID** [Type = SID]: SID of the service account in the Kerberos Realm to which TGT request was sent. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Domain controllers have a specific service account (**krbtgt**) that is used by the [Key Distribution Center \(KDC\)](#) service to issue Kerberos tickets. It has a built-in, pre-defined SID: S-1-5-21-[DOMAIN IDENTIFIER](#)-502.

- **NULL SID** – this value shows in [4768](#) Failure events.

Network Information:

- **Client Address** [Type = UnicodeString]: IP address of the computer from which the TGT request was received. Formats vary, and include the following:
 - IPv6 or IPv4 address.
 - ::ffff:IPv4_address.
 - ::1 - localhost.
- **Client Port** [Type = UnicodeString]: source port number of client network connection (TGT request connection).
 - 0 for local (localhost) requests.

Additional information:

- **Ticket Options** [Type = HexInt32]: this is a set of different [Ticket Flags](#) in hexadecimal format.

Example:

- Ticket Options: 0x40810010
- Binary view: 010000001000000100000000000010000
- Using **MSB 0** bit numbering we have bit 1, 8, 15 and 27 set = Forwardable, Renewable, Canonicalize, Renewable-ok.

In the table below “**MSB 0**” bit numbering is used, because RFC documents use this style. In “**MSB 0**” style bit numbering begins from left.

	0		7
1	0	0	1

The most common values:

- 0x40810010 - Forwardable, Renewable, Canonicalize, Renewable-ok
- 0x40810000 - Forwardable, Renewable, Canonicalize
- 0x60810010 - Forwardable, Forwarded, Renewable, Canonicalize, Renewable-ok

Bit	Flag Name	Description
0	Reserved	-
1	Forwardable	(TGT only). Tells the ticket-granting service that it can issue a new TGT—based on the presented TGT—with a different network address based on the presented TGT.
2	Forwarded	Indicates either that a TGT has been forwarded or that a ticket was issued from a forwarded TGT.
3	Proxyable	(TGT only). Tells the ticket-granting service that it can issue tickets with a network address that differs from the one in the TGT.
4	Proxy	Indicates that the network address in the ticket is different from the one in the TGT used to obtain the ticket.
5	Allow-postdate	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
6	Postdated	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
7	Invalid	This flag indicates that a ticket is invalid, and it must be validated by the KDC before use. Application servers must reject tickets which have this flag set.
8	Renewable	Used in combination with the End Time and Renew Till fields to cause tickets with long life spans to be renewed at the KDC

		periodically.
9	Initial	Indicates that a ticket was issued using the authentication service (AS) exchange and not issued based on a TGT.
10	Pre-authent	Indicates that the client was authenticated by the KDC before a ticket was issued. This flag usually indicates the presence of an authenticator in the ticket. It can also flag the presence of credentials taken from a smart card logon.
11	Opt-hardware-auth	This flag was originally intended to indicate that hardware-supported authentication was used during pre-authentication. This flag is no longer recommended in the Kerberos V5 protocol. KDCs MUST NOT issue a ticket with this flag set. KDCs SHOULD NOT preserve this flag if it is set by another KDC.
12	Transited-policy-checked	KILE MUST NOT check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag.
13	Ok-as-delegate	The KDC MUST set the OK-AS-DELEGATE flag if the service account is trusted for delegation.
14	Request-anonymous	KILE not use this flag.
15	Name-canonicalize	In order to request referrals the Kerberos client MUST explicitly request the "canonicalize" KDC option for the AS-REQ or TGS-REQ.
16-25	Unused	-
26	Disable-transited-check	By default the KDC will check the transited field of a TGT against the policy of the local realm before it will issue derivative tickets based on the TGT. If this flag is set in the request, checking of the transited field is disabled. Tickets issued without the performance of this check will be noted by the reset (0) value of the TRANSITED-POLICY-CHECKED flag, indicating to the application server that the transited field must be checked locally. KDCs are encouraged but not required to honor the DISABLE-TRANSITED-CHECK option. Should not be in use, because Transited-policy-checked flag is not supported by KILE.
27	Renewable-ok	The RENEWABLE-OK option indicates that a renewable ticket will be acceptable if a ticket with the requested life cannot otherwise be provided, in which case a renewable ticket may be issued with a renew-till equal to the requested end time. The value of the renew-till field may still be limited by local limits, or limits selected by the individual principal or server.
28	Enc-tkt-in-skey	No information.
29	Unused	-
30	Renew	The RENEW option indicates that the present request is for a renewal. The ticket provided is encrypted in the secret key for the server on which it is valid. This option will only be honored if the ticket to be renewed has its RENEWABLE flag set and if the time in its renew-till field has not passed. The ticket to be renewed is passed in the padata field as part of the authentication header.
31	Validate	This option is used only by the ticket-granting service. The VALIDATE option indicates that the request is to validate a postdated ticket. Should not be in use, because postdated tickets are not supported by KILE.

Table 2. Kerberos ticket flags.

KILE (**M**icrosoft **K**erberos **P**rotocol **E**xtension) – Kerberos protocol extensions used in Microsoft operating systems. These extensions provide additional capability for authorization information including group memberships, interactive logon information, and integrity levels.

- **Result Code** [Type = HexInt32]: hexadecimal result code of TGT issue operation. The “Table 3. TGT/TGS issue error codes.” contains the list of the most common error codes for this event.

Code	Code Name	Description	Possible causes
0x0	KDC_ERR_NONE	No error	No errors were found.
0x1	KDC_ERR_NAME_EXP	Client's entry in KDC database has expired	No information.
0x2	KDC_ERR_SERVICE_EXP	Server's entry in KDC database has expired	No information.
0x3	KDC_ERR_BAD_PVNO	Requested Kerberos version number not supported	No information.
0x4	KDC_ERR_C_OLD_MAST_KVNO	Client's key encrypted in old master key	No information.
0x5	KDC_ERR_S_OLD_MAST_KVNO	Server's key encrypted in old master key	No information.
0x6	KDC_ERR_C_PRINCIPAL_UNKNOWN	Client not found in Kerberos database	The username doesn't exist.
0x7	KDC_ERR_S_PRINCIPAL_UNKNOWN	Server not found in Kerberos database	This error can occur if the domain controller cannot find the server's name in Active Directory. This error is similar to KDC_ERR_C_PRINCIPAL_UNKNOWN except that it occurs when the server name cannot be found.
0x8	KDC_ERR_PRINCIPAL_NOT_UNIQUE	Multiple principal entries in KDC database	This error occurs if duplicate principal names exist. Unique principal names are crucial for ensuring mutual authentication. Thus, duplicate principal names are strictly forbidden, even across multiple realms. Without unique principal names, the client has no way of ensuring that the server it is communicating with is the correct one.
0x9	KDC_ERR_NULL_KEY	The client or server has a null key (master key)	No master key was found for client or server. Usually it means that administrator should reset the password on the account.
0xA	KDC_ERR_CANNOT_POSTDATE	Ticket (TGT) not eligible for postdating	This error can occur if a client requests postdating of a Kerberos ticket. Postdating is the act of requesting that a ticket's start time be set into the future. It also can occur if there is a time difference between the client and the KDC.
0xB	KDC_ERR_NEVER_VALID	Requested start time is later than end time	There is a time difference between the KDC and the client.
0xC	KDC_ERR_POLICY	Requested start time is later than end time	This error is usually the result of logon restrictions in place on a user's account. For example workstation restriction, smart card authentication requirement or logon time restriction.
0xD	KDC_ERR_BADOPTION	KDC cannot accommodate requested option	Impending expiration of a TGT. The SPN to which the client is attempting to delegate credentials is not in its Allowed-to-delegate-to list
0xE	KDC_ERR_ETYPE_NOTSUPP	KDC has no support for encryption type	In general, this error occurs when the KDC or a client receives a packet that it cannot decrypt.
0xF	KDC_ERR_SUMTYPE_NOSUPP	KDC has no support for checksum type	The KDC, server, or client receives a packet for which it does not have a

			key of the appropriate encryption type. The result is that the computer is unable to decrypt the ticket.
0x10	KDC_ERR_PADATA_TYPE_NOSUPP	KDC has no support for PADATA type (pre-authentication data)	Smart card logon is being attempted and the proper certificate cannot be located. This can happen because the wrong certification authority (CA) is being queried or the proper CA cannot be contacted. It can also happen when a domain controller doesn't have a certificate installed for smart cards (Domain Controller or Domain Controller Authentication templates). This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x11	KDC_ERR_TRTYPE_NO_SUPP	KDC has no support for transited type	No information.
0x12	KDC_ERR_CLIENT_REVOKED	Client's credentials have been revoked	This might be because of an explicit disabling or because of other restrictions in place on the account. For example: account disabled, expired, or locked out.
0x13	KDC_ERR_SERVICE_REVOKED	Credentials for server have been revoked	No information.
0x14	KDC_ERR_TGT_REVOKED	TGT has been revoked	Since the remote KDC may change its PKCROSS key while there are PKCROSS tickets still active, it SHOULD cache the old PKCROSS keys until the last issued PKCROSS ticket expires. Otherwise, the remote KDC will respond to a client with a KRB-ERROR message of type KDC_ERR_TGT_REVOKED. See RFC1510 for more details.
0x15	KDC_ERR_CLIENT_NOTYET	Client not yet valid—try again later	No information.
0x16	KDC_ERR_SERVICE_NOTYET	Server not yet valid—try again later	No information.
0x17	KDC_ERR_KEY_EXPIRED	Password has expired—change password to reset	The user's password has expired. This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x18	KDC_ERR_PREAUTH_FAILED	Pre-authentication information was invalid	The wrong password was provided. This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x19	KDC_ERR_PREAUTH_REQUIRED	Additional pre-authentication required	This error often occurs in UNIX interoperability scenarios. MIT-Kerberos clients do not request pre-authentication when they send a KRB_AS_REQ message. If pre-authentication is required (the default), Windows systems will send this error. Most MIT-Kerberos clients will respond to this error by giving the pre-authentication, in which case the error can be ignored, but some clients might not respond in this way.

0x1A	KDC_ERR_SERVER_NOMATCH	KDC does not know about the requested server	No information.
0x1B	KDC_ERR_SVC_UNAVAILABLE	KDC is unavailable	No information.
0x1F	KRB_AP_ERR_BAD_INTEGRITY	Integrity check on decrypted field failed	The authenticator was encrypted with something other than the session key. The result is that the client cannot decrypt the resulting message. The modification of the message could be the result of an attack or it could be because of network noise.
0x20	KRB_AP_ERR_TKT_EXPIRED	The ticket has expired	The smaller the value for the “Maximum lifetime for user ticket” Kerberos policy setting, the more likely it is that this error will occur. Because ticket renewal is automatic, you should not have to do anything if you get this message.
0x21	KRB_AP_ERR_TKT_NYV	The ticket is not yet valid	The ticket presented to the server is not yet valid (in relationship to the server time). The most probable cause is that the clocks on the KDC and the client are not synchronized. If cross-realm Kerberos authentication is being attempted, then you should verify time synchronization between the KDC in the target realm and the KDC in the client realm, as well.
0x22	KRB_AP_ERR_REPEAT	The request is a replay	This error indicates that a specific authenticator showed up twice — the KDC has detected that this session ticket duplicates one that it has already received.
0x23	KRB_AP_ERR_NOT_US	The ticket is not for us	The server has received a ticket that was meant for a different realm.
0x24	KRB_AP_ERR_BADMATCH	The ticket and authenticator do not match	The KRB_TGS_REQ is being sent to the wrong KDC. There is an account mismatch during protocol transition.
0x25	KRB_AP_ERR_SKEW	The clock skew is too great	This error is logged if a client computer sends a timestamp whose value differs from that of the server’s timestamp by more than the number of minutes found in the “Maximum tolerance for computer clock synchronization” setting in Kerberos policy.
0x26	KRB_AP_ERR_BADADDR	Network address in network layer header doesn't match address inside ticket	Session tickets MAY include the addresses from which they are valid. This error can occur if the address of the computer sending the ticket is different from the valid address in the ticket. A possible cause of this could be an Internet Protocol (IP) address change. Another possible cause is when a ticket is passed through a proxy server or NAT. The client is unaware of the address scheme used by the proxy server, so unless the program caused the client to request a proxy server ticket with the proxy server's source address, the ticket could be invalid.
0x27	KRB_AP_ERR_BADVERSION	Protocol version numbers don't match (PVNO)	When an application receives a KRB_SAFE message, it verifies it. If any error occurs, an error code is reported for use by the application. The message is first checked by verifying that the protocol version and

			type fields match the current version and KRB_SAFE, respectively. A mismatch generates a KRB_AP_ERR_BADVERSION. See RFC4120 for more details.
0x28	KRB_AP_ERR_MSG_TYPE	Message type is unsupported	This message is generated when target server finds that message format is wrong. This applies to KRB_AP_REQ, KRB_SAFE, KRB_PRIV and KRB_CRED messages. This error also generated if use of UDP protocol is being attempted with User-to-User authentication.
0x29	KRB_AP_ERR_MODIFIED	Message stream modified and checksum didn't match	The authentication data was encrypted with the wrong key for the intended server. The authentication data was modified in transit by a hardware or software error, or by an attacker. The client sent the authentication data to the wrong server because incorrect DNS data caused the client to send the request to the wrong server. The client sent the authentication data to the wrong server because DNS data was out-of-date on the client.
0x2A	KRB_AP_ERR_BADORDER	Message out of order (possible tampering)	This event generates for KRB_SAFE and KRB_PRIV messages if an incorrect sequence number is included, or if a sequence number is expected but not present. See RFC4120 for more details.
0x2C	KRB_AP_ERR_BADKEYVER	Specified version of key is not available	This error might be generated on server side during receipt of invalid KRB_AP_REQ message. If the key version indicated by the Ticket in the KRB_AP_REQ is not one the server can use (e.g., it indicates an old key, and the server no longer possesses a copy of the old key), the KRB_AP_ERR_BADKEYVER error is returned.
0x2D	KRB_AP_ERR_NOKEY	Service key not available	This error might be generated on server side during receipt of invalid KRB_AP_REQ message. Because it is possible for the server to be registered in multiple realms, with different keys in each, the realm field in the unencrypted portion of the ticket in the KRB_AP_REQ is used to specify which secret key the server should use to decrypt that ticket. The KRB_AP_ERR_NOKEY error code is returned if the server doesn't have the proper key to decipher the ticket.
0x2E	KRB_AP_ERR_MUT_FAIL	Mutual authentication failed	No information.
0x2F	KRB_AP_ERR_BADDIRECTION	Incorrect message direction	No information.
0x30	KRB_AP_ERR_METHOD	Alternative authentication method required	According RFC4120 this error message is obsolete.
0x31	KRB_AP_ERR_BADSEQ	Incorrect sequence number in message	No information.
0x32	KRB_AP_ERR_INAPP_CKSUM	Inappropriate type of checksum in message (checksum may be unsupported)	When KDC receives KRB_TGS_REQ message it decrypts it, and after that, the user-supplied checksum in the Authenticator MUST be

			verified against the contents of the request. The message MUST be rejected either if the checksums do not match (with an error code of KRB_AP_ERR_MODIFIED) or if the checksum is not collision-proof (with an error code of KRB_AP_ERR_INAPP_CKSUM).
0x33	KRB_AP_PATH_NOT_ACCEPTED	Desired path is unreachable	No information.
0x34	KRB_ERR_RESPONSE_TOO_BIG	Too much data	The size of a ticket is too large to be transmitted reliably via UDP. In a Windows environment, this message is purely informational. A computer running a Windows operating system will automatically try TCP if UDP fails.
0x3C	KRB_ERR_GENERIC	Generic error	Group membership has overloaded the PAC. Multiple recent password changes have not propagated. Crypto subsystem error caused by running out of memory. SPN too long. SPN has too many parts.
0x3D	KRB_ERR_FIELD_TOOLONG	Field is too long for this implementation	Each request (KRB_KDC_REQ) and response (KRB_KDC REP or KRB_ERROR) sent over the TCP stream is preceded by the length of the request as 4 octets in network byte order. The high bit of the length is reserved for future expansion and MUST currently be set to zero. If a KDC that does not understand how to interpret a set high bit of the length encoding receives a request with the high order bit of the length set, it MUST return a KRB-ERROR message with the error KRB_ERR_FIELD_TOOLONG and MUST close the TCP stream.
0x3E	KDC_ERR_CLIENT_NOT_TRUSTED	The client trust failed or is not implemented	This typically happens when user's smart-card certificate is revoked or the root Certification Authority that issued the smart card certificate (in a chain) is not trusted by the domain controller.
0x3F	KDC_ERR_KDC_NOT_TRUSTED	The KDC server trust failed or could not be verified	The trustedCertifiers field contains a list of certification authorities trusted by the client, in the case that the client does not possess the KDC's public key certificate. If the KDC has no certificate signed by any of the trustedCertifiers, then it returns an error of type KDC_ERR_KDC_NOT_TRUSTED. See RFC1510 for more details.
0x40	KDC_ERR_INVALID_SIG	The signature is invalid	This error is related to PKINIT. If a PKI trust relationship exists, the KDC then verifies the client's signature on AuthPack (TGT request signature). If that fails, the KDC returns an error message of type KDC_ERR_INVALID_SIG.
0x41	KDC_ERR_KEY_TOO_WEAK	A higher encryption level is needed	If the clientPublicValue field is filled in, indicating that the client wishes to use Diffie-Hellman key agreement, then the KDC checks to see that the parameters satisfy its policy. If they do not (e.g., the prime size is insufficient for the expected encryption type), then the KDC sends back

0x42	KRB_AP_ERR_USER_TO_USER_REQUIRED	User-to-user authorization is required	an error message of type KDC_ERR_KEY_TOO_WEAK.
0x43	KRB_AP_ERR_NO_TGT	No TGT was presented or available	In the case that the client application doesn't know that a service requires user-to-user authentication, and requests and receives a conventional KRB_AP REP, the client will send the KRB_AP REP request, and the server will respond with a KRB_ERROR token as described in RFC1964 , with a msg-type of KRB_AP_ERR_USER_TO_USER_REQUIRED.
0x44	KDC_ERR_WRONG_REALM	Incorrect domain or principal	In user-to-user authentication if the service does not possess a ticket granting ticket, it should return the error KRB_AP_ERR_NO_TGT.
			Although this error rarely occurs, it occurs when a client presents a cross-realm TGT to a realm other than the one specified in the TGT. Typically, this results from incorrectly configured DNS.

Table 3. TGT/TGS issue error codes.

- **Ticket Encryption Type** [Type = HexInt32]: the cryptographic suite that was used for issued TGT.

Type	Type Name	Description
0x1	DES-CBC-CRC	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x3	DES-CBC-MD5	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x11	AES128-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x12	AES256-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x17	RC4-HMAC	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0x18	RC4-HMAC-EXP	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0xFFFFFFFF or 0xffffffff	-	This type shows in Audit Failure events.

Table 4. Kerberos encryption types

- **Pre-Authentication Type** [Type = UnicodeString]: the code number of [pre-Authentication](#) type which was used in TGT request.

Type	Type Name	Description
0	-	Logon without Pre-Authentication.
2	PA-ENC-TIMESTAMP	This is a normal type for standard password authentication.
11	PA-ETYPE-INFO	The ETYPE-INFO pre-authentication type is sent by the KDC in a KRB-ERROR indicating a requirement for additional pre-authentication. It is usually used to notify a client of which key to use for the encryption of an encrypted timestamp for the purposes of sending a PA-ENC-TIMESTAMP pre-authentication value. Never saw this Pre-Authentication Type in Microsoft Active Directory environment.
15	PA-PK-AS-REP_OLD	Used for Smart Card logon authentication.
17	PA-PK-AS-REP	This type should also be used for Smart Card authentication, but in certain Active Directory environments, it is never seen.
19	PA-ETYPE-INFO2	The ETYPE-INFO2 pre-authentication type is sent by the KDC in a KRB-ERROR indicating a requirement for additional pre-authentication. It is usually used to notify a client of which key to use for the encryption of an encrypted timestamp for the purposes of sending a PA-ENC-TIMESTAMP pre-authentication value.

		Never saw this Pre-Authentication Type in Microsoft Active Directory environment.
20	PA-SVR-REFERRAL-INFO	Used in KDC Referrals tickets.
138	PA-ENCRYPTED-CHALLENGE	Logon using Kerberos Armoring (FAST). Supported starting from Windows Server 2012 domain controllers and Windows 8 clients.
-		This type shows in Audit Failure events.

Table 5. Kerberos Pre-Authentication types.

Certificate Information:

Certificate Issuer Name [Type = UnicodeString]: the name of the Certification Authority that issued the smart card certificate. Populated in **Issued by** field in certificate.

Certificate Serial Number [Type = UnicodeString]: smart card certificate's serial number. Can be found in **Serial number** field in the certificate.

Certificate Thumbprint [Type = UnicodeString]: smart card certificate's thumbprint. Can be found in **Thumbprint** field in the certificate.

Security Monitoring Recommendations:

For 4768(S, F): A Kerberos authentication ticket (TGT) was requested.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ User ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ User ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ User ID ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ User ID ” for accounts that are outside the whitelist.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Supplied Realm Name ” corresponding to another domain or “external” location.
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ User ID ” for names that don’t comply with naming conventions.

- You can track all [4768](#) events where the **Client Address** is not from your internal IP range or not from private IP ranges.
- If you know that **Account Name** should be used only from known list of IP addresses, track all **Client Address** values for this **Account Name** in [4768](#) events. If **Client Address** is not from the whitelist, generate the alert.
- All **Client Address** = ::1 means local authentication. If you know the list of accounts which should log on to the domain controllers, then you need to monitor for all possible violations, where **Client Address** = ::1 and **Account Name** is not allowed to log on to any domain controller.
- All [4768](#) events with **Client Port** field value > 0 and < 1024 should be examined, because a well-known port was used for outbound connection.
- Also consider monitoring the fields shown in the following table, to discover the issues listed:

Field	Issue to discover
Certificate Issuer Name	Certification authority name is not from your PKI infrastructure.
Certificate Issuer Name	Certification authority name is not authorized to issue smart card authentication certificates.
Pre-Authentication Type	Value is 0 , which means that pre-authentication was not used. All accounts should use Pre-Authentication, except accounts configured with "Do not require Kerberos preauthentication," which is a security risk. For more information, see Table 5. Kerberos Pre-Authentication types .
Pre-Authentication Type	Value is not 15 when account must use a smart card for authentication. For more information, see Table 5. Kerberos Pre-Authentication types .
Pre-Authentication Type	Value is not 2 when only standard password authentication is in use in the organization. For more information, see Table 5. Kerberos Pre-Authentication types .
Pre-Authentication Type	Value is not 138 when Kerberos Armoring is enabled for all Kerberos communications in the organization. For more information, see Table 5. Kerberos Pre-Authentication types .
Ticket Encryption Type	Value is 0x1 or 0x3 , which means the DES algorithm was used. DES should not be in use, because of low security and known vulnerabilities. It is disabled by default starting from Windows 7 and Windows Server 2008 R2. For more information, see Table 4. Kerberos encryption types .
Ticket Encryption Type	Starting with Windows Vista and Windows Server 2008, monitor for values other than 0x11 and 0x12 . These are the expected values, starting with these operating systems, and represent AES-family algorithms. For more information, see Table 4. Kerberos encryption types .
Result Code	0x6 (The username doesn't exist), if you see, for example N events in last N minutes. This can be an indicator of account enumeration attack, especially for highly critical accounts.
Result Code	0x7 (Server not found in Kerberos database). This error can occur if the domain controller cannot find the server's name in Active Directory.
Result Code	0x8 (Multiple principal entries in KDC database). This will help you to find duplicate SPNs faster.
Result Code	0x9 (The client or server has a null key (master key)). This error can help you to identify problems with Kerberos authentication faster.
Result Code	0xA (Ticket (TGT) not eligible for postdating). Microsoft systems should not request postdated tickets. These events could help identify anomaly activity.
Result Code	0xC (Requested start time is later than end time), if you see, for example N events in last N minutes. This can be an indicator of an account compromise attempt, especially for highly critical accounts.
Result Code	0xE (KDC has no support for encryption type). In general, this error occurs when the KDC or a client receives a packet that it cannot decrypt. Monitor for these events because this should not happen in a standard Active Directory environment.
Result Code	0xF (KDC has no support for checksum type). Monitor for these events because this should not happen in a standard Active Directory environment.
Result Code	0x12 (Client's credentials have been revoked), if you see, for example N events in last N minutes. This can be an indicator of anomaly activity or brute-force attack, especially for highly critical accounts.
Result Code	0x1F (Integrity check on decrypted field failed). The authenticator was encrypted with something other than the session key. The result is that the

KDC cannot decrypt the TGT. The modification of the message could be the result of an attack or it could be because of network noise.

Result Code	0x22 (The request is a replay). This error indicates that a specific authenticator showed up twice—the KDC has detected that this session ticket duplicates one that it has already received. It could be a sign of attack attempt.
Result Code	0x29 (Message stream modified and checksum didn't match). The authentication data was encrypted with the wrong key for the intended server. The authentication data was modified in transit by a hardware or software error, or by an attacker. Monitor for these events because this should not happen in a standard Active Directory environment.
Result Code	0x3C (Generic error). This error can help you more quickly identify problems with Kerberos authentication.
Result Code	0x3E (The client trust failed or is not implemented). This error helps you identify logon attempts with revoked certificates and the situations when the root Certification Authority that issued the smart card certificate (through a chain) is not trusted by a domain controller.
Result Code	0x3F, 0x40, 0x41 errors. These errors can help you more quickly identify smart-card related problems with Kerberos authentication.

Event Properties - Event 4771, Microsoft Windows security auditing.

General Details

Kerberos pre-authentication failed.

Account Information:

- Security ID: CONTOSO\dadmin
- Account Name: dadmin

Service Information:

- Service Name: krbtgt/CONTOSO.LOCAL

Network Information:

- Client Address: ::ffff:10.0.0.12

Ticket Options: 0x40810010
Failure Code: 0x10
Pre-Authentication Type: 15

Certificate Information:

- Certificate Issuer Name:
- Certificate Serial Number:
- Certificate Thumbprint:

Certificate information is only provided if a certificate was used for pre-authentication.

Pre-authentication types, ticket options and failure codes are defined in RFC 4120.

If the ticket was malformed or damaged during transit and could not be decrypted, then many fields in this event might not be present.

Log Name: Security
Source: Microsoft Windows security
Event ID: 4771
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 8/7/2015 11:10:21 AM
Task Category: Kerberos Authentication Service
Keywords: Audit Failure
Computer: DC01.contoso.local

Copy **Close**

4771(F): Kerberos pre-authentication failed.

Event Description:

This event generates every time the Key Distribution Center fails to issue a Kerberos Ticket Granting Ticket (TGT). This can occur when a domain controller doesn't have a certificate installed for smart card authentication (for example, with a "Domain Controller" or "Domain Controller Authentication" template), the user's password has expired, or the wrong password was provided.

This event generates only on domain controllers.

This event is not generated if "Do not require Kerberos preauthentication" option is set for the account.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4771</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14339</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-08-07T18:10:21.495462300Z" />
<EventRecordID>166708</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1084" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="ServiceName">krbtgt/CONTOSO.LOCAL</Data>
<Data Name="TicketOptions">0x40810010</Data>
<Data Name="Status">0x10</Data>
<Data Name="PreAuthType">15</Data>
<Data Name="IpAddress">::ffff:10.0.0.12</Data>
<Data Name="IpPort">49254</Data>
<Data Name="CertIssuerName" />
<Data Name="CertSerialNumber" />
<Data Name="CertThumbprint" />
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Account Information:

- **Security ID** [Type = SID]: SID of account object for which (TGT) ticket was requested. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
For example: CONTOSO\dadmin or CONTOSO\WIN81\$.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name:** [Type = UnicodeString]: the name of account, for which (TGT) ticket was requested. Computer account name ends with \$ character.
 - User account example: dadmin
 - Computer account example: WIN81\$

Service Information:

- **Service Name** [Type = UnicodeString]: the name of the service in the Kerberos Realm to which TGT request was sent. Typically has one of the following formats:
 - krbtgt/DOMAIN_NETBIOS_NAME. Example: krbtgt/CONTOSO
 - krbtgt/DOMAIN_FULL_NAME. Example: krbtgt/CONTOSO.LOCAL

Network Information:

- **Client Address** [Type = UnicodeString]: IP address of the computer from which the TGT request was received. Formats vary, and include the following:

- IPv6 or IPv4 address.
- ::ffff:IPv4_address.
- ::1 - localhost.
- **Client Port** [Type = UnicodeString]: source port number of client network connection (TGT request connection).
 - 0 for local (localhost) requests.

Additional Information:

- **Ticket Options:** [Type = HexInt32]: this is a set of different Ticket Flags in hexadecimal format.

Example:

- Ticket Options: 0x40810010
- Binary view: 010000001000000100000000000010000
- Using **MSB 0** bit numbering we have bit 1, 8, 15 and 27 set = Forwardable, Renewable, Canonicalize, Renewable-ok.

In the table below “**MSB 0**” bit numbering is used, because RFC documents use this style. In “**MSB 0**” style bit numbering begins from left.

	0		7					
	1	0	0	1	0	1	1	0

The most common values:

- 0x40810010 - Forwardable, Renewable, Canonicalize, Renewable-ok
- 0x40810000 - Forwardable, Renewable, Canonicalize
- 0x60810010 - Forwardable, Forwarded, Renewable, Canonicalize, Renewable-ok

Bit	Flag Name	Description
0	Reserved	-
1	Forwardable	(TGT only). Tells the ticket-granting service that it can issue a new TGT—based on the presented TGT—with a different network address based on the presented TGT.
2	Forwarded	Indicates either that a TGT has been forwarded or that a ticket was issued from a forwarded TGT.
3	Proxyable	(TGT only). Tells the ticket-granting service that it can issue tickets with a network address that differs from the one in the TGT.
4	Proxy	Indicates that the network address in the ticket is different from the one in the TGT used to obtain the ticket.
5	Allow-postdate	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
6	Postdated	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
7	Invalid	This flag indicates that a ticket is invalid, and it must be validated by the KDC before use. Application servers must reject tickets which have this flag set.
8	Renewable	Used in combination with the End Time and Renew Till fields to cause tickets with long life spans to be renewed at the KDC periodically.
9	Initial	Indicates that a ticket was issued using the authentication service (AS) exchange and not issued based on a TGT.
10	Pre-authent	Indicates that the client was authenticated by the KDC before a ticket was issued. This flag usually indicates the presence of an authenticator in the ticket. It can also flag the presence of credentials taken from a smart card logon.
11	Opt-hardware-auth	This flag was originally intended to indicate that hardware-supported authentication was used during pre-authentication.

		This flag is no longer recommended in the Kerberos V5 protocol. KDCs MUST NOT issue a ticket with this flag set. KDCs SHOULD NOT preserve this flag if it is set by another KDC.
12	Transited-policy-checked	KILE MUST NOT check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag.
13	Ok-as-delegate	The KDC MUST set the OK-AS-DELEGATE flag if the service account is trusted for delegation.
14	Request-anonymous	KILE not use this flag.
15	Name-canonicalize	In order to request referrals the Kerberos client MUST explicitly request the "canonicalize" KDC option for the AS-REQ or TGS-REQ.
16-25	Unused	-
26	Disable-transited-check	By default the KDC will check the transited field of a TGT against the policy of the local realm before it will issue derivative tickets based on the TGT. If this flag is set in the request, checking of the transited field is disabled. Tickets issued without the performance of this check will be noted by the reset (0) value of the TRANSITED-POLICY-CHECKED flag, indicating to the application server that the transited field must be checked locally. KDCs are encouraged but not required to honor the DISABLE-TRANSITED-CHECK option. Should not be in use, because Transited-policy-checked flag is not supported by KILE.
27	Renewable-ok	The RENEWABLE-OK option indicates that a renewable ticket will be acceptable if a ticket with the requested life cannot otherwise be provided, in which case a renewable ticket may be issued with a renew-till equal to the requested end time. The value of the renew-till field may still be limited by local limits, or limits selected by the individual principal or server.
28	Enc-tkt-in-skey	No information.
29	Unused	-
30	Renew	The RENEW option indicates that the present request is for a renewal. The ticket provided is encrypted in the secret key for the server on which it is valid. This option will only be honored if the ticket to be renewed has its RENEWABLE flag set and if the time in its renew-till field has not passed. The ticket to be renewed is passed in the padata field as part of the authentication header.
31	Validate	This option is used only by the ticket-granting service. The VALIDATE option indicates that the request is to validate a postdated ticket. Should not be in use, because postdated tickets are not supported by KILE.

Table 6. Kerberos ticket flags.

- **Failure Code** [Type = HexInt32]: hexadecimal failure code of failed TGT issue operation. The table below contains the list of the most common error codes for this event:

Code	Code Name	Description	Possible causes
0x10	KDC_ERR_PADATA_TYPE_NOSUPP	KDC has no support for PADATA type (pre-authentication data)	Smart card logon is being attempted and the proper certificate cannot be located. This can happen because the wrong certification authority (CA) is being queried or the proper CA cannot be contacted in order to get Domain Controller or Domain Controller Authentication certificates for the domain controller. It can also happen when a domain controller doesn't have a certificate installed for smart cards (Domain Controller or Domain Controller Authentication templates).

0x17	KDC_ERR_KEY_EXPIRED	Password has expired—change password to reset	The user's password has expired.
0x18	KDC_ERR_PREAUTH_FAILED	Pre-authentication information was invalid	The wrong password was provided.

- **Pre-Authentication Type** [Type = UnicodeString]: the code of [pre-Authentication](#) type which was used in TGT request.

Type	Type Name	Description
0	-	Logon without Pre-Authentication.
2	PA-ENC-TIMESTAMP	This is a normal type for standard password authentication.
11	PA-ETYPE-INFO	The ETYPE-INFO pre-authentication type is sent by the KDC in a KRB-ERROR indicating a requirement for additional pre-authentication. It is usually used to notify a client of which key to use for the encryption of an encrypted timestamp for the purposes of sending a PA-ENC-TIMESTAMP pre-authentication value. Never saw this Pre-Authentication Type in Microsoft Active Directory environment.
15	PA-PK-AS-REP_OLD	Used for Smart Card logon authentication.
17	PA-PK-AS-REP	This type should also be used for Smart Card authentication, but in certain Active Directory environments, it is never seen.
19	PA-ETYPE-INFO2	The ETYPE-INFO2 pre-authentication type is sent by the KDC in a KRB-ERROR indicating a requirement for additional pre-authentication. It is usually used to notify a client of which key to use for the encryption of an encrypted timestamp for the purposes of sending a PA-ENC-TIMESTAMP pre-authentication value. Never saw this Pre-Authentication Type in Microsoft Active Directory environment.
20	PA-SVR-REFERRAL-INFO	Used in KDC Referrals tickets.
138	PA-ENCRYPTED-CHALLENGE	Logon using Kerberos Armoring (FAST). Supported starting from Windows Server 2012 domain controllers and Windows 8 clients.
-		This type shows in Audit Failure events.

Certificate Information:

Certificate Issuer Name [Type = UnicodeString]: the name of Certification Authority which issued smart card certificate. Populated in **Issued by** field in certificate. Always empty for [4771](#) events.

Certificate Serial Number [Type = UnicodeString]: smart card certificate's serial number. Can be found in **Serial number** field in the certificate. Always empty for [4771](#) events.

Certificate Thumbprint [Type = UnicodeString]: smart card certificate's thumbprint. Can be found in **Thumbprint** field in the certificate. Always empty for [4771](#) events.

Security Monitoring Recommendations:

For 4771(F): Kerberos pre-authentication failed.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Security ID ” that corresponds to the high-value account or accounts.

Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.

When you monitor for anomalies or malicious actions, use the “**Security ID**” (with other information) to monitor how or when a particular account is being used.

Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.

Monitor this event with the “**Security ID**” that corresponds to the accounts that should never be used.

Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.

If this event corresponds to a “whitelist-only” action, review the “**Security ID**” for accounts that are outside the whitelist.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Account Name**” for names that don’t comply with naming conventions.

- You can track all [4771](#) events where the **Client Address** is not from your internal IP range or not from private IP ranges.
- If you know that **Account Name** should be used only from known list of IP addresses, track all **Client Address** values for this **Account Name** in [4771](#) events. If **Client Address** is not from the whitelist, generate the alert.
- All **Client Address** = ::1 means local authentication. If you know the list of accounts which should log on to the domain controllers, then you need to monitor for all possible violations, where **Client Address** = ::1 and **Account Name** is not allowed to log on to any domain controller.
- All [4771](#) events with **Client Port** field value > 0 and < 1024 should be examined, because a well-known port was used for outbound connection.
- Also monitor the fields shown in the following table, to discover the issues listed:

Field	Issue to discover
Pre-Authentication Type	Value is not 15 when account must use a smart card for authentication. For more information, see Table 5. Kerberos Pre-Authentication types .
Pre-Authentication Type	Value is not 2 when only standard password authentication is in use in the organization. For more information, see Table 5. Kerberos Pre-Authentication types .
Pre-Authentication Type	Value is not 138 when Kerberos Armoring is enabled for all Kerberos communications in the organization. For more information, see Table 5. Kerberos Pre-Authentication types .
Result Code	0x10 (KDC has no support for PADATA type (pre-authentication data)). This error can help you to more quickly identify smart-card related problems with Kerberos authentication.
Result Code	0x18 ((Pre-authentication information was invalid), if you see, for example N events in last N minutes. This can be an indicator of brute-force attack on the account password, especially for highly critical accounts.

4772(F): A Kerberos authentication ticket request failed.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system. [4768](#) failure event is generated instead.

Audit Kerberos Service Ticket Operations

Audit Kerberos Service Ticket Operations determines whether the operating system generates security audit events for Kerberos service ticket requests.

Events are generated every time Kerberos is used to authenticate a user who wants to access a protected network resource. Kerberos service ticket operation audit events can be used to track user activity.

Event volume: Very High on Kerberos Key Distribution Center servers.

This subcategory contains events about issued TGSs and failed TGS requests.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	Yes	Yes	Yes	<p>Expected volume is very high on domain controllers.</p> <p>IF - We recommend Success auditing, because you will see all Kerberos Service Ticket requests (TGS requests), which are part of service use and access requests by specific accounts. Also, you can see the IP address from which this account requested TGS, when TGS was requested, which encryption type was used, and so on. For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections.</p> <p>We recommend Failure auditing, because you will see all failed requests and be able to investigate the reason for failure. You will also be able to detect Kerberos issues or possible attack attempts.</p>
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Events List:

- [4769](#)(S, F): A Kerberos service ticket was requested.
- [4770](#)(S): A Kerberos service ticket was renewed.
- [4773](#)(F): A Kerberos service ticket request failed.

4769(S, F): A Kerberos service ticket was requested.

Event Properties - Event 4769, Microsoft Windows security auditing.

General **Details**

A Kerberos service ticket was requested.

Account Information:
 Account Name: dadmin@CONTOSO.LOCAL
 Account Domain: CONTOSO.LOCAL

Service Name: WIN2008R2\$
 Service ID: CONTOSO\WIN2008R2\$

Network Information:
 Client Address: ::ffff:10.0.0.12
 Client Port: 49272

Additional Information:
 Ticket Options: 0x40810000
 Ticket Encryption Type: 0x12
 Failure Code: 0x0
 Transited Services: -

This event is generated every time access is requested to a resource such as a computer or a Windows service. The service name indicates the resource to which access was requested.

This event can be correlated with Windows logon events by comparing the Logon GUID fields in each event. The logon event occurs on the machine that was accessed, which is often a different machine than the domain controller which issued the service ticket.

Ticket options, encryption types, and failure codes are defined in RFC 4120.

Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4769
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy **Close**

Event Description:

This event generates every time Key Distribution Center gets a Kerberos Ticket Granting Service (TGS) ticket request. This event generates only on domain controllers.
 If TGS issue fails then you will see Failure event with **Failure Code** field not equal to “0x0”. You will typically see many Failure events with **Failure Code “0x20”**, which simply means that a TGS ticket has expired. These are informational messages and have little to no security relevance.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4769</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14337</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-07T18:13:46.043256100Z" />
<EventRecordID>166746</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1496" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">dadmin@CONTOSO.LOCAL</Data>
<Data Name="TargetDomainName">CONTOSO.LOCAL</Data>

<Data Name="ServiceName">WIN2008R2$</Data>
<Data Name="ServiceSid">S-1-5-21-3457937927-2839227994-823803824-2102</Data>
<Data Name="TicketOptions">0x40810000</Data>
<Data Name="TicketEncryptionType">0x12</Data>
<Data Name="IpAddress">::ffff:10.0.0.12</Data>
<Data Name="IpPort">49272</Data>
<Data Name="Status">0x0</Data>
```

```
<Data Name="LogonGuid">{F85C455E-C66E-205C-6B39-F6C60A7FE453}</Data>
<Data Name="TransmittedServices">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Account Information:

- **Account Name** [Type = UnicodeString]: the User Principal Name (UPN) of the account that requested the ticket. Computer account name ends with \$ character in UPN. This field typically has the following value format: user_account_name@FULL_DOMAIN_NAME.
 - User account example: dadmin@CONTOSO.LOCAL
 - Computer account example: WIN81\$@CONTOSO.LOCAL

This parameter in this event is optional and can be empty in some cases.

- **Account Domain** [Type = UnicodeString]: the name of the Kerberos Realm that **Account Name** belongs to. This can appear in a variety of formats, including the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

This parameter in this event is optional and can be empty in some cases.

- **Logon GUID** [Type = GUID]: a GUID that can help you correlate this event (on a domain controller) with other events (on the target computer for which the TGS was issued) that can contain the same **Logon GUID**. These events are “[4624](#): An account was successfully logged on”, “[4648\(S\)](#): A logon was attempted using explicit credentials” and “[4964\(S\)](#): Special groups have been assigned to a new logon.”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Service Information:

- **Service Name** [Type = UnicodeString]: the name of the account or computer for which the TGS ticket was requested.
 - This parameter in this event is optional and can be empty in some cases.
- **Service ID** [Type = SID]: SID of the account or computer object for which the TGS ticket was requested. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
 - **NULL SID** – this value shows in Failure events.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

Network Information:

- **Client Address** [Type = UnicodeString]: IP address of the computer from which the TGS request was received. Formats vary, and include the following:

- IPv6 or IPv4 address.
 - ::ffff:IPv4_address.
 - ::1 - localhost.
- **Client Port** [Type = UnicodeString]: source port number of client network connection (TGS request connection).
 - 0 for local (localhost) requests.

Additional information:

- **Ticket Options:** [Type = HexInt32]: this is a set of different Ticket Flags in hexadecimal format.

Example:

- Ticket Options: 0x40810010
- Binary view: 010000001000000100000000000010000
- Using **MSB 0** bit numbering we have bit 1, 8, 15 and 27 set = Forwardable, Renewable, Canonicalize, Renewable-ok.

In the table below “**MSB 0**” bit numbering is used, because RFC documents use this style. In “**MSB 0**” style bit numbering begins from left.

	0		7					
	1	0	0	1	0	1	1	0

The most common values:

- 0x40810010 - Forwardable, Renewable, Canonicalize, Renewable-ok
- 0x40810000 - Forwardable, Renewable, Canonicalize
- 0x60810010 - Forwardable, Forwarded, Renewable, Canonicalize, Renewable-ok

Bit	Flag Name	Description
0	Reserved	-
1	Forwardable	(TGT only). Tells the ticket-granting service that it can issue a new TGT—based on the presented TGT—with a different network address based on the presented TGT.
2	Forwarded	Indicates either that a TGT has been forwarded or that a ticket was issued from a forwarded TGT.
3	Proxyable	(TGT only). Tells the ticket-granting service that it can issue tickets with a network address that differs from the one in the TGT.
4	Proxy	Indicates that the network address in the ticket is different from the one in the TGT used to obtain the ticket.
5	Allow-postdate	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
6	Postdated	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
7	Invalid	This flag indicates that a ticket is invalid, and it must be validated by the KDC before use. Application servers must reject tickets which have this flag set.
8	Renewable	Used in combination with the End Time and Renew Till fields to cause tickets with long life spans to be renewed at the KDC periodically.
9	Initial	Indicates that a ticket was issued using the authentication service (AS) exchange and not issued based on a TGT.
10	Pre-authent	Indicates that the client was authenticated by the KDC before a ticket was issued. This flag usually indicates the presence of an authenticator in the ticket. It can also flag the presence of credentials taken from a smart card logon.
11	Opt-hardware-auth	This flag was originally intended to indicate that hardware-supported authentication was used during pre-authentication.

		This flag is no longer recommended in the Kerberos V5 protocol. KDCs MUST NOT issue a ticket with this flag set. KDCs SHOULD NOT preserve this flag if it is set by another KDC.
12	Transited-policy-checked	KILE MUST NOT check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag.
13	Ok-as-delegate	The KDC MUST set the OK-AS-DELEGATE flag if the service account is trusted for delegation.
14	Request-anonymous	KILE not use this flag.
15	Name-canonicalize	In order to request referrals the Kerberos client MUST explicitly request the "canonicalize" KDC option for the AS-REQ or TGS-REQ.
16-25	Unused	-
26	Disable-transited-check	By default the KDC will check the transited field of a TGT against the policy of the local realm before it will issue derivative tickets based on the TGT. If this flag is set in the request, checking of the transited field is disabled. Tickets issued without the performance of this check will be noted by the reset (0) value of the TRANSITED-POLICY-CHECKED flag, indicating to the application server that the transited field must be checked locally. KDCs are encouraged but not required to honor the DISABLE-TRANSITED-CHECK option. Should not be in use, because Transited-policy-checked flag is not supported by KILE.
27	Renewable-ok	The RENEWABLE-OK option indicates that a renewable ticket will be acceptable if a ticket with the requested life cannot otherwise be provided, in which case a renewable ticket may be issued with a renew-till equal to the requested end time. The value of the renew-till field may still be limited by local limits, or limits selected by the individual principal or server.
28	Enc-tkt-in-skey	No information.
29	Unused	-
30	Renew	The RENEW option indicates that the present request is for a renewal. The ticket provided is encrypted in the secret key for the server on which it is valid. This option will only be honored if the ticket to be renewed has its RENEWABLE flag set and if the time in its renew-till field has not passed. The ticket to be renewed is passed in the padata field as part of the authentication header.
31	Validate	This option is used only by the ticket-granting service. The VALIDATE option indicates that the request is to validate a postdated ticket. Should not be in use, because postdated tickets are not supported by KILE.

- **Ticket Encryption Type:** [Type = HexInt32]: the cryptographic suite that was used for issued TGS.

Type	Type Name	Description
0x1	DES-CBC-CRC	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x3	DES-CBC-MD5	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x11	AES128-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x12	AES256-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x17	RC4-HMAC	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0x18	RC4-HMAC-EXP	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0xFFFFFFFF or 0xffffffff	-	This type shows in Audit Failure events.

- **Failure Code [Type = HexInt32]:** hexadecimal result code of TGS issue operation. The table below contains the list of the most common error codes for this event:

Code	Code Name	Description	Possible causes
0x0	KDC_ERR_NONE	No error	No errors were found.
0x1	KDC_ERR_NAME_EXP	Client's entry in KDC database has expired	No information.
0x2	KDC_ERR_SERVICE_EXP	Server's entry in KDC database has expired	No information.
0x3	KDC_ERR_BAD_PVNO	Requested Kerberos version number not supported	No information.
0x4	KDC_ERR_C_OLD_MAST_KVNO	Client's key encrypted in old master key	No information.
0x5	KDC_ERR_S_OLD_MAST_KVNO	Server's key encrypted in old master key	No information.
0x6	KDC_ERR_C_PRINCIPAL_UNKNOWN	Client not found in Kerberos database	The username doesn't exist.
0x7	KDC_ERR_S_PRINCIPAL_UNKNOWN	Server not found in Kerberos database	This error can occur if the domain controller cannot find the server's name in Active Directory. This error is similar to KDC_ERR_C_PRINCIPAL_UNKNOWN except that it occurs when the server name cannot be found.
0x8	KDC_ERR_PRINCIPAL_NOT_UNIQUE	Multiple principal entries in KDC database	This error occurs if duplicate principal names exist. Unique principal names are crucial for ensuring mutual authentication. Thus, duplicate principal names are strictly forbidden, even across multiple realms. Without unique principal names, the client has no way of ensuring that the server it is communicating with is the correct one.
0x9	KDC_ERR_NULL_KEY	The client or server has a null key (master key)	No master key was found for client or server. Usually it means that administrator should reset the password on the account.
0xA	KDC_ERR_CANNOT_POSTDATE	Ticket (TGT) not eligible for postdating	This error can occur if a client requests postdating of a Kerberos ticket. Postdating is the act of requesting that a ticket's start time be set into the future. It also can occur if there is a time difference between the client and the KDC.
0xB	KDC_ERR_NEVER_VALID	Requested start time is later than end time	There is a time difference between the KDC and the client.
0xC	KDC_ERR_POLICY	Requested start time is later than end time	This error is usually the result of logon restrictions in place on a user's account. For example workstation restriction, smart card authentication requirement or logon time restriction.
0xD	KDC_ERR_BADOPTION	KDC cannot accommodate requested option	Impending expiration of a TGT. The SPN to which the client is attempting to delegate credentials is not in its Allowed-to-delegate-to list
0xE	KDC_ERRETYPE_NOTSUPP	KDC has no support for encryption type	In general, this error occurs when the KDC or a client receives a packet that it cannot decrypt.
0xF	KDC_ERR_SUMTYPE_NOSUPP	KDC has no support for checksum type	The KDC, server, or client receives a packet for which it does not have a key of the appropriate encryption type. The result is that the computer

			is unable to decrypt the ticket.
0x10	KDC_ERR_PADATA_TYPE_NOSUPP	KDC has no support for PADATA type (pre-authentication data)	Smart card logon is being attempted and the proper certificate cannot be located. This can happen because the wrong certification authority (CA) is being queried or the proper CA cannot be contacted. It can also happen when a domain controller doesn't have a certificate installed for smart cards (Domain Controller or Domain Controller Authentication templates). This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x11	KDC_ERR_TRTYPE_NO_SUPP	KDC has no support for transited type	No information.
0x12	KDC_ERR_CLIENT_REVOKED	Client's credentials have been revoked	This might be because of an explicit disabling or because of other restrictions in place on the account. For example: account disabled, expired, or locked out.
0x13	KDC_ERR_SERVICE_REVOKED	Credentials for server have been revoked	No information.
0x14	KDC_ERR_TGT_REVOKED	TGT has been revoked	Since the remote KDC may change its PKCROSS key while there are PKCROSS tickets still active, it SHOULD cache the old PKCROSS keys until the last issued PKCROSS ticket expires. Otherwise, the remote KDC will respond to a client with a KRB-ERROR message of type KDC_ERR_TGT_REVOKED. See RFC1510 for more details.
0x15	KDC_ERR_CLIENT_NOTYET	Client not yet valid—try again later	No information.
0x16	KDC_ERR_SERVICE_NOTYET	Server not yet valid—try again later	No information.
0x17	KDC_ERR_KEY_EXPIRED	Password has expired—change password to reset	The user's password has expired. This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x18	KDC_ERR_PREAUTH_FAILED	Pre-authentication information was invalid	The wrong password was provided. This error code cannot occur in event " 4768 . A Kerberos authentication ticket (TGT) was requested". It occurs in " 4771 . Kerberos pre-authentication failed" event.
0x19	KDC_ERR_PREAUTH_REQUIRED	Additional pre-authentication required	This error often occurs in UNIX interoperability scenarios. MIT-Kerberos clients do not request pre-authentication when they send a KRB_AS_REQ message. If pre-authentication is required (the default), Windows systems will send this error. Most MIT-Kerberos clients will respond to this error by giving the pre-authentication; in which case the error can be ignored, but some clients might not respond in this way.
0x1A	KDC_ERR_SERVER_NOMATCH	KDC does not know about the requested	No information.

server			
0x1B	KDC_ERR_SVC_UNAVAILABLE	KDC is unavailable	No information.
0x1F	KRB_AP_ERR_BAD_INTEGRITY	Integrity check on decrypted field failed	The authenticator was encrypted with something other than the session key. The result is that the client cannot decrypt the resulting message. The modification of the message could be the result of an attack or it could be because of network noise.
0x20	KRB_AP_ERR_TKT_EXPIRED	The ticket has expired	The smaller the value for the “Maximum lifetime for user ticket” Kerberos policy setting, the more likely it is that this error will occur. Because ticket renewal is automatic, you should not have to do anything if you get this message.
0x21	KRB_AP_ERR_TKT_NYV	The ticket is not yet valid	The ticket presented to the server is not yet valid (in relationship to the server time). The most probable cause is that the clocks on the KDC and the client are not synchronized. If cross-realm Kerberos authentication is being attempted, then you should verify time synchronization between the KDC in the target realm and the KDC in the client realm, as well.
0x22	KRB_AP_ERR_REPEAT	The request is a replay	This error indicates that a specific authenticator showed up twice — the KDC has detected that this session ticket duplicates one that it has already received.
0x23	KRB_AP_ERR_NOT_US	The ticket is not for us	The server has received a ticket that was meant for a different realm.
0x24	KRB_AP_ERR_BADMATCH	The ticket and authenticator do not match	The KRB_TGS_REQ is being sent to the wrong KDC. There is an account mismatch during protocol transition.
0x25	KRB_AP_ERR_SKEW	The clock skew is too great	This error is logged if a client computer sends a timestamp whose value differs from that of the server’s timestamp by more than the number of minutes found in the “Maximum tolerance for computer clock synchronization” setting in Kerberos policy.
0x26	KRB_AP_ERR_BADADDR	Network address in network layer header doesn't match address inside ticket	Session tickets MAY include the addresses from which they are valid. This error can occur if the address of the computer sending the ticket is different from the valid address in the ticket. A possible cause of this could be an Internet Protocol (IP) address change. Another possible cause is when a ticket is passed through a proxy server or NAT. The client is unaware of the address scheme used by the proxy server, so unless the program caused the client to request a proxy server ticket with the proxy server's source address, the ticket could be invalid.
0x27	KRB_AP_ERR_BADVERSION	Protocol version numbers don't match (PVNO)	When an application receives a KRB_SAFE message, it verifies it. If any error occurs, an error code is reported for use by the application. The message is first checked by verifying that the protocol version and type fields match the current version and KRB_SAFE, respectively. A

			mismatch generates a KRB_AP_ERR_BADVERSION. See RFC4120 for more details.
0x28	KRB_AP_ERR_MSG_TYPE	Message type is unsupported	This message is generated when target server finds that message format is wrong. This applies to KRB_AP_REQ, KRB_SAFE, KRB_PRIV and KRB_CRED messages. This error also generated if use of UDP protocol is being attempted with User-to-User authentication.
0x29	KRB_AP_ERR_MODIFIED	Message stream modified and checksum didn't match	The authentication data was encrypted with the wrong key for the intended server. The authentication data was modified in transit by a hardware or software error, or by an attacker. The client sent the authentication data to the wrong server because incorrect DNS data caused the client to send the request to the wrong server. The client sent the authentication data to the wrong server because DNS data was out-of-date on the client.
0x2A	KRB_AP_ERR_BADORDER	Message out of order (possible tampering)	This event generates for KRB_SAFE and KRB_PRIV messages if an incorrect sequence number is included, or if a sequence number is expected but not present. See RFC4120 for more details.
0x2C	KRB_AP_ERR_BADKEYVER	Specified version of key is not available	This error might be generated on server side during receipt of invalid KRB_AP_REQ message. If the key version indicated by the Ticket in the KRB_AP_REQ is not one the server can use (e.g., it indicates an old key, and the server no longer possesses a copy of the old key), the KRB_AP_ERR_BADKEYVER error is returned.
0x2D	KRB_AP_ERR_NOKEY	Service key not available	This error might be generated on server side during receipt of invalid KRB_AP_REQ message. Because it is possible for the server to be registered in multiple realms, with different keys in each, the realm field in the unencrypted portion of the ticket in the KRB_AP_REQ is used to specify which secret key the server should use to decrypt that ticket. The KRB_AP_ERR_NOKEY error code is returned if the server doesn't have the proper key to decipher the ticket.
0x2E	KRB_AP_ERR_MUT_FAIL	Mutual authentication failed	No information.
0x2F	KRB_AP_ERR_BADDIRECTION	Incorrect message direction	No information.
0x30	KRB_AP_ERR_METHOD	Alternative authentication method required	According RFC4120 this error message is obsolete.
0x31	KRB_AP_ERR_BADSEQ	Incorrect sequence number in message	No information.
0x32	KRB_AP_ERR_INAPP_CKSUM	Inappropriate type of checksum in message (checksum may be unsupported)	When KDC receives KRB_TGS_REQ message it decrypts it, and after the user-supplied checksum in the Authenticator MUST be verified against the contents of the request, and the message MUST be rejected if the

			checksums do not match (with an error code of KRB_AP_ERR_MODIFIED) or if the checksum is not collision-proof (with an error code of KRB_AP_ERR_INAPP_CKSUM).
0x33	KRB_AP_PATH_NOT_ACCEPTED	Desired path is unreachable	No information.
0x34	KRB_ERR_RESPONSE_TOO_BIG	Too much data	The size of a ticket is too large to be transmitted reliably via UDP. In a Windows environment, this message is purely informational. A computer running a Windows operating system will automatically try TCP if UDP fails.
0x3C	KRB_ERR_GENERIC	Generic error	Group membership has overloaded the PAC. Multiple recent password changes have not propagated. Crypto subsystem error caused by running out of memory. SPN too long. SPN has too many parts.
0x3D	KRB_ERR_FIELD_TOOLONG	Field is too long for this implementation	Each request (KRB_KDC_REQ) and response (KRB_KDC REP or KRB_ERROR) sent over the TCP stream is preceded by the length of the request as 4 octets in network byte order. The high bit of the length is reserved for future expansion and MUST currently be set to zero. If a KDC that does not understand how to interpret a set high bit of the length encoding receives a request with the high order bit of the length set, it MUST return a KRB-ERROR message with the error KRB_ERR_FIELD_TOOLONG and MUST close the TCP stream.
0x3E	KDC_ERR_CLIENT_NOT_TRUSTED	The client trust failed or is not implemented	This typically happens when user's smart-card certificate is revoked or the root Certification Authority that issued the smart card certificate (in a chain) is not trusted by the domain controller.
0x3F	KDC_ERR_KDC_NOT_TRUSTED	The KDC server trust failed or could not be verified	The trustedCertifiers field contains a list of certification authorities trusted by the client, in the case that the client does not possess the KDC's public key certificate. If the KDC has no certificate signed by any of the trustedCertifiers, then it returns an error of type KDC_ERR_KDC_NOT_TRUSTED. See RFC1510 for more details.
0x40	KDC_ERR_INVALID_SIG	The signature is invalid	This error is related to PKINIT. If a PKI trust relationship exists, the KDC then verifies the client's signature on AuthPack (TGT request signature). If that fails, the KDC returns an error message of type KDC_ERR_INVALID_SIG.
0x41	KDC_ERR_KEY_TOO_WEAK	A higher encryption level is needed	If the clientPublicValue field is filled in, indicating that the client wishes to use Diffie-Hellman key agreement, then the KDC checks to see that the parameters satisfy its policy. If they do not (e.g., the prime size is insufficient for the expected encryption type), then the KDC sends back an error message of type KDC_ERR_KEY_TOO_WEAK.

0x42	KRB_AP_ERR_USER_TO_USER_REQUR ED	User-to-user authorization is required	In the case that the client application doesn't know that a service requires user-to-user authentication, and requests and receives a conventional KRB_AP REP, the client will send the KRB_AP REP request, and the server will respond with a KRB_ERROR token as described in RFC1964 , with a msg-type of KRB_AP_ERR_USER_TO_USER_REQUIRED.
0x43	KRB_AP_ERR_NO_TGT	No TGT was presented or available	In user-to-user authentication if the service does not possess a ticket granting ticket, it should return the error KRB_AP_ERR_NO_TGT.
0x44	KDC_ERR_WRONG_REALM	Incorrect domain or principal	Although this error rarely occurs, it occurs when a client presents a cross-realm TGT to a realm other than the one specified in the TGT. Typically, this results from incorrectly configured DNS.

- Transited Services [Type = UnicodeString]: this field contains list of SPNs which were requested if Kerberos delegation was used.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

Security Monitoring Recommendations:

For 4769(S, F): A Kerberos service ticket was requested.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Account Information\Account Name ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Account Information\Account Name ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Account Information\Account Name ” that corresponds to the accounts that should never be used.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Account Information\Account Domain ” corresponding to another domain or “external” location.

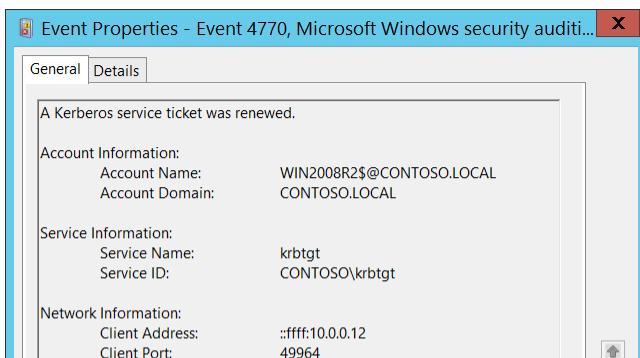
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.

Monitor the target **Computer**: (or other target device) for actions performed by the **"Account Information\Account Name"** that you are concerned about.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor **"User ID"** for names that don't comply with naming conventions.

- If you know that **Account Name** should never request any tickets for (that is, never get access to) a particular computer account or service account, monitor for [4769](#) events with the corresponding **Account Name** and **Service ID** fields.
- You can track all [4769](#) events where the **Client Address** is not from your internal IP range or not from private IP ranges.
- If you know that **Account Name** should be able to request tickets (should be used) only from a known whitelist of IP addresses, track all **Client Address** values for this **Account Name** in [4769](#) events. If **Client Address** is not from your whitelist of IP addresses, generate the alert.
- All **Client Address** = ::1 means local TGS requests, which means that the **Account Name** logged on to a domain controller before making the TGS request. If you have a whitelist of accounts allowed to log on to domain controllers, monitor events with **Client Address** = ::1 and any **Account Name** outside the whitelist.
- All [4769](#) events with **Client Port** field value > 0 and < 1024 should be examined, because a well-known port was used for outbound connection.



A screenshot of the Windows Event Viewer showing the properties of Event 4770. The event details are as follows:

- General:**
 - A Kerberos service ticket was renewed.
 - Account Information:** Account Name: WIN2008R2\$@CONTOSO.LOCAL; Account Domain: CONTOSO.LOCAL
 - Service Information:** Service Name: krbtgt; Service ID: CONTOSO\krbtgt
 - Network Information:** Client Address: ::ffff:10.0.0.12; Client Port: 49964

- Monitor for a **Ticket Encryption Type** of **0x1** or **0x3**, which means the DES algorithm was used. DES should not be in use, because of low security and known vulnerabilities. It is disabled by default starting from Windows 7 and Windows Server 2008 R2.
- Starting with Windows Vista and Windows Server 2008, monitor for a **Ticket Encryption Type** other than **0x11** and **0x12**. These are the expected values, starting with these operating systems, and represent AES-family algorithms.
- If you have a list of important **Failure Codes**, monitor for these codes.

4770(S): A Kerberos service ticket was renewed.

Event Description:

This event generates for every Ticket Granting Service (TGS) ticket renewal.

This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



The screenshot shows the XML representation of Event 4770. The XML is as follows:

```

<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
    <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
    <EventID>4770</EventID>
    <Version>0</Version>
    <Level>0</Level>
    <Task>14337</Task>
    <Opcode>0</Opcode>
  </System>
  <Keywords>Audit Success</Keywords>
  <Computer>DC01.contoso.local</Computer>
</Event>

```

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4770</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>14337</Task>
  <Opcode>0</Opcode>
</System>
<Keywords>Audit Success</Keywords>
<Computer>DC01.contoso.local</Computer>
</Event>

```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-07T03:26:23.466552900Z" />
<EventRecordID>166481</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1084" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserName">WIN2008R2$@CONTOSO.LOCAL</Data>
  <Data Name="TargetDomainName">CONTOSO.LOCAL</Data>
  <Data Name="ServiceName">krbtgt</Data>
  <Data Name="ServiceSid">S-1-5-21-3457937927-2839227994-823803824-502</Data>
  <Data Name="TicketOptions">0x2</Data>
  <Data Name="TicketEncryptionType">0x12</Data>
  <Data Name="IpAddress">::ffff:10.0.0.12</Data>
  <Data Name="IpPort">49964</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Account Information:

- **Account Name** [Type = UnicodeString]: the User Principal Name (UPN) of the account that requested ticket renewal. Computer account name ends with \$ character in UPN. This field typically has the following value format: user_account_name@FULL_DOMAIN_NAME.
 - User account example: dadmin@CONTOSO.LOCAL
 - Computer account example: WIN81\$@CONTOSO.LOCAL

This parameter in this event is optional and can be empty in some cases.

- **Account Domain** [Type = UnicodeString]: the name of the Kerberos Realm that **Account Name** belongs to. This can appear in a variety of formats, including the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

This parameter in this event is optional and can be empty in some cases.

Service Information:

- **Service Name** [Type = UnicodeString]: the name of the account or computer for which the TGS ticket was renewed.
 - This parameter in this event is optional and can be empty in some cases.

- **Service ID** [Type = SID]: SID of the account or computer object for which the TGS ticket was renewed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

Network Information:

- **Client Address** [Type = UnicodeString]: IP address of the computer from which the TGS renewal request was received. Formats vary, and include the following:
 - IPv6 or **IPv4** address.
 - **::ffff:IPv4_address**.
 - **::1** - localhost.
- **Client Port** [Type = UnicodeString]: source port number of client network connection (TGS renewal request connection).
 - 0 for local (localhost) requests.

Additional information:

- **Ticket Options:** [Type = HexInt32]: this is a set of different Ticket Flags in hexadecimal format.

Example:

- Ticket Options: 0x40810010
- Binary view: 010000001000000100000000000010000
- Using **MSB 0** bit numbering we have bit 1, 8, 15 and 27 set = Forwardable, Renewable, Canonicalize, Renewable-ok.

In the table below “**MSB 0**” bit numbering is used, because RFC documents use this style. In “**MSB 0**” style bit numbering begins from left.



The most common values:

- 0x40810010 - Forwardable, Renewable, Canonicalize, Renewable-ok
- 0x40810000 - Forwardable, Renewable, Canonicalize
- 0x60810010 - Forwardable, Forwarded, Renewable, Canonicalize, Renewable-ok

Bit	Flag Name	Description
0	Reserved	-
1	Forwardable	(TGT only). Tells the ticket-granting service that it can issue a new TGT—based on the presented TGT—with a different network address based on the presented TGT.
2	Forwarded	Indicates either that a TGT has been forwarded or that a ticket was issued from a forwarded TGT.
3	Proxyable	(TGT only). Tells the ticket-granting service that it can issue tickets with a network address that differs from the one in the TGT.
4	Proxy	Indicates that the network address in the ticket is different from the one in the TGT used to obtain the ticket.
5	Allow-postdate	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).

6	Postdated	Postdated tickets SHOULD NOT be supported in KILE (Microsoft Kerberos Protocol Extension).
7	Invalid	This flag indicates that a ticket is invalid, and it must be validated by the KDC before use. Application servers must reject tickets which have this flag set.
8	Renewable	Used in combination with the End Time and Renew Till fields to cause tickets with long life spans to be renewed at the KDC periodically.
9	Initial	Indicates that a ticket was issued using the authentication service (AS) exchange and not issued based on a TGT.
10	Pre-authent	Indicates that the client was authenticated by the KDC before a ticket was issued. This flag usually indicates the presence of an authenticator in the ticket. It can also flag the presence of credentials taken from a smart card logon.
11	Opt-hardware-auth	This flag was originally intended to indicate that hardware-supported authentication was used during pre-authentication. This flag is no longer recommended in the Kerberos V5 protocol. KDCs MUST NOT issue a ticket with this flag set. KDCs SHOULD NOT preserve this flag if it is set by another KDC.
12	Transited-policy-checked	KILE MUST NOT check for transited domains on servers or a KDC. Application servers MUST ignore the TRANSITED-POLICY-CHECKED flag.
13	Ok-as-delegate	The KDC MUST set the OK-AS-DELEGATE flag if the service account is trusted for delegation.
14	Request-anonymous	KILE not use this flag.
15	Name-canonicalize	In order to request referrals the Kerberos client MUST explicitly request the "canonicalize" KDC option for the AS-REQ or TGS-REQ.
16-25	Unused	-
26	Disable-transited-check	By default the KDC will check the transited field of a TGT against the policy of the local realm before it will issue derivative tickets based on the TGT. If this flag is set in the request, checking of the transited field is disabled. Tickets issued without the performance of this check will be noted by the reset (0) value of the TRANSITED-POLICY-CHECKED flag, indicating to the application server that the transited field must be checked locally. KDCs are encouraged but not required to honor the DISABLE-TRANSITED-CHECK option. Should not be in use, because Transited-policy-checked flag is not supported by KILE.
27	Renewable-ok	The RENEWABLE-OK option indicates that a renewable ticket will be acceptable if a ticket with the requested life cannot otherwise be provided, in which case a renewable ticket may be issued with a renew-till equal to the requested end time. The value of the renew-till field may still be limited by local limits, or limits selected by the individual principal or server.
28	Enc-tkt-in-skey	No information.
29	Unused	-
30	Renew	The RENEW option indicates that the present request is for a renewal. The ticket provided is encrypted in the secret key for the server on which it is valid. This option will only be honored if the ticket to be renewed has its RENEWABLE flag set and if the time in its renew-till field has not passed. The ticket to be renewed is passed in the padata field as part of the authentication header.
31	Validate	This option is used only by the ticket-granting service. The VALIDATE option indicates that the request is to validate a postdated ticket. Should not be in use, because postdated tickets are not supported by KILE.

- **Ticket Encryption Type:** [Type = HexInt32]: the cryptographic suite that was used in renewed TGS.

Type	Type Name	Description
0x1	DES-CBC-CRC	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x3	DES-CBC-MD5	Disabled by default starting from Windows 7 and Windows Server 2008 R2.
0x11	AES128-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x12	AES256-CTS-HMAC-SHA1-96	Supported starting from Windows Server 2008 and Windows Vista.
0x17	RC4-HMAC	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0x18	RC4-HMAC-EXP	Default suite for operating systems before Windows Server 2008 and Windows Vista.
0xFFFFFFFF or 0xffffffff	-	This type shows in Audit Failure events.

Security Monitoring Recommendations:

For 4770(S): A Kerberos service ticket was renewed.

- This event typically has informational only purpose.

4773(F): A Kerberos service ticket request failed.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system. [4769](#) failure event is generated instead.

Audit Other Account Logon Events

General Subcategory Information:

This auditing subcategory does not contain any events. It is intended for future use.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	No	No	This auditing subcategory does not contain any events. It is intended for future use, and there is no reason to enable it.
Member Server	No	No	No	No	This auditing subcategory does not contain any events. It is intended for future use, and there is no reason to enable it.
Workstation	No	No	No	No	This auditing subcategory does not contain any events. It is intended for future use, and there is no reason to enable it.

Account Management

Audit Application Group Management

Audit Application Group Management generates events for actions related to [application groups](#), such as group creation, modification, addition or removal of group member and some other actions.

[Application groups](#) are used by [Authorization Manager](#).

Audit Application Group Management subcategory is out of scope of this document, because [Authorization Manager](#) is very rarely in use and it is deprecated starting from Windows Server 2012.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	-	-	-	-	This subcategory is outside the scope of this document.
Member Server	-	-	-	-	This subcategory is outside the scope of this document.
Workstation	-	-	-	-	This subcategory is outside the scope of this document.

4783(S): A basic application group was created.

4784(S): A basic application group was changed.

4785(S): A member was added to a basic application group.

4786(S): A member was removed from a basic application group.

4787(S): A non-member was added to a basic application group.

4788(S): A non-member was removed from a basic application group.

4789(S): A basic application group was deleted.

4790(S): An LDAP query group was created.

4791(S): An LDAP query group was changed.

4792(S): An LDAP query group was deleted.

Audit Computer Account Management

Audit Computer Account Management determines whether the operating system generates audit events when a computer account is created, changed, or deleted.

This policy setting is useful for tracking account-related changes to computers that are members of a domain.

Event volume: Low on domain controllers.

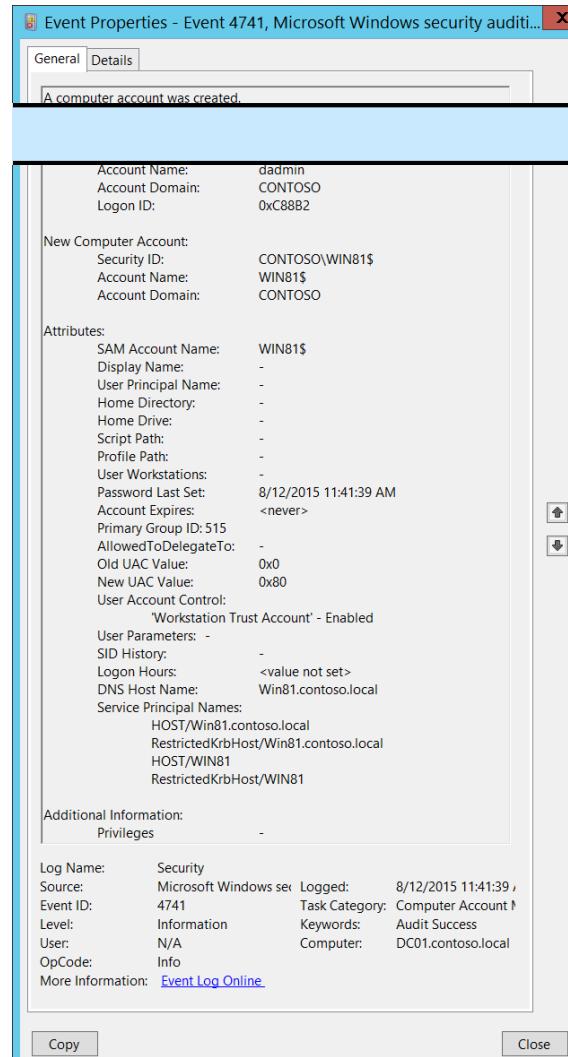
This subcategory allows you to audit events generated by changes to computer accounts such as when a computer account is created, changed, or deleted.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	We recommend monitoring changes to critical computer objects in Active Directory, such as domain controllers, administrative workstations, and critical servers. It's especially important to be informed if any critical computer account objects are deleted. Additionally, events in this subcategory will give you information about who deleted, created, or modified a computer object, and when the action was taken. Typically volume of these events is low on domain controllers. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	No	No	No	No	This subcategory generates events only on domain controllers.
Workstation	No	No	No	No	This subcategory generates events only on domain controllers.

Events List:

- [4741\(S\)](#): A computer account was created.
- [4742\(S\)](#): A computer account was changed.
- [4743\(S\)](#): A computer account was deleted.

4741(S): A computer account was created.



Event Description:

This event generates every time a new computer object is created.
This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4741</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13825</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-12T18:41:39.201898100Z" />
  <EventRecordID>170254</EventRecordID>
  <Correlation />
  <Execution ProcessID="520" ThreadID="1096" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="TargetUserName">WIN81$</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6116</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0xc88b2</Data>
  <Data Name="PrivilegeList"></Data>
```

```
<Data Name="SamAccountName">WIN81$</Data>
<Data Name="DisplayName">-</Data>
<Data Name="UserPrincipalName">-</Data>
<Data Name="HomeDirectory">-</Data>
```

```
<Data Name="HomePath"></Data>
<Data Name="ScriptPath"></Data>
<Data Name="ProfilePath"></Data>
<Data Name="UserWorkstations"></Data>
<Data Name="PasswordLastSet">8/12/2015 11:41:39 AM</Data>
<Data Name="AccountExpires">%1794</Data>
<Data Name="PrimaryGroupId">515</Data>
<Data Name="AllowedToDelegateTo"></Data>
<Data Name="OldUacValue">0x0</Data>
<Data Name="NewUacValue">0x80</Data>
<Data Name="UserAccountControl">%2087</Data>
<Data Name="UserParameters"></Data>
<Data Name="SidHistory"></Data>
<Data Name="LogonHours">%1793</Data>
<Data Name="DnsHostName">Win81.contoso.local</Data>
<Data Name="ServicePrincipalNames">HOST/Win81.contoso.local RestrictedKrbHost/Win81.contoso.local HOST/WIN81 RestrictedKrbHost/WIN81</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested the “create Computer object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]:** the name of the account that requested the “create Computer object” operation.
- **Account Domain [Type = UnicodeString]:** subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID [Type = HexInt64]:** hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

New Computer Account:

- **Security ID** [Type = SID]: SID of created computer account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the computer account that was created. For example: WIN81\$
- **Account Domain** [Type = UnicodeString]: domain name of created computer account. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

Attributes:

- **SAM Account Name** [Type = UnicodeString]: logon name for account used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). The value of **sAMAccountName** attribute of new computer object. For example: WIN81\$.
- **Display Name** [Type = UnicodeString]: the value of **displayName** attribute of new computer object. It is a name displayed in the address book for a particular account (typically – user account). This is usually the combination of the user's first name, middle initial, and last name. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **User Principal Name** [Type = UnicodeString]: internet-style login name for the account, based on the Internet standard RFC 822. By convention this should map to the account's email name. This parameter contains the value of **userPrincipalName** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Home Directory** [Type = UnicodeString]: user's home directory. If **homeDrive** attribute is set and specifies a drive letter, **homeDirectory** should be a UNC path. The path must be a network UNC of the form \\Server\Share\Directory. This parameter contains the value of **homeDirectory** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Home Drive** [Type = UnicodeString]: specifies the drive letter to which to map the UNC path specified by **homeDirectory** account's attribute. The drive letter must be specified in the form “**DRIVE LETTER:**”. For example – “H:”. This parameter contains the value of **homeDrive** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Script Path** [Type = UnicodeString]: specifies the path of the account's logon script. This parameter contains the value of **scriptPath** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Profile Path** [Type = UnicodeString]: specifies a path to the account's profile. This value can be a null string, a local absolute path, or a UNC path. This parameter contains the value of **profilePath** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **User Workstations** [Type = UnicodeString]: contains the list of NetBIOS or DNS names of the computers from which the user can logon. Each computer name is separated by a comma. The name of a computer is the **sAMAccountName** property of a computer object. This parameter contains the value of **userWorkstations** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Password Last Set** [Type = UnicodeString]: last time the account's password was modified. For manually created computer account, using Active Directory Users and Computers snap-in, this field typically has value “<never>”. For computer account created during standard domain join procedure this field will contain time when computer object was

created, because password creates during domain join procedure. For example: 8/12/2015 11:41:39 AM. This parameter contains the value of **pwdLastSet** attribute of new computer object.

- **Account Expires** [Type = UnicodeString]: the date when the account expires. This parameter contains the value of **accountExpires** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Primary Group ID** [Type = UnicodeString]: Relative Identifier (RID) of computer's object primary group.

Relative identifier (RID) is a variable length number that is assigned to objects at creation and becomes part of the object's Security Identifier (SID) that uniquely identifies an account or group within a domain.

Typically, **Primary Group** field for new computer accounts has the following values:

- 516 (Domain Controllers) – for domain controllers.
- 521 (Read-only Domain Controllers) – for read-only domain controllers (RODC).
- 515 (Domain Computers) – for member servers and workstations.

See this article <https://support.microsoft.com/en-us/kb/243330> for more information. This parameter contains the value of **primaryGroupID** attribute of new computer object.

- **AllowedToDelegateTo** [Type = UnicodeString]: the list of SPNs to which this account can present delegated credentials. Can be changed using Active Directory Users and Computers management console in **Delegation** tab of computer account. Typically it is set to “-” for new computer objects. This parameter contains the value of **AllowedToDelegateTo** attribute of new computer object. See description of **AllowedToDelegateTo** field for “[4742: A computer account was changed](#)” event for more details.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Old UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user or computer account. **Old UAC value** always “**0x0**” for new computer accounts. This parameter contains the previous value of **userAccountControl** attribute of computer object.
- **New UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user or computer account. This parameter contains the value of **userAccountControl** attribute of new computer object.

To decode this value, you can go through the property value definitions in the “Table 7. User’s or Computer’s account UAC flags.” from largest to smallest. Compare each property value to the flags value in the event. If the flags value in the event is greater than or equal to the property value, then the property is “set” and applies to that event. Subtract the property value from the flags value in the event and note that the flag applies and then go on to the next flag.

Here's an example: Flags value from event: **0x1**

Decoding:

- PASSWD_NOTREQD 0x0020
- LOCKOUT 0x0010
- HOMEDIR_REQUIRED 0x0008
- (undeclared) 0x0004
- ACCOUNTDISABLE 0x0002
- SCRIPT 0x0001

0x0020 > 0x15, so PASSWD_NOTREQD does not apply to this event
 0x10 < 0x15, so LOCKOUT applies to this event. 0x15 - 0x10 = 0x5
 0x4 < 0x5, so the undeclared value is set. We'll pretend it doesn't mean anything. 0x5 - 0x4 = 0x1
 0x2 > 0x1, so ACCOUNTDISABLE does not apply to this event
 0x1 = 0x1, so SCRIPT applies to this event. 0x1 - 0x1 = 0x0, we're done.
 So this UAC flags value decodes to: LOCKOUT and SCRIPT

- **User Account Control** [Type = UnicodeString]: shows the list of changes in **userAccountControl** attribute. You will see a line of text for each change. For new computer accounts, when the object for this account was created, the **userAccountControl** value was considered to be “**0x0**”, and then it was changed from “**0x0**” to the real value for the account’s **userAccountControl** attribute. See possible values in the table below. In the “User Account Control field text” column, you can see the text that will be displayed in the **User Account Control** field in 4741 event.

Flag Name	userAccount Control in hexadecimal	userAccount Control in decimal	Description	User Account Control field text
SCRIPT	0x0001	1	The logon script will be run.	Changes of this flag do not show in 4741 events.
ACCOUNTDISABLE	0x0002	2	The user account is disabled.	Account Disabled Account Enabled
Undeclared	0x0004	4	This flag is undeclared.	Changes of this flag do not show in 4741 events.
HOMEDIR_REQUIRED	0x0008	8	The home folder is required.	'Home Directory Required' - Enabled 'Home Directory Required' - Disabled
LOCKOUT	0x0010	16		Changes of this flag do not show in 4741 events.
PASSWD_NOTREQD	0x0020	32	No password is required.	'Password Not Required' - Enabled 'Password Not Required' - Disabled
PASSWD_CANT_CHANGE	0x0040	64	The user cannot change the password. This is a permission on the user's object.	Changes of this flag do not show in 4741 events.
ENCRYPTED_TEXT_PWD_ALLO_WED	0x0080	128	The user can send an encrypted password. Can be set using “Store password using reversible encryption” checkbox.	'Encrypted Text Password Allowed' - Disabled 'Encrypted Text Password Allowed' - Enabled
TEMP_DUPLICATE_ACCOUNT	0x0100	256	This is an account for users whose primary account is in another domain. This account provides user access to this domain, but not to any domain that trusts this domain. This is sometimes referred to as a local user account.	Cannot be set for computer account.
NORMAL_ACCOUNT	0x0200	512	This is a default account type that represents a typical user.	'Normal Account' - Disabled 'Normal Account' - Enabled
INTERDOMAIN_TRUST_ACCOUNT	0x0800	2048	This is a permit to trust an account for a system domain that trusts other domains.	Cannot be set for computer account.

WORKSTATION_TRUST_ACCOUNT	0x1000	4096	This is a computer account for a computer that is running Microsoft Windows NT 4.0 Workstation, Microsoft Windows NT 4.0 Server, Microsoft Windows 2000 Professional, or Windows 2000 Server and is a member of this domain.	'Workstation Trust Account' - Disabled 'Workstation Trust Account' - Enabled
SERVER_TRUST_ACCOUNT	0x2000	8192	This is a computer account for a domain controller that is a member of this domain.	'Server Trust Account' - Enabled 'Server Trust Account' - Disabled
DONT_EXPIRE_PASSWORD	0x10000	65536	Represents the password, which should never expire on the account. Can be set using "Password never expires" checkbox.	'Don't Expire Password' - Disabled 'Don't Expire Password' - Enabled
MNS_LOGON_ACCOUNT	0x20000	131072	This is an MNS logon account.	'MNS Logon Account' - Disabled 'MNS Logon Account' - Enabled
SMARTCARD_REQUIRED	0x40000	262144	When this flag is set, it forces the user to log on by using a smart card.	'Smartcard Required' - Disabled 'Smartcard Required' - Enabled
TRUSTED_FOR_DELEGATION	0x80000	524288	When this flag is set, the service account (the user or computer account) under which a service runs is trusted for Kerberos delegation. Any such service can impersonate a client requesting the service. To enable a service for Kerberos delegation, you must set this flag on the userAccountControl property of the service account. If you enable Kerberos constraint or unconstraint delegation or disable these types of delegation in Delegation tab you will get this flag changed.	'Trusted For Delegation' - Enabled 'Trusted For Delegation' - Disabled
NOT_DELEGATED	0x100000	1048576	When this flag is set, the security context of the user is not delegated to a service even if the service account is set as trusted for Kerberos delegation. Can be set using "Account is sensitive and cannot be delegated" checkbox.	'Not Delegated' - Disabled 'Not Delegated' - Enabled
USE_DES_KEY_ONLY	0x200000	2097152	Restrict this principal to use only Data Encryption Standard (DES) encryption types for keys. Can be set using "Use Kerberos DES encryption types for this account" checkbox.	'Use DES Key Only' - Disabled 'Use DES Key Only' - Enabled
DONT_REQ_PREAUTH	0x400000	4194304	This account does not require Kerberos pre-authentication for logging on. Can be set using "Do not require Kerberos preauthentication" checkbox.	'Don't Require Preauth' - Disabled 'Don't Require Preauth' - Enabled

PASSWORD_EXPIRED	0x8000000	8388608	The user's password has expired.	Changes of this flag do not show in 4741 events.
TRUSTED_TO_AUTH_FOR_DELEGATION	0x10000000	16777216	The account is enabled for delegation. This is a security-sensitive setting. Accounts that have this option enabled should be tightly controlled. This setting lets a service that runs under the account assume a client's identity and authenticate as that user to other remote servers on the network. If you enable Kerberos protocol transition delegation or disable this type of delegation in Delegation tab you will get this flag changed.	'Trusted To Authenticate For Delegation' - Disabled 'Trusted To Authenticate For Delegation' - Enabled
PARTIAL_SECRETS_ACCOUNT	0x040000000	67108864	The account is a read-only domain controller (RODC). This is a security-sensitive setting. Removing this setting from an RODC compromises security on that server.	No information.

Table 7. User's or Computer's account UAC flags.

- **User Parameters** [Type = UnicodeString]: if you change any setting using Active Directory Users and Computers management console in Dial-in tab of computer's account properties, then you will see **<value changed, but not displayed>** in this field in "[4742\(S\)](#): A computer account was changed." This parameter might not be captured in the event, and in that case appears as "-".
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sIDHistory** property. This parameter contains the value of **sIDHistory** attribute of new computer object. This parameter might not be captured in the event, and in that case appears as "-".
- **Logon Hours** [Type = UnicodeString]: hours that the account is allowed to logon to the domain. The value of **logonHours** attribute of new computer object. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. You will see **<value not set>** value for new created computer accounts in event 4741.
- **DNS Host Name** [Type = UnicodeString]: name of computer account as registered in DNS. The value of **dNSHostName** attribute of new computer object. For manually created computer account objects this field has value "-".
- **Service Principal Names** [Type = UnicodeString]: The list of SPNs, registered for computer account. For new computer accounts it will typically contain HOST SPNs and RestrictedKrbHost SPNs. The value of **servicePrincipalName** attribute of new computer object. For manually created computer objects it is typically equals "-". This is an example of **Service Principal Names** field for new domain joined workstation:

HOST/Win81.contoso.local
 RestrictedKrbHost/Win81.contoso.local
 HOST/WIN81
 RestrictedKrbHost/WIN81

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in the table below:

Privilege Name	User Right Group Policy Name	Description
----------------	------------------------------	-------------

SeAssignPrimaryTokenPrivilege	Replace a process-level token	<p>Required to assign the <i>primary token</i> of a process.</p> <p>With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.</p>
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	<p>Required to perform backup operations.</p> <p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object.</p>

	delegation	The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
SeMachineAccountPrivilege	Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY

		<ul style="list-style-type: none"> • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	<p>This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers.</p> <p>With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.</p>
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	<p>Required to gather profiling information for the entire system.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.</p>
SeSystemtimePrivilege	Change the system time	<p>Required to modify the system time.</p> <p>With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.</p>
SeTakeOwnershipPrivilege	Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>
SeTcbPrivilege	Act as part of the operating system	<p>This privilege identifies its holder as part of the trusted computer base.</p> <p>This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.</p>
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivil	Access Credential Manager as	Required to access Credential Manager as a trusted caller.

ege	a trusted caller	
SeUndockPrivilege	Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

Table 8. User Privileges.

Security Monitoring Recommendations:

For 4741(S): A computer account was created.

Important For this event, also see [Appendix A: Security monitoring recommendations for many audit events](#).

- If your information security monitoring policy requires you to monitor computer account creation, monitor this event.
- Consider whether to track the following fields and values:

Field and value to track	Reason to track
SAM Account Name: empty or -	This field must contain the computer account name. If it is empty or -, it might indicate an anomaly.
Display Name is not - User Principal Name is not - Home Directory is not - Home Drive is not - Script Path is not - Profile Path is not - User Workstations is not - AllowedToDelegateTo is not -	Typically these fields are - for new computer accounts. Other values might indicate an anomaly and should be monitored.
Password Last Set is <never>	This typically means this is a manually created computer account, which you might need to monitor.
Account Expires is not <never>	Typically this field is <never> for new computer accounts. Other values might indicate an anomaly and should be monitored.
Primary Group ID is any value other than 515.	Typically, the Primary Group ID value is one of the following: <ul style="list-style-type: none"> • 516 for domain controllers • 521 for read only domain controllers (RODCs) • 515 for servers and workstations (domain computers) If the Primary Group ID is 516 or 521, it is a new domain controller or RODC, and the event should be monitored. If the value is not 516, 521, or 515, it is not a typical value and should be monitored.

Old UAC Value is not 0x0

Typically this field is **0x0** for new computer accounts. Other values might indicate an anomaly and should be monitored.

SID History is not -	This field will always be set to - unless the account was migrated from another domain.
Logon Hours value other than <value not set>	This should always be <value not set> for new computer accounts.

- Consider whether to track the following account control flags:

User account control flag to track	Information about the flag
'Encrypted Text Password Allowed' – Enabled	Should not be set for computer accounts. By default, it will not be set, and it cannot be set in the account properties in Active Directory Users and Computers.
'Server Trust Account' – Enabled	Should be enabled only for domain controllers.
'Don't Expire Password' – Enabled	Should not be enabled for new computer accounts, because the password automatically changes every 30 days by default. For computer accounts, this flag cannot be set in the account properties in Active Directory Users and Computers.
'Smartcard Required' – Enabled	Should not be enabled for new computer accounts.
'Trusted For Delegation' – Enabled	Should not be enabled for new member servers and workstations. It is enabled by default for new domain controllers.
'Not Delegated' – Enabled	Should not be enabled for new computer accounts.
'Use DES Key Only' – Enabled	Should not be enabled for new computer accounts. For computer accounts, it cannot be set in the account properties in Active Directory Users and Computers.
'Don't Require Preauth' – Enabled	Should not be enabled for new computer accounts. For computer accounts, it cannot be set in the account properties in Active Directory Users and Computers.
'Trusted To Authenticate For Delegation' – Enabled	Should not be enabled for new computer accounts by default.

4742(S): A computer account was changed.

Event Properties - Event 4742, Microsoft Windows security auditi... X

General Details

A computer account was changed.

Subject:

Security ID:	CONTOSO\admind
Account Name:	admind
Account Domain:	CONTOSO
Logon ID:	0x2E80C

Computer Account That Was Changed:

Security ID:	CONTOSO\WIN81\$
Account Name:	WIN81\$
Account Domain:	CONTOSO

Changed Attributes:

SAM Account Name:	-
Display Name:	-
User Principal Name:	-
Home Directory:	-
Home Drive:	-
Script Path:	-
Profile Path:	-
User Workstations:	-
Password Last Set:	-
Account Expires:	-
Primary Group ID:	-
AllowedToDelegateTo:	<value not set>

Event Description:

This event generates every time a computer object is changed.

This event generates only on domain controllers.

You might see the same values for **Subject\Security ID** and **Computer Account That Was Changed\Security ID** in this event.

This usually happens when you reboot a computer after adding it to the domain (the change takes effect after the reboot).

For each change, a separate 4742 event will be generated.

Some changes do not invoke a 4742 event, for example, changes made using Active Directory Users and Computers management console in **Managed By** tab in computer account properties.

You might see this event without any changes inside, that is, where all **Changed Attributes** appear as “-”. This usually happens when a change is made to an attribute that is not listed in the event. In this case there is no way to determine which attribute was changed. For example, this would happen if you change the **Description** of a group object using the Active Directory Users and Computers administrative console. Also, if the [discretionary access control list](#) (DACL) is changed, a 4742 event will generate, but all attributes will be “-”.

Important: If you manually change any user-related setting or attribute, for example if you set the SMARTCARD_REQUIRED flag in **userAccountControl** for the computer account, then the **sAMAccountType** of the computer account will be changed to **NORMAL_USER_ACCOUNT** and you will get “[4738: A user account was changed](#)” instead of 4742 for this computer account. Essentially, the computer account will “become” a user account. For **NORMAL_USER_ACCOUNT** you will always get events from [Audit User Account Management](#) subcategory. We strongly recommend that you avoid changing any user-related settings manually for computer objects.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Trusted For Delegation - Enabled

User Parameters: -
SID History: -
Logon Hours: -
DNS Host Name: -
Service Principal Names: -

Additional Information:
Privileges: -

Log Name: Security
Source: Microsoft Windows sev
Event ID: 4742
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 8/13/2015 7:35:01 PM
Task Category: Computer Account M
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4742</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13825</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-14T02:35:01.252397000Z" />
<EventRecordID>171754</EventRecordID>
<Correlation />

```

<Execution ProcessID="520" ThreadID="1108" />

<Channel>Security</Channel>

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ComputerAccountChange">-</Data>
<Data Name="TargetUserName">WIN81$</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6116</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x2e80c</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">-</Data>
<Data Name="DisplayName">-</Data>
<Data Name="UserPrincipalName">-</Data>
<Data Name="HomeDirectory">-</Data>
<Data Name="HomePath">-</Data>
<Data Name="ScriptPath">-</Data>
<Data Name="ProfilePath">-</Data>
<Data Name="UserWorkstations">-</Data>
<Data Name="PasswordLastSet">-</Data>
<Data Name="AccountExpires">-</Data>
<Data Name="PrimaryGroupId">-</Data>
<Data Name="AllowedToDelegateTo">%%1793</Data>
<Data Name="OldUacValue">0x80</Data>
<Data Name="NewUacValue">0x2080</Data>
<Data Name="UserAccountControl">%%2093</Data>
<Data Name="UserParameters">-</Data>
<Data Name="SidHistory">-</Data>
<Data Name="LogonHours">-</Data>
<Data Name="DnsHostName">-</Data>
<Data Name="ServicePrincipalNames">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change Computer object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change Computer object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Computer Account That Was Changed:

- **Security ID** [Type = SID]: SID of changed computer account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the computer account that was changed. For example: WIN81\$
- **Account Domain** [Type = UnicodeString]: domain name of changed computer account. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

Changed Attributes:

If attribute was not changed it will have “-“ value.

- **SAM Account Name** [Type = UnicodeString]: logon name for account used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). If the value of **sAMAccountName** attribute of computer object was changed, you will see the new value here. For example: WIN8\$.
- **Display Name** [Type = UnicodeString]: it is a name displayed in the address book for a particular account (typically – user account). This is usually the combination of the user’s first name, middle initial, and last name. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. If the value of **displayName** attribute of computer object was changed, you will see the new value here.
- **User Principal Name** [Type = UnicodeString]: internet-style login name for the account, based on the Internet standard RFC 822. By convention this should map to the account’s email name. If the value of **userPrincipalName** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.

- **Home Directory** [Type = UnicodeString]: user's home directory. If **homeDrive** attribute is set and specifies a drive letter, **homeDirectory** should be a UNC path. The path must be a network UNC of the form \\Server\Share\Directory. If the value of **homeDirectory** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Home Drive** [Type = UnicodeString]: specifies the drive letter to which to map the UNC path specified by **homeDirectory** account's attribute. The drive letter must be specified in the form "DRIVE LETTER:". For example – "H:". If the value of **homeDrive** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Script Path** [Type = UnicodeString]: specifies the path of the account's logon script. If the value of **scriptPath** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Profile Path** [Type = UnicodeString]: specifies a path to the account's profile. This value can be a null string, a local absolute path, or a UNC path. If the value of **profilePath** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **User Workstations** [Type = UnicodeString]: contains the list of NetBIOS or DNS names of the computers from which the user can logon. Each computer name is separated by a comma. The name of a computer is the **sAMAccountName** property of a computer object. If the value of **userWorkstations** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Password Last Set** [Type = UnicodeString]: last time the account's password was modified. If the value of **pwdLastSet** attribute of computer object was changed, you will see the new value here. For example: 8/12/2015 11:41:39 AM. This value will be changed, for example, after manual computer account reset action or automatically every 30 days by default for computer objects.
- **Account Expires** [Type = UnicodeString]: the date when the account expires. If the value of **accountExpires** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Primary Group ID** [Type = UnicodeString]: Relative Identifier (RID) of computer's object primary group.

Relative identifier (RID) is a variable length number that is assigned to objects at creation and becomes part of the object's Security Identifier (SID) that uniquely identifies an account or group within a domain.

This field will contain some value if computer's object primary group was changed. You can change computer's primary group using Active Directory Users and Computers management console in the **Member Of** tab of computer object properties. You will see a RID of new primary group as a field value. For example, 515 (Domain Computers) for workstations, is a default primary group.

Typical **Primary Group** values for computer accounts:

- 516 (Domain Controllers) – for domain controllers.
- 521 (Read-only Domain Controllers) – read-only domain controllers (RODC).
- 515 (Domain Computers) – servers and workstations.

See this article <https://support.microsoft.com/en-us/kb/243330> for more information. If the value of **primaryGroupId** attribute of computer object was changed, you will see the new value here.

- **AllowedToDelegateTo** [Type = UnicodeString]: the list of SPNs to which this account can present delegated credentials. Can be changed using Active Directory Users and Computers management console in **Delegation** tab of computer account. If the SPNs list on **Delegation** tab of a computer account was changed, you will see the new SPNs list in **AllowedToDelegateTo** field (note that you will see the new list instead of changes) of this event. This is an example of **AllowedToDelegateTo**:
 - dcom/WIN2012
 - dcom/WIN2012.contoso.local

If the value of **msDS-AllowedToDelegateTo** attribute of computer object was changed, you will see the new value here.

The value can be <value not set>, for example, if delegation was disabled.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Old UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user or computer account. This parameter contains the previous value of **userAccountControl** attribute of computer object.
- **New UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user or computer account. If the value of **userAccountControl** attribute of computer object was changed, you will see the new value here.

To decode this value, you can go through the property value definitions in the “Table 7. User’s or Computer’s account UAC flags.” from largest to smallest. Compare each property value to the flags value in the event. If the flags value in the event is greater than or equal to the property value, then the property is “set” and applies to that event. Subtract the property value from the flags value in the event and note that the flag applies and then go on to the next flag.

Here's an example: Flags value from event: 0x15

Decoding:

- PASSWD_NOTREQD 0x0020
- LOCKOUT 0x0010
- HOMEDIR_REQUIRED 0x0008
- (undeclared) 0x0004
- ACCOUNTDISABLE 0x0002
- SCRIPT 0x0001

0x0020 > 0x15, so PASSWD_NOTREQD does not apply to this event

0x10 < 0x15, so LOCKOUT applies to this event. 0x15 - 0x10 = 0x5

0x4 < 0x5, so the undeclared value is set. We'll pretend it doesn't mean anything. 0x5 - 0x4 = 0x1

0x2 > 0x1, so ACCOUNTDISABLE does not apply to this event

0x1 = 0x1, so SCRIPT applies to this event. 0x1 - 0x1 = 0x0, we're done.

So this UAC flags value decodes to: LOCKOUT and SCRIPT

- **User Account Control** [Type = UnicodeString]: shows the list of changes in **userAccountControl** attribute. You will see a line of text for each change. See possible values in here: “Table 7. User’s or Computer’s account UAC flags.”. In the “User Account Control field text” column, you can see text that will be displayed in the **User Account Control** field in 4742 event.
- **User Parameters** [Type = UnicodeString]: if you change any setting using Active Directory Users and Computers management console in Dial-in tab of computer’s account properties, then you will see <value changed, but not displayed> in this field.
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sIDHistory** property. If the value of **sIDHistory** attribute of computer object was changed, you will see the new value here.

- **Logon Hours** [Type = UnicodeString]: hours that the account is allowed to logon to the domain. If the value of **logonHours** attribute of computer object was changed, you will see the new value here. For computer objects, it is optional, and typically is not set. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **DNS Host Name** [Type = UnicodeString]: name of computer account as registered in DNS. If the value of **dNSHostName** attribute of computer object was changed, you will see the new value here.
- **Service Principal Names** [Type = UnicodeString]: The list of SPNs, registered for computer account. If the SPN list of a computer account changed, you will see the new SPN list in **Service Principal Names** field (note that you will see the new list instead of changes). If the value of **servicePrincipalName** attribute of computer object was changed, you will see the new value here.

Here is an example of **Service Principal Names** field for new domain joined workstation in event 4742 on domain controller, after workstation reboots:

```

HOST/Win81.contoso.local
RestrictedKrbHost/Win81.contoso.local
HOST/WIN81
RestrictedKrbHost/WIN81
TERMSRV/Win81.contoso.local

```

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4742(S): A computer account was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have critical domain computer accounts (database servers, domain controllers, administration workstations, and so on) for which you need to monitor each change, monitor this event with the “**Computer Account That Was Changed\Security ID**” that corresponds to the high-value account or accounts.
- If you have computer accounts for which any change in the services list on the **Delegation** tab should be monitored, monitor this event when **AllowedToDelegateTo** is not -. This value means the services list was changed.
- Consider whether to track the following fields and values:

Field and value to track	Reason to track
Display Name is not - User Principal Name is not - Home Directory is not - Home Drive is not - Script Path is not - Profile Path is not - User Workstations is not - Account Expires is not - Logon Hours is not -	Typically these fields are - for computer accounts. Other values might indicate an anomaly and should be monitored.

Password Last Set changes occur more often than usual	Changes that are more frequent than the default (typically once a month) might indicate an anomaly or attack.
Primary Group ID is not 516, 521, or 515	<p>Typically, the Primary Group ID value is one of the following:</p> <ul style="list-style-type: none"> • 516 for domain controllers • 521 for read only domain controllers (RODCs) • 515 for servers and workstations (domain computers) <p>Other values should be monitored.</p>
For computer accounts for which the services list (on the Delegation tab) should not be empty: AllowedToDelegateTo is marked <value not set>	If AllowedToDelegateTo is marked <value not set> on computers that previously had a services list (on the Delegation tab), it means the list was cleared.
SID History is not -	This field will always be set to - unless the account was migrated from another domain.

- Consider whether to track the following account control flags:

User account control flag to track	Information about the flag
'Password Not Required' – Enabled	Should not be set for computer accounts. Computer accounts typically require a password by default, except manually created computer objects.
'Encrypted Text Password Allowed' – Enabled	Should not be set for computer accounts. By default, it will not be set, and it cannot be set in the account properties in Active Directory Users and Computers.
'Server Trust Account' – Enabled	Should be enabled only for domain controllers.
'Server Trust Account' – Disabled	Should not be disabled for domain controllers.
'Don't Expire Password' – Enabled	Should not be enabled for computer accounts, because the password automatically changes every 30 days by default. For computer accounts, this flag cannot be set in the account properties in Active Directory Users and Computers.
'Smartcard Required' – Enabled	Should not be enabled for computer accounts.
'Trusted For Delegation' – Enabled	Means that Kerberos Constraint or Unconstraint delegation was enabled for the computer account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Trusted For Delegation' – Disabled	Means that Kerberos Constraint or Unconstraint delegation was disabled for the computer account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action. Also, if you have a list of computer accounts for which delegation is critical and should not be disabled, monitor this for those accounts.
'Trusted To Authenticate For Delegation' – Enabled	Means that Protocol Transition delegation was enabled for the computer account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.

'Trusted To Authenticate For Delegation' – Disabled	Means that Protocol Transition delegation was disabled for the computer account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action. Also, if you have a list of computer accounts for which delegation is critical and should not be disabled, monitor this for those accounts.
'Not Delegated' – Enabled	Means that Account is sensitive and cannot be delegated was selected for the computer account. For computer accounts, this flag cannot be set using the graphical interface. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Use DES Key Only' – Enabled	Should not be enabled for computer accounts. For computer accounts, it cannot be set in the account properties in Active Directory Users and Computers.
'Don't Require Preauth' - Enabled	Should not be enabled for computer accounts. For computer accounts, it cannot be set in the account properties in Active Directory Users and Computers.

4743(S): A computer account was deleted.

Event Properties - Event 4743, Microsoft Windows security auditing.

General Details

Security ID: CONTOSO\dadmin
 Account Name: dadmin
 Account Domain: CONTOSO
 Logon ID: 0x3007B

Target Computer:
 Security ID: S-1-5-21-3457937927-2839227994-823803824-6118
 Account Name: COMPUTERACCOUNT\$
 Account Domain: CONTOSO

Additional Information:
 Privileges: -

Log Name: Security
 Source: Microsoft Windows sec
 Event ID: 4743
 Level: Information
 User: N/A
 OpCode: Info

Logged: 8/14/2015 8:57:08 AM
 Task Category: Computer Account Manager
 Keywords: Audit Success
 Computer: DC01.contoso.local

More Information: [Event Log Online](#)

[Copy](#) [Close](#)

Event Description:

This event generates every time a computer object is deleted.
 This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4743</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13825</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-14T15:57:08.104214100Z" />
<EventRecordID>172103</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1108" />
<Channel>Security</Channel>
```

<Computer>DC01.contoso.local</Computer>

```
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">COMPUTERACCOUNT$</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6118</Data>
<Data Name="SubjectUserId" S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete Computer object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete Computer object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Computer:

- **Security ID** [Type = SID]: SID of deleted computer account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the computer account that was deleted. For example: WIN81\$
- **Account Domain** [Type = UnicodeString]: domain name of deleted computer account. Formats vary, and include the following:

- Domain NETBIOS name example: CONTOSO
- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4743(S): A computer account was deleted.

[**Appendix A: Security monitoring recommendations for many audit events**](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have critical domain computer accounts (database servers, domain controllers, administration workstations, and so on) for which you need to monitor each action (especially deletion), monitor this event with the “**Target Computer\Security ID**” or “**Target Computer\Account Name**” that corresponds to the high-value account or accounts.

Audit Distribution Group Management

Audit Distribution Group Management determines whether the operating system generates audit events for specific distribution-group management tasks.

This subcategory generates events only on domain controllers.

Event volume: Low on domain controllers.

This subcategory allows you to audit events generated by changes to distribution groups such as the following:

- Distribution group is created, changed, or deleted.
- Member is added or removed from a distribution group.

If you need to monitor for group type changes, you need to monitor for "[4764](#): A group's type was changed." "Audit Security Group Management" subcategory success auditing must be enabled.

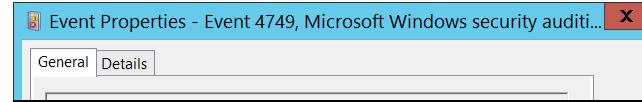
Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	No	IF	No	<p>IF - Typically actions related to distribution groups have low security relevance, much more important to monitor Security Group changes. But if you want to monitor for critical distribution groups changes, such as member was added to internal critical distribution group (executives, administrative group, for example), you need to enable this subcategory for Success auditing.</p> <p>Typically volume of these events is low on domain controllers.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	No	No	No	No	This subcategory generates events only on domain controllers.
Workstation	No	No	No	No	This subcategory generates events only on domain controllers.

Events List:

- [4749\(S\)](#): A security-disabled global group was created.
- [4750\(S\)](#): A security-disabled global group was changed.
- [4751\(S\)](#): A member was added to a security-disabled global group.
- [4752\(S\)](#): A member was removed from a security-disabled global group.
- [4753\(S\)](#): A security-disabled global group was deleted.
- [4759\(S\)](#): A security-disabled universal group was created.
- [4760\(S\)](#): A security-disabled universal group was changed.
- [4761\(S\)](#): A member was added to a security-disabled universal group.
- [4762\(S\)](#): A member was removed from a security-disabled universal group.
- [4763\(S\)](#): A security-disabled universal group was deleted.
- [4744\(S\)](#): A security-disabled local group was created.
- [4745\(S\)](#): A security-disabled local group was changed.
- [4746\(S\)](#): A member was added to a security-disabled local group.
- [4747\(S\)](#): A member was removed from a security-disabled local group.

- [4748\(S\)](#): A security-disabled local group was deleted.

4749(S): A security-disabled global group was created.

 Event Properties - Event 4749, Microsoft Windows security audit... X

General	Details						
<p>Event Description: This event generates every time a new security-disabled (distribution) global group was created. This event generates only on domain controllers.</p> <p>Note For recommendations, see Security Monitoring Recommendations for this event.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;"> Attributes: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x3007B </td> <td style="width: 50%; padding: 5px;"> Group: Security ID: CONTOSO\ServiceDesk Group Name: ServiceDesk Group Domain: CONTOSO </td> </tr> <tr> <td colspan="2" style="padding: 5px;"> Additional Information: SAM Account Name: ServiceDesk SID History: - </td> </tr> <tr> <td colspan="2" style="padding: 5px;"> Log Name: Security Source: Microsoft Windows security Event ID: 4749 Level: Information User: N/A OpCode: Info More Information: Event Log Online </td> </tr> </table>		Attributes: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x3007B	Group: Security ID: CONTOSO\ServiceDesk Group Name: ServiceDesk Group Domain: CONTOSO	Additional Information: SAM Account Name: ServiceDesk SID History: -		Log Name: Security Source: Microsoft Windows security Event ID: 4749 Level: Information User: N/A OpCode: Info More Information: Event Log Online	
Attributes: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x3007B	Group: Security ID: CONTOSO\ServiceDesk Group Name: ServiceDesk Group Domain: CONTOSO						
Additional Information: SAM Account Name: ServiceDesk SID History: -							
Log Name: Security Source: Microsoft Windows security Event ID: 4749 Level: Information User: N/A OpCode: Info More Information: Event Log Online							
Copy	Close						

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4749</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13827</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-14T16:16:35.568878700Z" />
<EventRecordID>172181</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1108" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />

```

</System>

```

- <EventData>
<Data Name="TargetUserName">ServiceDesk</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6119</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">ServiceDesk</Data>
<Data Name="SidHistory">-</Data>

```

</EventData>

</Event>

Required Server Roles: Active Directory domain controller.**Minimum OS Version:** Windows Server 2008.**Event Versions:** 0.**Field Descriptions:****Subject:**

- **Security ID** [Type = SID]: SID of account that requested the “create group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID** [Type = SID]: SID of created group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group that was created. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain name of created group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

Attributes:

- **SAM Account Name** [Type = UnicodeString]: This is a name of new group used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). The value of **sAMAccountName** attribute of new group object. For example: ServiceDesk
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sidHistory** property. This parameter contains the value of **sidHistory** attribute of new group object. This parameter might not be captured in the event, and in that case appears as “-”.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4749(S): A security-disabled global group was created.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor each time a new distribution group is created, to see who created the group and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
- If your organization has naming conventions for account names, monitor “**Attributes\SAM Account Name**” for names that don’t comply with the naming conventions.

4750(S): A security-disabled global group was changed.

Event Properties - Event 4750, Microsoft Windows security audit...

General **Details**

A security-disabled global group was changed.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x3007B

Group:

Security ID:	CONTOSO\ServiceDesk
Group Name:	ServiceDeskMain
Group Domain:	CONTOSO

Changed Attributes:

SAM Account Name:	ServiceDeskMain
SID History:	-

Event Description:

This event generates every time security-disabled (distribution) global group is changed.

This event generates only on domain controllers.

Some changes do not invoke a 4750 event, for example, changes made using the Active Directory Users and Computers management console in **Managed By** tab in group account properties.

If you change the name of the group (SAM Account Name), you also get “[4781: The name of an account was changed](#)” if “[Audit User Account Management](#)” subcategory success auditing is enabled.

If you change the group type, you get a change event from the new group type auditing subcategory instead of 4750. If you need to monitor for group type changes, it is better to monitor for “[4764: A group’s type was changed](#).” These events are generated for any group type when group type is changed. “[Audit Security Group Management](#)” subcategory success auditing must be enabled.

From 4750 event you can get information about changes of **sAMAccountName** and **sIDHistory** attributes or you will see that something changed, but will not be able to see what exactly changed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name: Security
 Source: Microsoft Windows sec
 Event ID: 4750
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Logged: 8/14/2015 9:38:37 A
 Task Category: Distribution Group M
 Keywords: Audit Success
 Computer: DC01.contoso.local

Copy **Close**

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4750</EventID>
<Version>0</Version>
<Level>0</Level>
```

```

<Task>13827</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-14T16:38:37.902710700Z" />
```

```
<EventRecordID>172188</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1108" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">ServiceDeskMain</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6119</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">ServiceDeskMain</Data>
<Data Name="SidHistory">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID** [Type = SID]: SID of changed group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.

Sometimes you can see the **Group\Security ID** field contains an old group name in Event Viewer (as you can see in the event example). That happens because Event Viewer caches names for SIDs that it has already resolved for the current session.

Security ID field has the same value as new group name (**Changed Attributes>SAM Account Name**). That happens because event is generated after name was changed and SID resolves to the new name. It is always better to use SID instead of group names for queries or filtering of events, because you will know for sure that this is the right object you are looking for or want to monitor.

- **Group Name** [Type = UnicodeString]: the name of the group that was changed. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain name of changed group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - [Built-in groups](#): Builtin

Changed Attributes:

If attribute was not changed it will have “-“ value.

You might see a 4750 event without any changes inside, that is, where all **Changed Attributes** appear as “-“. This usually happens when a change is made to an attribute that is not listed in the event. In this case there is no way to determine which attribute was changed. For example, this would happen if you change the **Description** of a group object using the Active Directory Users and Computers administrative console. Also, if the [discretionary access control list](#) (DACL) is changed, a 4750 event will generate, but all attributes will be “-“.

- **SAM Account Name** [Type = UnicodeString]: This is a new name of changed group used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). If the value of **sAMAccountName** attribute of group object was changed, you will see the new value here. For example: ServiceDesk.
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sidHistory** property. If the value of **sidHistory** attribute of group object was changed, you will see the new value here.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-“. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4750(S): A security-disabled global group was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical distribution groups in the organization, and need to specifically monitor these groups for any change, monitor events with the “**Group\Group Name**” values that correspond to the critical distribution groups.
- If you need to monitor each time a member is added to a distribution group, to see who added the member and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
- If your organization has naming conventions for account names, monitor “**Attributes\SAM Account Name**” for names that don’t comply with the naming conventions.

4751(S): A member was added to a security-disabled global group.



A member was added to a security-disabled global group.

Subject:
Security ID: CONTOSO\dadmin
Account Name: dadmin

Event Description:
This event generates every time a new member was added to a security-disabled (distribution) global group.
This event generates only on domain controllers.
For every added member you will get separate 4751 event.
You will typically see “[4750](#): A security-disabled global group was changed.” event without any changes in it prior to 4751 event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Member:
Security ID: CONTOSO\Auditor
Account Name: CN=Auditor,CN=Users,DC=contoso,DC=local

Group:
Security ID: CONTOSO\ServiceDesk
Group Name: ServiceDeskSecond
Group Domain: CONTOSO

Additional Information:
Privileges: -

Log Name: Security
Source: Microsoft Windows security
Event ID: 4751
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4751</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13827</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-15T00:01:10.821144700Z" />
<EventRecordID>172221</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1108" />
<Channel>Security</Channel>
```

<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="MemberName">CN=Auditor,CN=Users,DC=contoso,DC=local</Data>

```
<Data Name="MemberSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="TargetUserName">ServiceDeskSecond</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6119</Data>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “add member to the group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A security identifier (SID) is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “add member to the group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Member:

- **Security ID** [Type = SID]: SID of account that was added to the group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: distinguished name of account that was added to the group. For example: “CN=Auditor,CN=Users,DC=contoso,DC=local”. For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “-”.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

Group:

- **Security ID** [Type = SID]: SID of the group to which new member was added. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group to which new member was added. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain name of the group to which new member was added. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in "Table 8. User Privileges."

Security Monitoring Recommendations:

For 4751(S): A member was added to a security-disabled global group.

Type of monitoring required	Recommendation
Addition of members to distribution groups: You might need to monitor the addition of members to distribution groups.	If you need to monitor each time a member is added to a distribution group, to see who added the member and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
High-value distribution groups: You might have a list of critical distribution groups in the organization, and need to specifically monitor these groups for the addition of new members (or for other changes).	Monitor this event with the " Group\Group Name " values that correspond to the high-value distribution groups.
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the " Subject\Security ID " and " Member\Security ID " that correspond to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting	When you monitor for anomalies or malicious actions, use the " Subject\Security ID " (with

anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.

Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.

Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.

Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.

External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).

Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.

Account naming conventions: Your organization might have specific naming conventions for account names.

other information) to monitor how or when a particular account is being used.

Monitor this event with the “**Subject\Security ID**” and “**Member\Security ID**” that correspond to the accounts that should never be used.

If this event corresponds to a “whitelist-only” action, review the “**Subject\Security ID**” for accounts that are outside the whitelist.

If this event corresponds to an action you want to monitor for certain account types, review the “**Subject\Security ID**” to see whether the account type is as expected.

Monitor this event for the “**Subject\Account Domain**” corresponding to accounts from another domain or “external” accounts.

Monitor the target **Computer:** (or other target device) for actions performed by the “**Subject\Security ID**” that you are concerned about.

Monitor “**Subject\Account Name**” for names that don’t comply with naming conventions.

4752(S): A member was removed from a security-disabled global group.

Event Properties - Event 4752, Microsoft Windows security auditi... X

General Details

A member was removed from a security-disabled global group.

Account Domain:	CONTOSO
Logon ID:	0x3007B
Member:	
Security ID:	CONTOSO\Auditor
Account Name:	CN=Auditor,CN=Users,DC=contoso,DC=local
Group:	
Security ID:	CONTOSO\ServiceDesk
Group Name:	ServiceDeskSecond
Group Domain:	CONTOSO
Additional Information:	
Privileges:	-
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4752
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time member was removed from the security-disabled (distribution) global group.
 This event generates only on domain controllers.
 For every removed member you will get separate 4752 event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4752</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13827</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-15T00:20:57.315863900Z" />
<EventRecordID>172229</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1108" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```

</System>
- <EventData>
<Data Name="MemberName">CN=Auditor,CN=Users,DC=contoso,DC=local</Data>
<Data Name="MemberSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="TargetUserName">ServiceDeskSecond</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6119</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
```

</Event>

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “remove member from the group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “remove member from the group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Member:

- **Security ID** [Type = SID]: SID of account that was removed from the group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: distinguished name of account that was removed from the group. For example: “CN=Auditor,CN=Users,DC=contoso,DC=local”. For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “-”.

The LDAP API references an LDAP object by its **distinguished name (DN)**. A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

Group:

- **Security ID** [Type = SID]: SID of the group from which the member was removed. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group from which the member was removed. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain name of the group from which the member was removed. Formats vary, and include the following:

- Domain NETBIOS name example: CONTOSO
- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL
- [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4752(S): A member was removed from a security-disabled global group.

Type of monitoring required	Recommendation
Removal of members from distribution groups: You might need to monitor the removal of members from distribution groups.	If you need to monitor each time a member is removed from a distribution group, to see who removed the member and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
High-value distribution groups: You might have a list of critical distribution groups in the organization, and need to specifically monitor these groups for the removal of members (or for other changes).	Monitor this event with the “ Group\Group Name ” values that correspond to the high-value distribution groups.
Distribution groups with required members: You might need to ensure that for certain distribution groups, particular members are never removed.	Monitor this event with the “ Group\Group Name ” that corresponds to the group of interest, and the “ Member\Security ID ” of the members who should not be removed.
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Subject\Security ID ” and “ Member\Security ID ” that correspond to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Security ID ” and “ Member\Security ID ” that correspond to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist.

Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.

If this event corresponds to an action you want to monitor for certain account types, review the “**Subject\Security ID**” to see whether the account type is as expected.

External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).

Monitor this event for the “**Subject\Account Domain**” corresponding to accounts from another domain or “external” accounts.

Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.

Monitor the target **Computer**: (or other target device) for actions performed by the “**Subject\Security ID**” that you are concerned about.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Account Name**” for names that don’t comply with naming conventions.

4753(S): A security-disabled global group was deleted.

Event Properties - Event 4753, Microsoft Windows security audit... X

General		Details	
Security ID:	CONTOSO\dadmin	Group:	Security ID: CONTOSO\ServiceDesk
Account Name:	dadmin	Group Name:	ServiceDeskSecond
Account Domain:	CONTOSO	Group Domain:	CONTOSO
Logon ID:	0x3007B		
Additional Information:			
Privileges:	-		
Log Name:	Security	Source:	Microsoft Windows security
Event ID:	4753	Logged:	8/14/2015 5:59:33 PM
Level:	Information	Task Category:	Distribution Group M
User:	N/A	Keywords:	Audit Success
Opcode:	Info	Computer:	DC01.contoso.local
More Information: Event Log Online			
Copy		Close	

Event Description:

This event generates every time security-disabled (distribution) global group is deleted. This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4753</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13827</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-15T00:59:33.621155200Z" />
<EventRecordID>172230</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1504" />
<Channel>Security</Channel>
```

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">ServiceDeskSecond</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6119</Data>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3007b</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID** [Type = SID]: SID of deleted group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group that was deleted. For example: ServiceDesk

- **Group Domain** [Type = UnicodeString]: domain name of deleted group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4753(S): A security-disabled global group was deleted.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical distribution groups in the organization, and need to specifically monitor these groups for any change, especially group deletion, monitor events with the “**Group\Group Name**” values that correspond to the critical distribution groups.
- If you need to monitor each time a distribution group is deleted, to see who deleted it and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.

4759(S): A security-disabled universal group was created.

See event “[4749](#): A security-disabled global group was created.” Event 4759 is the same, but it is generated for a **universal** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4760(S): A security-disabled universal group was changed.

See event “[4750](#): A security-disabled global group was changed.” Event 4760 is the same, but it is generated for a **universal** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4761(S): A member was added to a security-disabled universal group.

See event “[4751](#): A member was added to a security-disabled global group.” Event 4761 is the same, but it is generated for a **universal** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4762(S): A member was removed from a security-disabled universal group.

See event “[4752](#): A member was removed from a security-disabled global group.” Event 4762 is the same, but it is generated for a **universal** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4763(S): A security-disabled universal group was deleted.

See event “[4753](#): A security-disabled global group was deleted.” Event 4763 is the same, but it is generated for a **universal** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4744(S): A security-disabled local group was created.

See event "[4749](#): A security-disabled global group was created." Event 4744 is the same, but it is generated for a **local** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4745(S): A security-disabled local group was changed.

See event "[4750](#): A security-disabled global group was changed." Event 4745 is the same, but it is generated for a **local** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4746(S): A member was added to a security-disabled local group.

See event "[4751](#): A member was added to a security-disabled global group." Event 4746 is the same, but it is generated for a **local** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4747(S): A member was removed from a security-disabled local group.

See event "[4752](#): A member was removed from a security-disabled global group." Event 4747 is the same, but it is generated for a **local** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

4748(S): A security-disabled local group was deleted.

See event "[4753](#): A security-disabled global group was deleted." Event 4748 is the same, but it is generated for a **local** distribution group instead of a **global** distribution group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Audit Other Account Management Events

Audit Other Account Management Events determines whether the operating system generates user account management audit events.

Event volume: Typically Low on all types of computers.

This subcategory allows you to audit next events:

- The password hash of a user account was accessed. This happens during an Active Directory Management Tool password migration.
- The Password Policy Checking API was called. Password Policy Checking API allows an application to check password compliance against an application-provided account database or single account and verify that passwords meet the complexity, aging, minimum length, and history reuse requirements of a password policy.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	Yes	No	Yes	No	The only reason to enable Success auditing on domain controllers is to monitor " 4782(S) : The password hash an account was accessed." This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	No	No	No	No	The only event which is generated on Member Servers is " 4793(S) : The Password Policy Checking API was called.", this event is a typical information event with little to no security relevance. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	No	No	No	No	The only event which is generated on Workstations is " 4793(S) : The Password Policy Checking API was called.", this event is a typical information event with little to no security relevance. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Events List:

- [4782\(S\)](#): The password hash an account was accessed.
- [4793\(S\)](#): The Password Policy Checking API was called.

4782(S): The password hash an account was accessed.

Event Properties - Event 4782, Microsoft Windows security audit... X

General	Details
<p>Security ID: SYSTEM Account Name: DC01\$ Account Domain: CONTOSO Logon ID: 0x3E7</p> <p>Target Account: Account Name: Andrei Account Domain: CONTOSO</p> <p>Log Name: Security Source: Microsoft Windows security audit Event ID: 4782 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p>	<p>Up</p> <p>Down</p> <p>Copy</p> <p>Close</p>

Event Description:

This event generates on domain controllers during password migration of an account using [Active Directory Migration Toolkit](#). Typically "**Subject\Security ID**" is the SYSTEM account.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4782</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13829</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-18T21:23:46.435367800Z" />
<EventRecordID>174829</EventRecordID>
```

<Correlation />

<Execution ProcessID="512" ThreadID="1232" />

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserName">Andrei</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">DC01$</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x3e7</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested hash migration operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested hash migration operation.
- **Account Domain** [Type = UnicodeString]: subject's domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For ANONYMOUS LOGON you will see **NT AUTHORITY** value for this field.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624: An account was successfully logged on](#).”

Target Account:

- **Account Name** [Type = UnicodeString]: the name of the account for which the password hash was migrated. For example: ServiceDesk
 - User account example: Andrei
 - Computer account example: DC01\$
- **Account Domain** [Type = UnicodeString]: domain name of the account for which the password hash was migrated. Formats vary, and include the following:

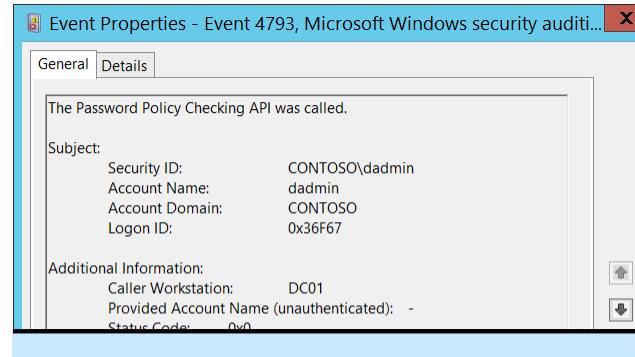
- Domain NETBIOS name example: FABRIKAM
- Lowercase full domain name: fabrikam.local
- Uppercase full domain name: FABRIKAM.LOCAL

Security Monitoring Recommendations:

For 4782(S): The password hash an account was accessed.

- Monitor for all events of this type, because any actions with account's password hashes should be planned. If this action was not planned, investigate the reason for the change.

4793(S): The Password Policy Checking API was called.

 Event Properties - Event 4793, Microsoft Windows security auditi... X

General **Details**

The Password Policy Checking API was called.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x36F67

Additional Information:

Caller Workstation:	DC01
Provided Account Name (unauthenticated):	-
Status Code:	0x0

Event Description:

This event generates each time the [Password Policy Checking API](#) is called.

The Password Policy Checking API allows an application to check password compliance against an application-provided account database or single account and verify that passwords meet the complexity, aging, minimum length, and history reuse requirements of a password policy.

This event, for example, generates during Directory Services Restore Mode ([DSRM](#)) account password reset procedure to check new DSRM password.

This event generates on the computer where Password Policy Checking API was called.

Note that starting with Microsoft SQL Server 2005, the "SQL Server password policy" feature can generate many 4793 events on a SQL Server.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Source: Microsoft Windows Security Auditing Logged: 8/17/2015 7:57:40 PM
 Event ID: 4793 Task Category: Other Account Management
 Level: Information Keywords: Audit Success
 User: N/A Computer: DC01.contoso.local
 OpCode: Info
 More Information: [Event Log Online](#)

Copy Close

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4793</EventID>
<Version>0</Version>
```

```

<Level>0</Level>
<Task>13829</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-18T02:37:46.322424300Z" />
<EventRecordID>172342</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="2964" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x36f67</Data>
<Data Name="Workstation">DC01</Data>
<Data Name="TargetUserName"></Data>
<Data Name="Status">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested Password Policy Checking API operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested Password Policy Checking API operation.
- **Account Domain** [Type = UnicodeString]: subject's domain name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Additional Information:

- **Caller Workstation** [Type = UnicodeString]: name of the computer from which the Password Policy Checking API was called. Typically, this is the same computer where this event was generated, for example, DC01. Computer name here does not contain \$ symbol at the end. It also can be an IP address or the DNS name of the computer.
- **Provided Account Name (unauthenticated)** [Type = UnicodeString]: the name of account, which password was provided/requested for validation. This parameter might not be captured in the event, and in that case appears as "-".
- **Status Code** [Type = HexInt32]: typically has "0x0" value. Status code is "0x0", no matter meets password domain Password Policy or not.

Security Monitoring Recommendations:

For 4793(S): The Password Policy Checking API was called.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Typically this is an informational event, and can give you information about when Password Policy Checking APIs were invoked, and who invoked them. The **Provided Account Name** does not always have a value—sometimes it's not really possible to determine for which account the password policy check was performed.

Audit Security Group Management

Audit Security Group Management determines whether the operating system generates audit events when specific security group management tasks are performed.

Event volume: Low.

This subcategory allows you to audit events generated by changes to security groups such as the following:

- Security group is created, changed, or deleted.
- Member is added or removed from a security group.
- Group type is changed.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	We recommend Success auditing of security groups, to see new group creation events, changes and deletion of critical groups. Also you will get information about new members of security groups, when a member was removed from a group and when security group membership was enumerated. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	Yes	No	Yes	No	We recommend Success auditing of security groups, to see new group creation events, changes and deletion of critical groups. Also you will get information about new members of security groups, when a member was removed from a group and when security group membership was enumerated. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	Yes	No	Yes	No	We recommend Success auditing of security groups, to see new group creation events, changes and deletion of critical groups. Also you will get information about new members of security groups, when a member was removed from a group and when security group membership was enumerated. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Events List:

- [4727\(S\)](#): A security-enabled global group was created.
- [4737\(S\)](#): A security-enabled global group was changed.
- [4728\(S\)](#): A member was added to a security-enabled global group.
- [4729\(S\)](#): A member was removed from a security-enabled global group.
- [4730\(S\)](#): A security-enabled global group was deleted.
- [4731\(S\)](#): A security-enabled local group was created.
- [4732\(S\)](#): A member was added to a security-enabled local group.
- [4733\(S\)](#): A member was removed from a security-enabled local group.
- [4734\(S\)](#): A security-enabled local group was deleted.
- [4735\(S\)](#): A security-enabled local group was changed.

- [4754\(S\)](#): A security-enabled universal group was created.
- [4755\(S\)](#): A security-enabled universal group was changed.
- [4756\(S\)](#): A member was added to a security-enabled universal group.
- [4757\(S\)](#): A member was removed from a security-enabled universal group.
- [4758\(S\)](#): A security-enabled universal group was deleted.
- [4764\(S\)](#): A group's type was changed.
- [4799\(S\)](#): A security-enabled local group membership was enumerated.

4727(S): A security-enabled global group was created.

See event "[4731](#): A security-enabled local group was created." Event 4727 is the same, but it is generated for a **global** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4731](#) do not apply.

4737(S): A security-enabled global group was changed.

See event "[4735](#): A security-enabled local group was changed." Event 4737 is the same, but it is generated for a **global** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4735](#) do not apply.

4728(S): A member was added to a security-enabled global group.

See event "[4732](#): A member was added to a security-enabled local group." Event 4728 is the same, but it is generated for a **global** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4732](#) do not apply.

4729(S): A member was removed from a security-enabled global group.

See event "[4733](#): A member was removed from a security-enabled local group." Event 4729 is the same, but it is generated for a **global** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4733](#) do not apply.

4730(S): A security-enabled global group was deleted.

See event "[4734](#): A security-enabled local group was deleted." Event 4730 is the same, but it is generated for a **global** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4734](#) do not apply.

4731(S): A security-enabled local group was created.

Event Properties - Event 4731, Microsoft Windows security audit... X

[General](#) [Details](#)

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x3031E
New Group:	
Security ID:	CONTOSO\AccountOperators
Group Name:	AccountOperators
Group Domain:	CONTOSO
Attributes:	
SAM Account Name:	AccountOperators
SID History:	-
Additional Information:	
Privileges:	-
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4731
Level:	Information
User:	N/A
OpCode:	Info
Computer:	
DC01.contoso.local	
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time a new security-enabled (security) local group was created. This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4731</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13826</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-19T01:01:50.646049700Z" />
<EventRecordID>174849</EventRecordID>
<Correlation />
<Execution ProcessID="512" ThreadID="1092" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```

</System>
- <EventData>
<Data Name="TargetUserName">AccountOperators</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6605</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3031e</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">AccountOperators</Data>
<Data Name="SidHistory">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “create group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

New Group:

- **Security ID** [Type = SID]: SID of created group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group that was created. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the created group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this new group belongs, for example: “Win81”.

Attributes:

- **SAM Account Name** [Type = UnicodeString]: This is a name of new group used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). The value of **sAMAccountName** attribute of new group object. For example: ServiceDesk. For local groups it is simply a name of new group.
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sidHistory** property. This parameter contains the value of **sidHistory** attribute of new group object. This parameter might not be captured in the event, and in that case appears as “-”. For local groups it is not applicable and always has “-” value.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”

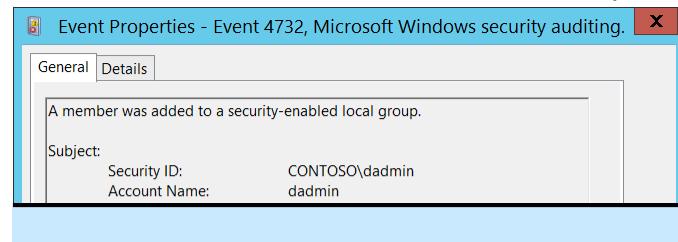
Security Monitoring Recommendations:

For 4731(S): A security-enabled local group was created.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor each time a new security group is created, to see who created the group and when, monitor this event.
- If you need to monitor the creation of local security groups on different servers, and you use Windows Event Forwarding to collect events in a central location, check “**New Group\Group Domain**.” It should not be the name of the domain, but instead should be the computer name.
- If your organization has naming conventions for account names, monitor “**Attributes\SAM Account Name**” for names that don’t comply with the naming conventions.

4732(S): A member was added to a security-enabled local group.

 Event Properties - Event 4732, Microsoft Windows security auditing.

General Details

A member was added to a security-enabled local group.

Subject: Security ID: CONTOSO\dadmin
Account Name: dadmin

Event Description:

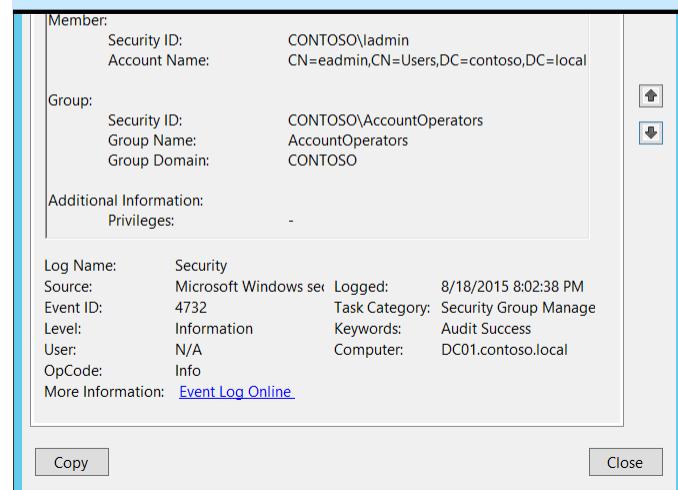
This event generates every time a new member was added to a security-enabled (security) local group.

This event generates on domain controllers, member servers, and workstations.

For every added member you will get separate 4732 event.

You will typically see “[4735: A security-enabled local group was changed.](#)” event without any changes in it prior to 4732 event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

 Event Properties - Event 4732, Microsoft Windows security auditing.

Member:	Security ID: CONTOSO\dadmin Account Name: CN=eadmin,CN=Users,DC=contoso,DC=local
Group:	Security ID: CONTOSO\AccountOperators Group Name: AccountOperators Group Domain: CONTOSO
Additional Information:	Privileges: -
Log Name: Security Source: Microsoft Windows security Event ID: 4732 Level: Information User: N/A OpCode: Info More Information: Event Log Online	Logged: 8/18/2015 8:02:38 PM Task Category: Security Group Management Keywords: Audit Success Computer: DC01.contoso.local

Copy **Close**

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4732</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13826</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-19T03:02:38.563110400Z" />
<EventRecordID>174856</EventRecordID>
<Correlation />
<Execution ProcessID="512" ThreadID="1092" />
```

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
```

```
- <EventData>
<Data Name="MemberName">CN=eadmin,CN=Users,DC=contoso,DC=local</Data>
<Data Name="MemberSid">S-1-5-21-3457937927-2839227994-823803824-500</Data>
<Data Name="TargetUserName">AccountOperators</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6605</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3031e</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “add member to the group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “add member to the group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Member:

- **Security ID** [Type = SID]: SID of account that was added to the group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.

- **Account Name** [Type = UnicodeString]: distinguished name of account that was added to the group. For example: "CN=Auditor,CN=Users,DC=contoso,DC=local". For local groups this field typically has “-” value, even if new member is a domain account. For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “-”.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

DC - domainComponent

CN - commonName

OU - organizationalUnitName

O - organizationName

Group:

- **Security ID** [Type = SID]: SID of the group to which new member was added. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group to which new member was added. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the group to which the new member was added. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this new group belongs, for example: “Win81”.
 - [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4732(S): A member was added to a security-enabled local group.

Type of monitoring required	Recommendation
Addition of members to local or domain security groups: You might need to monitor the addition of members to local or domain security groups.	If you need to monitor each time a member is added to a local or domain security group, to see who added the member and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
High-value local or domain security groups: You might have a list of critical local or domain security groups in the organization, and need to specifically monitor these groups for the addition of new members (or for other changes). Examples of critical local or domain groups are built-in local administrators group, domain admins, enterprise admins, and so on.	Monitor this event with the “ Group\Group Name ” values that correspond to the high-value local or domain security groups.

<p>High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.</p> <p>Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.</p>	Monitor this event with the “ Subject\Security ID ” and “ Member\Security ID ” that correspond to the high-value account or accounts.
<p>Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.</p>	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
<p>Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.</p>	Monitor this event with the “ Subject\Security ID ” and “ Member\Security ID ” that correspond to the accounts that should never be used.
<p>Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.</p>	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist.
<p>Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.</p>	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” to see whether the account type is as expected.
<p>External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).</p>	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
<p>Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.</p>	Monitor the target Computer: (or other target device) for actions performed by the “ Subject\Security ID ” that you are concerned about.
<p>Account naming conventions: Your organization might have specific naming conventions for account names.</p>	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.
<p>Mismatch between type of account (user or computer) and the group it was added to: You might want to monitor to ensure that a computer account was not added to a group intended for users, or a user account was not added to a group intended for computers.</p>	Monitor the type of account added to the group to see if it matches what the group is intended for.

4733(S): A member was removed from a security-enabled local group.

Event Properties - Event 4733, Microsoft Windows security auditing. X

General Details

A member was removed from a security-enabled local group.

Subject:
Security ID: CONTOSO\dadmin
Account Name: dadmin

Member:	CONTOSO\Auditor CN=Auditor,CN=Users,DC=contoso,DC=local
Group:	CONTOSO\AccountOperators AccountOperators Group Domain: CONTOSO
Additional Information:	Privileges: -

Log Name: Security
Source: Microsoft Windows security Log
Event ID: 4733
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 8/19/2015 9:51:00 AM
Task Category: Security Group Management
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Event Description:

This event generates every time member was removed from security-enabled (security) local group.

This event generates on domain controllers, member servers, and workstations.

For every removed member you will get separate 4733 event.

You will typically see "[4735](#): A security-enabled local group was changed." event without any changes in it prior to 4733 event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4733</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13826</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-19T16:51:00.376806500Z" />
<EventRecordID>175037</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1524" />
<Channel>Security</Channel>
```

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="MemberName">CN=Auditor,CN=Users,DC=contoso,DC=local</Data>
<Data Name="MemberSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="TargetUserName">AccountOperators</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6605</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x35e38</Data>
```

```
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “remove member from the group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “remove member from the group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Member:

- **Security ID** [Type = SID]: SID of account that was removed from the group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: distinguished name of account that was removed from the group. For example: “CN=Auditor,CN=Users,DC=contoso,DC=local”. For local groups this field typically has “-” value, even if removed member is a domain account. For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “-”.

The LDAP API references an LDAP object by its **distinguished name (DN)**. A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

Group:

- **Security ID** [Type = SID]: SID of the group from which the member was removed. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group from which the member was removed. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the group from which the member was removed. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this new group belongs, for example: "Win81".
 - Built-in groups: Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in "Table 8. User Privileges."

Security Monitoring Recommendations:

For 4733(S): A member was removed from a security-enabled local group.

Type of monitoring required	Recommendation
-----------------------------	----------------

Removal of members from local or domain security groups: You might need to monitor the removal of members from local or domain security groups.

If you need to monitor each time a member is removed from a local or domain security group, to see who added the member and when, monitor this event.

Typically, this event is used as an informational event, to be reviewed if needed.

High-value local or domain security groups: You might have a list of critical local or domain security groups in the organization, and need to specifically monitor these groups for the removal of members (or for other changes).

Examples of critical local or domain groups are built-in local administrators group, domain admins, enterprise admins, and so on.

Monitor this event with the “**Group\Group Name**” values that correspond to the high-value local or domain security groups.

Local or domain security groups with required members: You might need to ensure that for certain local or domain security groups, particular members are never removed.

Monitor this event with the “**Group\Group Name**” that corresponds to the group of interest, and the “**Member\Security ID**” of the members who should not be removed.

High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.

Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.

Monitor this event with the “**Subject\Security ID**” and “**Member\Security ID**” that correspond to the high-value account or accounts.

Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.

When you monitor for anomalies or malicious actions, use the “**Subject\Security ID**” (with other information) to monitor how or when a particular account is being used.

Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.

Monitor this event with the “**Subject\Security ID**” and “**Member\Security ID**” that correspond to the accounts that should never be used.

Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.

If this event corresponds to a “whitelist-only” action, review the “**Subject\Security ID**” for accounts that are outside the whitelist.

Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.

If this event corresponds to an action you want to monitor for certain account types, review the “**Subject\Security ID**” to see whether the account type is as expected.

External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).

Monitor this event for the “**Subject\Account Domain**” corresponding to accounts from another domain or “external” accounts.

Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.

Monitor the target **Computer**: (or other target device) for actions performed by the **"Subject\Security ID"** that you are concerned about.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor **"Subject\Account Name"** for names that don't comply with naming conventions.

4734(S): A security-enabled local group was deleted.

Event Properties - Event 4734, Microsoft Windows security audit...

General **Details**

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x35E38
Group:	
Security ID:	CONTOSO\AccountOperators
Group Name:	AccountOperators
Group Domain:	CONTOSO
Additional Information:	
Privileges:	-
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4734
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy **Close**

Event Description:

This event generates every time security-enabled (security) local group is deleted.

This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4734</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13826</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-19T18:23:42.426245700Z" />
<EventRecordID>175039</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1072" />
<Channel>Security</Channel>

```

```

<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">AccountOperators</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6605</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>

```

```
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x35e38</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID** [Type = SID]: SID of deleted group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group that was deleted. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the deleted group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this new group belongs, for example: “Win81”.
 - [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

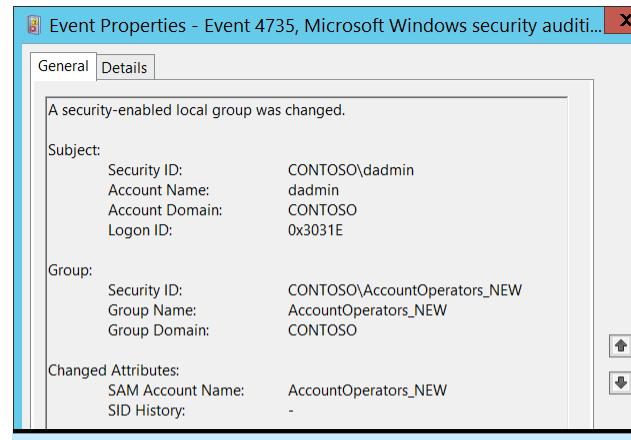
Security Monitoring Recommendations:

For 4734(S): A security-enabled local group was deleted.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical local or domain security groups in the organization, and need to specifically monitor these groups for any change, especially group deletion, monitor events with the “**Group\Group Name**” values that correspond to the critical local or domain security groups. Examples of critical local or domain groups are built-in local administrators group, domain admins, enterprise admins, and so on.
- If you need to monitor each time a local or domain security group is deleted, to see who deleted it and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.

4735(S): A security-enabled local group was changed.

 Event Properties - Event 4735, Microsoft Windows security audit...

Event Description:

This event generates every time a security-enabled (security) local group is changed.
 This event generates on domain controllers, member servers, and workstations.
 Some changes do not invoke a 4735 event, for example, changes made using Active Directory Users and Computers management console in **Managed By** tab in group account properties.
 If you change the name of the group (SAM Account Name), you also get “[4781: The name of an account was changed](#)” if “[Audit User Account Management](#)” subcategory success auditing is enabled.
 If you change the group type, you get a change event from the new group type auditing subcategory instead of 4735. If you need to monitor for group type changes, it is better to monitor for “[4764: A group's type was changed](#).” These events are generated for any group type when group type is changed. “[Audit Security Group Management](#)” subcategory success auditing must be enabled.
 From 4735 event you can get information about changes of **sAMAccountName** and **sIDHistory** attributes or you will see that something changed, but will not be able to see what exactly changed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4735
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

<Task>13826</Task>
<Opcode>0</Opcode>

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4735</EventID>
<Version>0</Version>
<Level>0</Level>
```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-19T02:00:45.537440000Z" />
<EventRecordID>174850</EventRecordID>
<Correlation />
<Execution ProcessID="512" ThreadID="1092" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserName">AccountOperators_NEW</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6605</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x3031e</Data>
  <Data Name="PrivilegeList"></Data>
  <Data Name="SamAccountName">AccountOperators_NEW</Data>
  <Data Name="SidHistory">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change group” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change group” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local

- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID** [Type = SID]: SID of changed group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.

Sometimes you can see the **Group\Security ID** field contains an old group name in Event Viewer (as you can see in the event example). That happens because Event Viewer caches names for SIDs that it has already resolved for the current session.

Security ID field has the same value as new group name (**Changed Attributes>SAM Account Name**). That is happens because event is generated after name was changed and SID resolves to the new name. It is always better to use SID instead of group names for queries or filtering of events, because you will know for sure that this the right object you are looking for or want to monitor.

- **Group Name** [Type = UnicodeString]: the name of the group that was changed. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the changed group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this new group belongs, for example: “Win81”.
 - [Built-in groups](#): Builtin

Changed Attributes:

If attribute was not changed it will have “-“ value.

You might see a 4735 event without any changes inside, that is, where all **Changed Attributes** appear as “-“. This usually happens when a change is made to an attribute that is not listed in the event. In this case there is no way to determine which attribute was changed. For example, this would happen if you change the **Description** of a group object using the Active Directory Users and Computers administrative console. Also, if the [discretionary access control list](#) (DACL) is changed, a 4735 event will generate, but all attributes will be “-“.

- **SAM Account Name** [Type = UnicodeString]: This is a new name of changed group used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). If the value of **sAMAccountName** attribute of group object was changed, you will see the new value here. For example: ServiceDesk. For local groups it is simply a new name of the group, if it was changed.
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sIDHistory** property. If the value of **sIDHistory** attribute of group object was changed, you will see the new value here. For local groups it is not applicable and always has “-“ value.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4735(S): A security-enabled local group was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical local or domain security groups in the organization, and need to specifically monitor these groups for any change, monitor events with the “**Group\Group Name**” values that correspond to the critical local or domain security groups.
- If you need to monitor each time a member is added to a local or domain security group, to see who added the member and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.
- If your organization has naming conventions for account names, monitor “**Attributes\SAM Account Name**” for names that don’t comply with the naming conventions.

4754(S): A security-enabled universal group was created.

See event “[4731](#): A security-enabled local group was created.”. Event 4754 is the same, but it is generated for a **universal** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4731](#) do not apply.

4755(S): A security-enabled universal group was changed.

See event “[4735](#): A security-enabled local group was changed.”. Event 4735 is the same, but it is generated for a **universal** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4735](#) do not apply.

4756(S): A member was added to a security-enabled universal group.

See event “[4732](#): A member was added to a security-enabled local group.”. Event 4756 is the same, but it is generated for a **universal** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4732](#) do not apply.

4757(S): A member was removed from a security-enabled universal group.

See event “[4733](#): A member was removed from a security-enabled local group.”. Event 4757 is the same, but it is generated for a **universal** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4733](#) do not apply.

4758(S): A security-enabled universal group was deleted.

See event “[4734](#): A security-enabled local group was deleted.”. Event 4758 is the same, but it is generated for a **universal** security group instead of a **local** security group. All event fields, XML, and recommendations are the same. The type of group is the only difference.

Important: this event generates only for domain groups, so the Local sections in event [4734](#) do not apply.

4764(S): A group's type was changed.

Event Properties - Event 4764, Microsoft Windows security auditing.

General **Details**

A group's type was changed.

Account Domain:	CONTOSO
Logon ID:	0x38200
Change Type:	Security Enabled Local Group Changed to Security Disabled Local Group.
Group:	Security ID: CONTOSO\CompanyAuditors Group Name: CompanyAuditors Group Domain: CONTOSO
Additional Information:	Privileges: -
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4764
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

Event Description:

This event generates every time group's type is changed.
 This event generates for both security and distribution groups.
 This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-
A5BA-3E3B0328C30D}" />
  <EventID>4764</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13826</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-20T00:25:33.459568000Z" />
  <EventRecordID>175221</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="1072" />

```

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="GroupTypeChange">Security Enabled Local Group Changed to Security Disabled Local Group.</Data>
  <Data Name="TargetUserName">CompanyAuditors</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6608</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x38200</Data>

```

```
<Data Name="PrivilegeList">-</Data>  
</EventData>  
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change group type” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change group type” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Change Type [Type = UnicodeString]: contains three parts: “<Param1> **Changed To** <Param2>.”. These two parameters can have the following values (they cannot have the same value at the same time):

- Security Disabled Local Group
- Security Disabled Universal Group
- Security Disabled Global Group
- Security Enabled Local Group
- Security Enabled Universal Group
- Security Enabled Global Group

Group:

- **Security ID** [Type = SID]: SID of changed group. Event Viewer automatically tries to resolve SIDs and show the group name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name** [Type = UnicodeString]: the name of the group, which type was changed. For example: ServiceDesk
- **Group Domain** [Type = UnicodeString]: domain or computer name of the changed group. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO

- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL
- For a local group, this field will contain the name of the computer to which this new group belongs, for example: "Win81".
- [Built-in groups](#): Builtin

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in "Table 8. User Privileges."

Security Monitoring Recommendations:

For 4764(S): A group's type was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical local or domain groups in the organization, and need to specifically monitor these groups for any change, especially group type change, monitor events with the "**Group\Group Name**" values that correspond to the critical distribution groups. Examples of critical local or domain groups are built-in local administrators group, domain admins, enterprise admins, critical distribution groups, and so on.
- If you need to monitor each time any group's type is changed, to see who changed it and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.

Event Properties - Event 4799, Microsoft Windows security auditing.

General Details

A security-enabled local group membership was enumerated.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x72D9D

Group Name: Administrators
Group Domain: Builtin

Process Information:

Process ID:	0xc80
Process Name:	C:\Windows\System32\mmc.exe

Log Name: Security
Source: Microsoft Windows security
Event ID: 4799
Level: Information
User: N/A
OpCode: Info

More Information: [Event Log Online Help](#)

Logged: 11/11/2015 7:50:23 PM
Task Category: Security Group Management
Keywords: Audit Success
Computer: WIN10-1.contoso.local

Copy Close

4799(S): A security-enabled local group membership was enumerated.

Event Description:

This event generates when a process enumerates the members of a security-enabled local group on the computer or device.

This event doesn't generate when group members were enumerated using Active Directory Users and Computers snap-in.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4799</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13826</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T03:50:23.625407600Z" />
```

© 2016 Microsoft. All rights reserved.

```
<EventRecordID>685</EventRecordID>
<Correlation ActivityID="{CBAEDE08-1CF0-0000-50DE-AECBF01CD101}" />
<Execution ProcessID="744" ThreadID="188" />
<Channel>Security</Channel>
<Computer>WIN10-1.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">Administrators</Data>
<Data Name="TargetDomainName">Builtin</Data>
<Data Name="TargetSid">S-1-5-32-544</Data>
<Data Name="SubjectUserSid">S-1-5-21-1377283216-344919071-3415362939-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x72d9d</Data>
<Data Name="CallerProcessId">0xc80</Data>
<Data Name="CallerProcessName">C:\Windows\System32\mmc.exe</Data>
</EventData>
</Event>
```

Required Server Roles: none.

Minimum OS Version: Windows Server 2016, Windows 10.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “enumerate security-enabled local group members” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enumerate security-enabled local group members” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

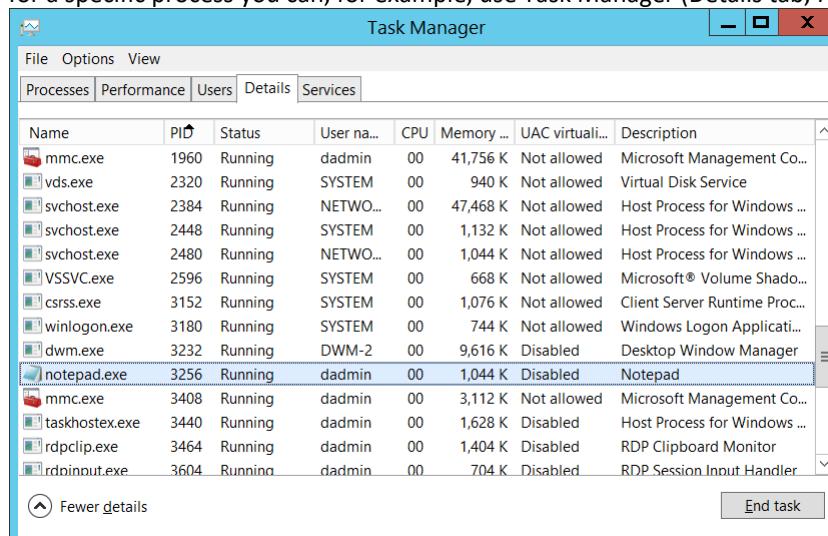
- **Logon ID [Type = HexInt64]:** hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Group:

- **Security ID [Type = SID]:** SID of the group which members were enumerated. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Group Name [Type = UnicodeString]:** the name of the group which members were enumerated.
- **Group Domain [Type = UnicodeString]: group's domain or computer name. Formats vary, and include the following:**
 - For Builtin groups this field has “Builtin” value.
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For a local group, this field will contain the name of the computer to which this group belongs, for example: “Win81”.

Process Information:

- **Process ID [Type = Pointer]:** hexadecimal Process ID of the process that enumerated the members of the group. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



Name	PID	Status	User na...	CPU	Memory ...	UAC virtuali...	Description
mmc.exe	1960	Running	dadmin	00	41,756 K	Not allowed	Microsoft Management Co...
vds.exe	2320	Running	SYSTEM	00	940 K	Not allowed	Virtual Disk Service
svchost.exe	2384	Running	NETWO...	00	47,468 K	Not allowed	Host Process for Windows ...
svchost.exe	2448	Running	SYSTEM	00	1,132 K	Not allowed	Host Process for Windows ...
svchost.exe	2480	Running	NETWO...	00	1,044 K	Not allowed	Host Process for Windows ...
VSSVC.exe	2596	Running	SYSTEM	00	668 K	Not allowed	Microsoft® Volume Shado...
csrss.exe	3152	Running	SYSTEM	00	1,076 K	Not allowed	Client Server Runtime Proc...
winlogon.exe	3180	Running	SYSTEM	00	744 K	Not allowed	Windows Logon Applicati...
dwm.exe	3232	Running	DWM-2	00	9,616 K	Disabled	Desktop Window Manager
notepad.exe	3256	Running	dadmin	00	1,044 K	Disabled	Notepad
mmc.exe	3408	Running	dadmin	00	3,112 K	Not allowed	Microsoft Management Co...
taskhostex.exe	3440	Running	dadmin	00	1,628 K	Disabled	Host Process for Windows ...
rdpclip.exe	3464	Running	dadmin	00	1,404 K	Disabled	RDP Clipboard Monitor
rdoimout.exe	3604	Running	dadmin	00	704 K	Disabled	RDP Session Input Handler

Fewer details End task

If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

- **Process Name [Type = UnicodeString]:** full path and the name of the executable for the process.

Security Monitoring Recommendations:

For 4799(S): A security-enabled local group membership was enumerated.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a list of critical local security groups in the organization, and need to specifically monitor these groups for any access (in this case, enumeration of group membership), monitor events with the “**Group\Group Name**” values that correspond to the critical local security groups. Examples of critical local groups are built-in local administrators, built-in backup operators, and so on.
- If you need to monitor each time the membership is enumerated for a local or domain security group, to see who enumerated the membership and when, monitor this event. Typically, this event is used as an informational event, to be reviewed if needed.

Audit User Account Management

Audit User Account Management determines whether the operating system generates audit events when specific user account management tasks are performed.

Event volume: Low.

This policy setting allows you to audit changes to user accounts. Events include the following:

- A user account is created, changed, deleted, renamed, disabled, enabled, locked out or unlocked.
- A user account's password is set or changed.
- A security identifier (SID) is added to the SID History of a user account, or fails to be added.
- The Directory Services Restore Mode password is configured.
- Permissions on administrative user accounts are changed.
- A user's local group membership was enumerated.
- Credential Manager credentials are backed up or restored.

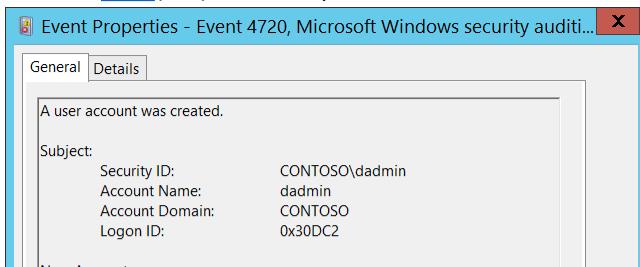
Some events in this subcategory, for example 4722, 4725, 4724, and 4781, are also generated for computer accounts.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	This subcategory contains many useful events for monitoring, especially for critical domain accounts, such as domain admins, service accounts, database admins, and so on. We recommend Failure auditing, mostly to see invalid password change and reset attempts for domain accounts, DSRM account password change failures, and failed SID History add attempts.
Member Server	Yes	Yes	Yes	Yes	We recommend monitoring all changes related to local user accounts, especially built-in local Administrator and other critical accounts. We recommend Failure auditing, mostly to see invalid password change and reset attempts for local accounts.
Workstation	Yes	Yes	Yes	Yes	We recommend monitoring all changes related to local user accounts, especially built-in local Administrator and other critical accounts. We recommend Failure auditing, mostly to see invalid password change and reset attempts for local accounts.

Events List:

- [4720\(S\)](#): A user account was created.
- [4722\(S\)](#): A user account was enabled.
- [4723\(S, F\)](#): An attempt was made to change an account's password.
- [4724\(S, F\)](#): An attempt was made to reset an account's password.
- [4725\(S\)](#): A user account was disabled.
- [4726\(S\)](#): A user account was deleted.
- [4738\(S\)](#): A user account was changed.
- [4740\(S\)](#): A user account was locked out.
- [4765\(S\)](#): SID History was added to an account.
- [4766\(F\)](#): An attempt to add SID History to an account failed.

- [4767\(S\)](#): A user account was unlocked.
- [4780\(S\)](#): The ACL was set on accounts which are members of administrators groups.
- [4781\(S\)](#): The name of an account was changed.
- [4794\(S, F\)](#): An attempt was made to set the Directory Services Restore Mode administrator password.



- [4798\(S\)](#): A user's local group membership was enumerated.
- [5376\(S\)](#): Credential Manager credentials were backed up.
- [5377\(S\)](#): Credential Manager credentials were restored from a backup.

4720(S): A user account was created.

Event Description:

This event generates every time a new user object is created.

This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

<p>Account Domain: CONTOSO</p> <p>Attributes:</p> <table border="0"> <tr><td>SAM Account Name:</td><td>ksmith</td></tr> <tr><td>Display Name:</td><td>Ken Smith</td></tr> <tr><td>User Principal Name:</td><td>ksmith@contoso.local</td></tr> <tr><td>Home Directory:</td><td>-</td></tr> <tr><td>Home Drive:</td><td>-</td></tr> <tr><td>Script Path:</td><td>-</td></tr> <tr><td>Profile Path:</td><td>-</td></tr> <tr><td>User Workstations:</td><td>-</td></tr> <tr><td>Password Last Set:</td><td><never></td></tr> <tr><td>Account Expires:</td><td><never></td></tr> <tr><td>Primary Group ID: 513</td><td></td></tr> <tr><td>Allowed To Delegate To:</td><td>-</td></tr> <tr><td>Old UAC Value:</td><td>0x0</td></tr> <tr><td>New UAC Value:</td><td>0x15</td></tr> <tr><td>User Account Control:</td><td></td></tr> <tr><td> Account Disabled</td><td></td></tr> <tr><td> 'Password Not Required' - Enabled</td><td></td></tr> <tr><td> 'Normal Account' - Enabled</td><td></td></tr> <tr><td>User Parameters:</td><td>-</td></tr> <tr><td>SID History:</td><td>-</td></tr> <tr><td>Logon Hours:</td><td><value not set></td></tr> </table> <p>Additional Information:</p> <table border="0"> <tr><td>Privileges</td><td>-</td></tr> </table> <p>Log Name: Security Source: Microsoft Windows sec... Event ID: 4720 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p>	SAM Account Name:	ksmith	Display Name:	Ken Smith	User Principal Name:	ksmith@contoso.local	Home Directory:	-	Home Drive:	-	Script Path:	-	Profile Path:	-	User Workstations:	-	Password Last Set:	<never>	Account Expires:	<never>	Primary Group ID: 513		Allowed To Delegate To:	-	Old UAC Value:	0x0	New UAC Value:	0x15	User Account Control:		Account Disabled		'Password Not Required' - Enabled		'Normal Account' - Enabled		User Parameters:	-	SID History:	-	Logon Hours:	<value not set>	Privileges	-	<p>Event XML:</p> <pre> - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>4720</EventID> <Version>0</Version> <Level>0</Level> <Task>13824</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-08-20T16:22:02.759912000Z" /> <EventRecordID>175408</EventRecordID> <Correlation /> <Execution ProcessID="520" ThreadID="1508" /> <Channel>Security</Channel> <Computer>DC01.contoso.local</Computer> <Security /> </System> - <EventData> <Data Name="TargetUserName">ksmith</Data> <Data Name="TargetDomainName">CONTOSO</Data> <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6609</Data> <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data> </pre>
SAM Account Name:	ksmith																																												
Display Name:	Ken Smith																																												
User Principal Name:	ksmith@contoso.local																																												
Home Directory:	-																																												
Home Drive:	-																																												
Script Path:	-																																												
Profile Path:	-																																												
User Workstations:	-																																												
Password Last Set:	<never>																																												
Account Expires:	<never>																																												
Primary Group ID: 513																																													
Allowed To Delegate To:	-																																												
Old UAC Value:	0x0																																												
New UAC Value:	0x15																																												
User Account Control:																																													
Account Disabled																																													
'Password Not Required' - Enabled																																													
'Normal Account' - Enabled																																													
User Parameters:	-																																												
SID History:	-																																												
Logon Hours:	<value not set>																																												
Privileges	-																																												

```
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30dc2</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">ksmith</Data>
<Data Name="DisplayName">Ken Smith</Data>
<Data Name="UserPrincipalName">ksmith@contoso.local</Data>
<Data Name="HomeDirectory">-</Data>
<Data Name="HomePath">-</Data>
<Data Name="ScriptPath">-</Data>
<Data Name="ProfilePath"></Data>
<Data Name="UserWorkstations">-</Data>
<Data Name="PasswordLastSet">%1794</Data>
<Data Name="AccountExpires">%1794</Data>
<Data Name="PrimaryGroupId">513</Data>
<Data Name="AllowedToDelegateTo">-</Data>
<Data Name="OldUacValue">0x0</Data>
<Data Name="NewUacValue">0x15</Data>
<Data Name="UserAccountControl">%2080 %2082 %2084</Data>
<Data Name="UserParameters">-</Data>
<Data Name="SidHistory">-</Data>
<Data Name="LogonHours">%1793</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “create user account” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create user account” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:

- Domain NETBIOS name example: CONTOSO
- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

New Account:

- **Security ID** [Type = SID]: SID of created user account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the user account that was created. For example: dadmin.
- **Account Domain** [Type = UnicodeString]: domain name of created user account. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local accounts, this field will contain the name of the computer to which this new account belongs, for example: “Win81”.

Attributes:

- **SAM Account Name** [Type = UnicodeString]: logon name for account used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). The value of **sAMAccountName** attribute of new user object. For example: ksmith. For local account this field contains the name of new user account.
- **Display Name** [Type = UnicodeString]: the value of **displayName** attribute of new user object. It is a name displayed in the address book for a particular account .This is usually the combination of the user's first name, middle initial, and last name. For example, Ken Smith. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. Local accounts contain **Full Name** attribute in this field, but for new local accounts this field typically has value “<value not set>”.
- **User Principal Name** [Type = UnicodeString]: internet-style login name for the account, based on the Internet standard RFC 822. By convention this should map to the account's email name. This parameter contains the value of **userPrincipalName** attribute of new user object. For example, ksmith@contoso.local. For local users this field is not applicable and has value “-”. You can change this attribute by using Active Directory Users and Computers, or through a script, for example.
- **Home Directory** [Type = UnicodeString]: user's home directory. If **homeDrive** attribute is set and specifies a drive letter, **homeDirectory** should be a UNC path. The path must be a network UNC of the form \\Server\Share\Directory. This parameter contains the value of **homeDirectory** attribute of new user object. For new local accounts this field typically has value “<value not set>”. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”.
- **Home Drive** [Type = UnicodeString]: specifies the drive letter to which to map the UNC path specified by **homeDirectory** account's attribute. The drive letter must be specified in the form “DRIVE LETTER:”. For example – “H:”. This parameter contains the value of **homeDrive** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”. For new local accounts this field typically has value “<value not set>”.
- **Script Path** [Type = UnicodeString]: specifies the path of the account's logon script. This parameter contains the value of **scriptPath** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”. For new local accounts this field typically has value “<value not set>”.

- **Profile Path** [Type = UnicodeString]: specifies a path to the account's profile. This value can be a null string, a local absolute path, or a UNC path. This parameter contains the value of **profilePath** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”. For new local accounts this field typically has value “<value not set>”.
- **User Workstations** [Type = UnicodeString]: contains the list of NetBIOS or DNS names of the computers from which the user can logon. Each computer name is separated by a comma. The name of a computer is the **sAMAccountName** property of a user object. This parameter contains the value of **userWorkstations** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”. For local users this field is not applicable and typically has value “<value not set>”.
- **Password Last Set** [Type = UnicodeString]: last time the account's password was modified. For manually created user account, using Active Directory Users and Computers snap-in, this field typically has value “<never>”. This parameter contains the value of **pwdLastSet** attribute of new user object.
- **Account Expires** [Type = UnicodeString]: the date when the account expires. This parameter contains the value of **accountExpires** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. This parameter might not be captured in the event, and in that case appears as “-”. For manually created local and domain user accounts this field typically has value “<never>”.
- **Primary Group ID** [Type = UnicodeString]: Relative Identifier (RID) of user's object primary group.

Relative identifier (RID) is a variable length number that is assigned to objects at creation and becomes part of the object's Security Identifier (SID) that uniquely identifies an account or group within a domain.

Typically, **Primary Group** field for new user accounts has the following values:

- 513 (Domain Users. For local accounts this RID means Users) – for domain and local users.

See this article <https://support.microsoft.com/en-us/kb/243330> for more information. This parameter contains the value of **primaryGroupID** attribute of new user object.

- **Allowed To Delegate To** [Type = UnicodeString]: the list of SPNs to which this account can present delegated credentials. Can be changed using Active Directory Users and Computers management console in **Delegation** tab of user account, if this account has at least one SPN registered. This parameter contains the value of **AllowedToDelegateTo** attribute of new user object. For local user accounts this field is not applicable and typically has value “-”. For new domain user accounts it is typically has value “-”. See description of **AllowedToDelegateTo** field for “[4738\(S\): A user account was changed.](#)” event for more details.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Old UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user account. **Old UAC value always “0x0”** for new user accounts. This parameter contains the previous value of **userAccountControl** attribute of user object.
- **New UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user account. This parameter contains the value of **userAccountControl** attribute of new user object.

To decode this value, you can go through the property value definitions in the “Table 7. User's or Computer's account UAC flags.” from largest to smallest. Compare each property value to the flags value in the event. If the flags value in the event is greater than or equal to the property value, then the property is “set” and applies to that event. Subtract the property value from the flags value in the event and note that the flag applies and then go on to the next flag.

Here's an example: Flags value from event: 0x15

Decoding:

- PASSWD_NOTREQD 0x0020
- LOCKOUT 0x0010
- HOMEDIR_REQUIRED 0x0008
- (undeclared) 0x0004
- ACCOUNTDISABLE 0x0002
- SCRIPT 0x0001

0x0020 > 0x15, so PASSWD_NOTREQD does not apply to this event

0x10 < 0x15, so LOCKOUT applies to this event. 0x15 - 0x10 = 0x5

0x4 < 0x5, so the undeclared value is set. We'll pretend it doesn't mean anything. 0x5 - 0x4 = 0x1

0x2 > 0x1, so ACCOUNTDISABLE does not apply to this event

0x1 = 0x1, so SCRIPT applies to this event. 0x1 - 0x1 = 0x0, we're done.

So this UAC flags value decodes to: LOCKOUT and SCRIPT

- User Account Control** [Type = UnicodeString]: shows the list of changes in **userAccountControl** attribute. You will see a line of text for each change. For new user accounts, when the object for this account was created, the **userAccountControl** value was considered to be “**0x0**”, and then it was changed from “**0x0**” to the real value for the account’s **userAccountControl** attribute. See possible values in the table below. In the “User Account Control field text” column, you can see the text that will be displayed in the **User Account Control** field in 4720 event.

Flag Name	userAccount Control in hexadecimal	userAccount Control in decimal	Description	User Account Control field text
SCRIPT	0x0001	1	The logon script will be run.	Changes of this flag do not show in 4720 events.
ACCOUNTDISABLE	0x0002	2	The user account is disabled.	Account Disabled Account Enabled
Undeclared	0x0004	4	This flag is undeclared.	Changes of this flag do not show in 4720 events.
HOMEDIR_REQUIRED	0x0008	8	The home folder is required.	'Home Directory Required' - Enabled 'Home Directory Required' - Disabled
LOCKOUT	0x0010	16		Changes of this flag do not show in 4720 events.
PASSWD_NOTREQD	0x0020	32	No password is required.	'Password Not Required' - Enabled 'Password Not Required' - Disabled
PASSWD_CANT_CHANGE	0x0040	64	The user cannot change the password. This is a permission on the user's object.	Changes of this flag do not show in 4720 events.
ENCRYPTED_TEXT_PWD_ALLOWED	0x0080	128	The user can send an encrypted password. Can be set using “Store password using reversible encryption” checkbox.	'Encrypted Text Password Allowed' - Disabled 'Encrypted Text Password Allowed' - Enabled
TEMP_DUPLICATE_ACCOUNT	0x0100	256	This is an account for users whose primary account is in another domain. This account provides user access to this domain, but not to any domain that	Cannot be set for computer account.

			trusts this domain. This is sometimes referred to as a local user account.	
NORMAL_ACCOUNT	0x0200	512	This is a default account type that represents a typical user.	'Normal Account' - Disabled 'Normal Account' - Enabled
INTERDOMAIN_TRUST_ACCOUNT	0x0800	2048	This is a permit to trust an account for a system domain that trusts other domains.	Cannot be set for computer account.
WORKSTATION_TRUST_ACCOUNT	0x1000	4096	This is a computer account for a computer that is running Microsoft Windows NT 4.0 Workstation, Microsoft Windows NT 4.0 Server, Microsoft Windows 2000 Professional, or Windows 2000 Server and is a member of this domain.	'Workstation Trust Account' - Disabled 'Workstation Trust Account' - Enabled
SERVER_TRUST_ACCOUNT	0x2000	8192	This is a computer account for a domain controller that is a member of this domain.	'Server Trust Account' - Enabled 'Server Trust Account' - Disabled
DONT_EXPIRE_PASSWORD	0x10000	65536	Represents the password, which should never expire on the account. Can be set using "Password never expires" checkbox.	'Don't Expire Password' - Disabled 'Don't Expire Password' - Enabled
MNS_LOGON_ACCOUNT	0x20000	131072	This is an MNS logon account.	'MNS Logon Account' - Disabled 'MNS Logon Account' - Enabled
SMARTCARD_REQUIRED	0x40000	262144	When this flag is set, it forces the user to log on by using a smart card.	'Smartcard Required' - Disabled 'Smartcard Required' - Enabled
TRUSTED_FOR_DELEGATION	0x80000	524288	When this flag is set, the service account (the user or computer account) under which a service runs is trusted for Kerberos delegation. Any such service can impersonate a client requesting the service. To enable a service for Kerberos delegation, you must set this flag on the userAccountControl property of the service account. If you enable Kerberos constraint or unconstraint delegation or disable these types of delegation in Delegation tab you will get this flag changed.	'Trusted For Delegation' - Enabled 'Trusted For Delegation' - Disabled
NOT_DELEGATED	0x100000	1048576	When this flag is set, the security context of the user is not delegated to a service even if the service account is set as trusted for Kerberos delegation. Can be set using "Account is sensitive and cannot be delegated" checkbox.	'Not Delegated' - Disabled 'Not Delegated' - Enabled
USE DES_KEY_ONLY	0x200000	2097152	Restrict this principal to use only Data Encryption Standard (DES) encryption types for keys.	'Use DES Key Only' - Disabled 'Use DES Key Only' - Enabled

			Can be set using "Use Kerberos DES encryption types for this account" checkbox.	
DONT_REQ_PREAUTH	0x4000000	4194304	This account does not require Kerberos pre-authentication for logging on. Can be set using "Do not require Kerberos preauthentication" checkbox.	'Don't Require Preauth' - Disabled 'Don't Require Preauth' - Enabled
PASSWORD_EXPIRED	0x8000000	8388608	The user's password has expired.	Changes of this flag do not show in 4720 events.
TRUSTED_TO_AUTH_FOR_DELEGATION	0x10000000	16777216	The account is enabled for delegation. This is a security-sensitive setting. Accounts that have this option enabled should be tightly controlled. This setting lets a service that runs under the account assume a client's identity and authenticate as that user to other remote servers on the network. If you enable Kerberos protocol transition delegation or disable this type of delegation in Delegation tab you will get this flag changed.	'Trusted To Authenticate For Delegation' - Disabled 'Trusted To Authenticate For Delegation' - Enabled
PARTIAL_SECRETS_ACCOUNT	0x040000000	67108864	The account is a read-only domain controller (RODC). This is a security-sensitive setting. Removing this setting from an RODC compromises security on that server.	No information.

For new, manually created, domain or local user accounts typical flags are:

- Account Disabled
- 'Password Not Required' - Enabled
- 'Normal Account' – Enabled

After new user creation event you will typically see couple of "[4738: A user account was changed.](#)" events with new flags:

- 'Password Not Required' – Disabled
- Account Enabled

- **User Parameters** [Type = UnicodeString]: if you change any setting using Active Directory Users and Computers management console in Dial-in tab of user's account properties, then you will see <value changed, but not displayed> in this field in "[4738: A user account was changed.](#)" This parameter might not be captured in the event, and in that case appears as "-". For new local accounts this field typically has value "<value not set>".
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sIDHistory** property. This parameter contains the value of **sIDHistory** attribute of new user object. This parameter might not be captured in the event, and in that case appears as "-".
- **Logon Hours** [Type = UnicodeString]: hours that the account is allowed to logon to the domain. The value of **logonHours** attribute of new user object. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. You will typically see "<value not set>" value for new manually created user accounts in event 4720. For new local accounts this field is not applicable and typically has value "All".

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4720(S): A user account was created.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Some organizations monitor every [4720](#) event.
- Consider whether to track the following fields and values:

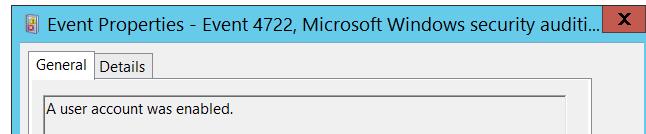
Field and value to track	Reason to track
SAM Account Name is empty or -	This field must contain the user account name. If it is empty or -, it might indicate an anomaly.
User Principal Name is empty or -	Typically this field should not be empty for new user accounts. If it is empty or -, it might indicate an anomaly.
Home Directory is not - Home Drive is not - Script Path is not - Profile Path is not - User Workstations is not -	Typically these fields are - for new user accounts. Other values might indicate an anomaly and should be monitored. For local accounts these fields should display <value not set>.
Password Last Set is <never>	This typically means this is a manually created user account, which you might need to monitor.
Password Last Set is a time in the future	This might indicate an anomaly.
Account Expires is not <never>	Typically this field is <never> for new user accounts. Other values might indicate an anomaly and should be monitored.
Primary Group ID is not 513	Typically, the Primary Group value is 513 for domain and local users. Other values should be monitored.
Allowed To Delegate To is not -	Typically this field is - for new user accounts. Other values might indicate an anomaly and should be monitored.
Old UAC Value is not 0x0	Typically this field is 0x0 for new user accounts. Other values might indicate an anomaly and should be monitored.
SID History is not -	This field will always be set to - unless the account was migrated from another domain.
Logon Hours value other than <value not set> or “All”	This should always be <value not set> for new domain user accounts, and “All” for new local user accounts.

- Consider whether to track the following user account control flags:

User account control flag to track	Information about the flag
'Normal Account' – Disabled	Should not be disabled for user accounts.
'Encrypted Text Password Allowed' – Enabled	By default, these flags should not be enabled for new user accounts created with the “Active Directory Users and Computers” snap-in.

'Smartcard Required' – Enabled	
'Not Delegated' – Enabled	
'Use DES Key Only' – Enabled	
'Don't Require Preauth' – Enabled	
'Trusted To Authenticate For Delegation' – Enabled	
'Server Trust Account' – Enabled	Should never be enabled for user accounts. Applies only to domain controller (computer) accounts.
'Don't Expire Password' – Enabled	Should be monitored for critical accounts, or all accounts if your organization does not allow this flag. By default, this flag should not be enabled for new user accounts created with the "Active Directory Users and Computers" snap-in.
'Trusted For Delegation' – Enabled	By default, this flag should not be enabled for new user accounts created with the "Active Directory Users and Computers" snap-in. It is enabled by default only for new domain controllers.

4722(S): A user account was enabled.

 Event Properties - Event 4722, Microsoft Windows security audit... X

General Details

A user account was enabled.

Account Domain:	CONTOSO
Logon ID:	0x30D5F
Target Account:	
Security ID:	CONTOSO\Auditor
Account Name:	Auditor
Account Domain:	CONTOSO
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4722
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time user or computer object is enabled.

For user accounts, this event generates on domain controllers, member servers, and workstations.

For computer accounts, this event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4722</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-21T23:55:11.038308600Z" />
<EventRecordID>175716</EventRecordID>
```

<Correlation />
<Execution ProcessID="520" ThreadID="1112" />

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserName">Auditor</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
  <Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x30d5f</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “enable account” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enable account” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

- **Security ID** [Type = SID]: SID of account that was enabled. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

- **Account Name** [Type = UnicodeString]: the name of the account that was enabled.
- **Account Domain** [Type = UnicodeString]: target account's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

Security Monitoring Recommendations:

For 4722(S): A user account was enabled.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a high-value domain or local account for which you need to monitor every change, monitor all [4722](#) events with the “**Target Account\Security ID**” that corresponds to the account.
- If you have domain or local accounts that should never be enabled, you can monitor all [4722](#) events with the “**Target Account\Security ID**” fields that correspond to the accounts.
- We recommend monitoring all [4722](#) events for local accounts, because these accounts usually do not change often. This is especially relevant for critical servers, administrative workstations, and other high value assets.

4723(S, F): An attempt was made to change an account's password.

 Event Properties - Event 4723, Microsoft Windows security audit...

General **Details**

An attempt was made to change an account's password.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x1A9B76

Target Account:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO

Event Description:

This event generates every time a user attempts to change his or her password.

For user accounts, this event generates on domain controllers, member servers, and workstations.

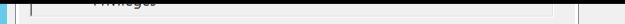
For domain accounts, a Failure event generates if new password fails to meet the password policy.

For local accounts, a Failure event generates if new password fails to meet the password policy or old password is wrong.

For domain accounts if old password was wrong, then “[4771: Kerberos pre-authentication failed](#)” or “[4776: The computer attempted to validate the credentials for an account](#)” will be generated on domain controller if specific subcategories were enabled on it.

Typically you will see 4723 events with the same **Subject\Security ID** and **Target Account\Security ID** fields, which is normal behavior.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

 Event Properties - Event 4723, Microsoft Windows security audit...

Properties

Log Name:	Security
Source:	Microsoft Windows se
Event ID:	4723
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4723</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>

```

```
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T01:32:51.494558000Z" />
<EventRecordID>175722</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1112" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserName">dadmin</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x1a9b76</Data>
  <Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made an attempt to change Target's Account password. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to change Target's Account password.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account: account for which the password change was requested.

- **Security ID** [Type = SID]: SID of account for which the password change was requested. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account for which the password change was requested.
- **Account Domain** [Type = UnicodeString]: target account’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4723(S, F): An attempt was made to change an account's password.

Appendix A: Security monitoring recommendations for many audit events

Event Properties - Event 4724, Microsoft Windows security audit...

General		Details	
An attempt was made to reset an account's password.			
Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x30D5F			
Target Account: Security ID: CONTOSO\User1 Account Name: User1 Account Domain: CONTOSO			
Log Name: Security Source: Microsoft Windows sec... Event ID: 4724 Level: Information User: N/A OpCode: Info More Information: Event Log Online		Logged: 8/21/2015 6:58:21 PM Task Category: User Account Mana... Keywords: Audit Success Computer: DC01.contoso.local	
<input type="button" value="Copy"/>		<input type="button" value="Close"/>	

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a high-value domain or local user account for which you need to monitor every password change attempt, monitor all [4723](#) events with the “**Target Account\Security ID**” that corresponds to the account.
- If you have a high-value domain or local account for which you need to monitor every change, monitor all [4723](#) events with the “**Target Account\Security ID**” that corresponds to the account.
- If you have domain or local accounts for which the password should never be changed, you can monitor all [4723](#) events with the “**Target Account\Security ID**” that corresponds to the account.

4724(S, F): An attempt was made to reset an account's password.

Event Description:

This event generates every time an account attempted to reset the password for another account. For user accounts, this event generates on domain controllers, member servers, and workstations. For domain accounts, a Failure event generates if the new password fails to meet the password policy. A Failure event does NOT generate if user gets “Access Denied” while doing the password reset procedure. This event also generates if a computer account reset procedure was performed. For local accounts, a Failure event generates if the new password fails to meet the local password policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4724</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T01:58:21.725864900Z" />
<EventRecordID>175740</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="548" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">User1</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-1107</Data>
<Data Name="SubjectUserId" S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d5f</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that made an attempt to reset Target's Account password. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to reset Target's Account password.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Target Account: account for which password reset was requested.

- **Security ID** [Type = SID]: SID of account for which password reset was requested. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account for which password reset was requested.
- **Account Domain** [Type = UnicodeString]: target account's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

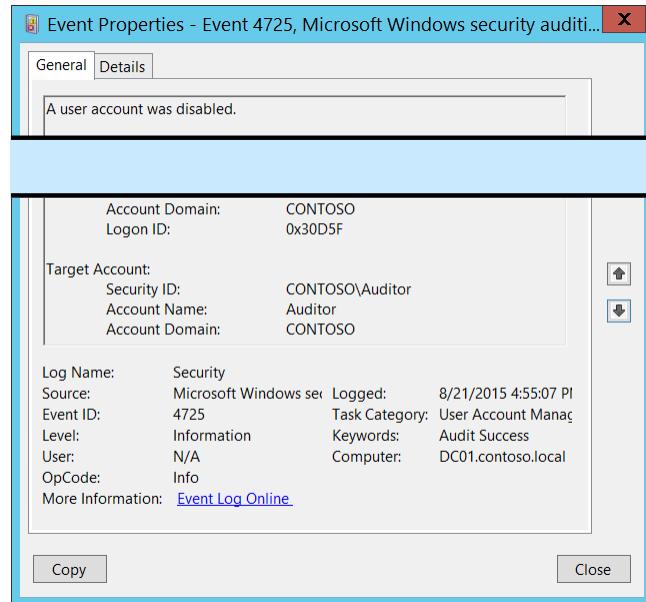
Security Monitoring Recommendations:

For 4724(S, F): An attempt was made to reset an account's password.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a high-value domain or local user account for which you need to monitor every password reset attempt, monitor all [4724](#) events with the "**Target Account\Security ID**" that corresponds to the account.
- If you have a high-value domain or local account for which you need to monitor every change, monitor all [4724](#) events with the "**Target Account\Security ID**" that corresponds to the account.
- If you have domain or local accounts for which the password should never be reset, you can monitor all [4724](#) events with the "**Target Account\Security ID**" that corresponds to the account.
- We recommend monitoring all [4724](#) events for local accounts, because their passwords usually do not change often. This is especially relevant for critical servers, administrative workstations, and other high value assets.

4725(S): A user account was disabled.

 Event Properties - Event 4725, Microsoft Windows security audit...

General Details

A user account was disabled.

Account Domain:	CONTOSO
Logon ID:	0x30D5F
Target Account:	
Security ID:	CONTOSO\Auditor
Account Name:	Auditor
Account Domain:	CONTOSO
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4725
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy **Close**

Event Description:

This event generates every time user or computer object is disabled.
 For user accounts, this event generates on domain controllers, member servers, and workstations.
 For computer accounts, this event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4725</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-21T23:55:07.657358900Z" />
<EventRecordID>175714</EventRecordID>
```

```
<Correlation />
<Execution ProcessID="520" ThreadID="1112" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">Auditor</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d5f</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.**Field Descriptions:****Subject:**

- **Security ID** [Type = SID]: SID of account that requested the “disable account” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “disable account” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

- **Security ID** [Type = SID]: SID of account that was disabled. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that was disabled.
- **Account Domain** [Type = UnicodeString]: target account’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

Security Monitoring Recommendations:

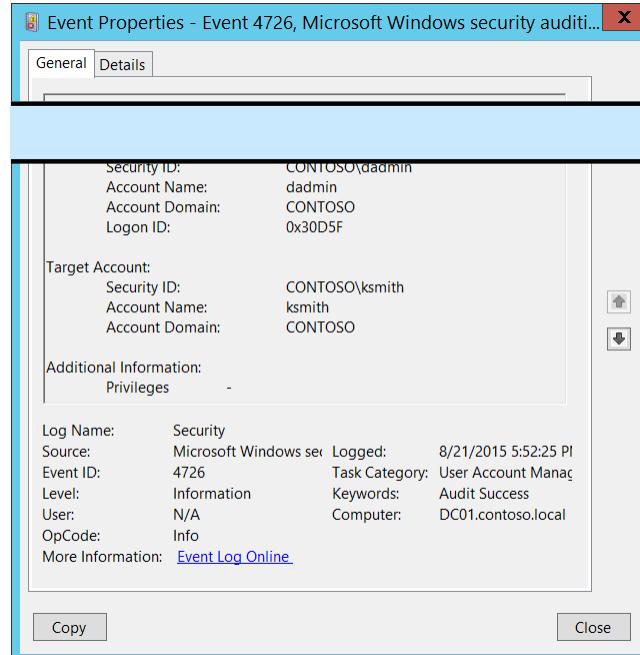
For 4725(S): A user account was disabled.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a high-value domain or local account for which you need to monitor every change, monitor all [4725](#) events with the “**Target Account\Security ID**” that corresponds to the account.
- If you have domain or local accounts that should never be disabled (for example, service accounts), you can monitor all [4725](#) events with the “**Target Account\Security ID**” that corresponds to the account.

- We recommend monitoring all [4725](#) events for local accounts, because these accounts usually do not change often. This is especially relevant for critical servers, administrative workstations, and other high value assets.

4726(S): A user account was deleted.

 Event Properties - Event 4726, Microsoft Windows security audit... X

<input type="checkbox"/> General	<input type="checkbox"/> Details
----------------------------------	----------------------------------

Event Description:
This event generates every time user object was deleted.
This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4726</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T00:52:25.104613800Z" />
<EventRecordID>175720</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1112" />
<Channel>Security</Channel>

```

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4726
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Security ID: CONTOSO\adadmin
Account Name: adadmin
Account Domain: CONTOSO
Logon ID: 0x30D5F

Target Account:
Security ID: CONTOSO\ksmith
Account Name: ksmith
Account Domain: CONTOSO

Additional Information:
Privileges: -

Copy Close

```

<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">ksmith</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6609</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d5f</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
```

</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete user account” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete user account” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

- **Security ID** [Type = SID]: SID of account that was deleted. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that was deleted.
- **Account Domain** [Type = UnicodeString]: target account’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

Additional Information:

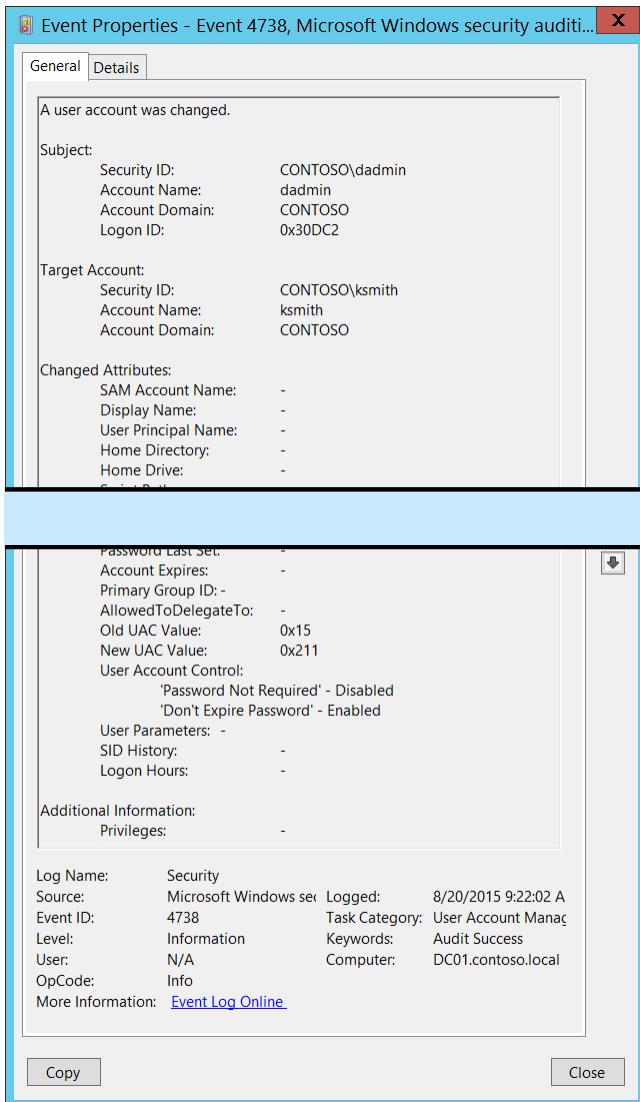
- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “–”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4726(S): A user account was deleted.

Appendix A: Security monitoring recommendations for many audit events

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a high-value domain or local account for which you need to monitor every change (or deletion), monitor all [4726](#) events with the “**Target Account\Security ID**” that corresponds to the account.

 Event Properties - Event 4738, Microsoft Windows security audit... X

General Details

A user account was changed.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x30DC2

Target Account:

Security ID:	CONTOSO\ksmith
Account Name:	ksmith
Account Domain:	CONTOSO

Changed Attributes:

SAM Account Name:	-
Display Name:	-
User Principal Name:	-
Home Directory:	-
Home Drive:	-
Script Path:	-

Additional Information:

Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4738
Level:	Information
User:	N/A
OpCode:	Info

Logged: 8/20/2015 9:22:02 A

Task Category: User Account Management

Keywords: Audit Success

Computer: DC01.contoso.local

More Information: [Event Log Online](#)

- If you have a domain or local account that should never be deleted (for example, service accounts), monitor all [4726](#) events with the “**Target Account\Security ID**” that corresponds to the account.
- We recommend monitoring all [4726](#) events for local accounts, because these accounts typically are not deleted often. This is especially relevant for critical servers, administrative workstations, and other high value assets.

4738(S): A user account was changed.

Event Description:

This event generates every time user object is changed.

This event generates on domain controllers, member servers, and workstations.

For each change, a separate 4738 event will be generated.

You might see this event without any changes inside, that is, where all **Changed Attributes** appear as “-”. This usually happens when a change is made to an attribute that is not listed in the event. In this case there is no way to determine which attribute was changed. For example, if the [discretionary access control list](#) (DACL) is changed, a 4738 event will generate, but all attributes will be “-”.

Some changes do not invoke a 4738 event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4738</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-20T16:22:02.792454100Z" />
<EventRecordID>175413</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1508" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="Dummy">-</Data>
<Data Name="TargetUserName">ksmith</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6609</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30dc2</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="SamAccountName">-</Data>
<Data Name="DisplayName">-</Data>
<Data Name="UserPrincipalName">-</Data>
<Data Name="HomeDirectory">-</Data>
<Data Name="HomePath">-</Data>
<Data Name="ScriptPath">-</Data>
<Data Name="ProfilePath">-</Data>
<Data Name="UserWorkstations">-</Data>
<Data Name="PasswordLastSet">-</Data>
<Data Name="AccountExpires">-</Data>
<Data Name="PrimaryGroupId">-</Data>
<Data Name="AllowedToDelegateTo">-</Data>
<Data Name="OldUacValue">0x15</Data>
<Data Name="NewUacValue">0x211</Data>
<Data Name="UserAccountControl">%%2050 %%2089</Data>
<Data Name="UserParameters">-</Data>
<Data Name="SidHistory">-</Data>
<Data Name="LogonHours">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change user account” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change user account” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

- **Security ID** [Type = SID]: SID of account that was changed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that was changed.
- **Account Domain** [Type = UnicodeString]: target account’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

Changed Attributes:

If attribute was not changed it will have “–” value.

Unfortunately, for local accounts, all fields, except changed attributes, will have previous values populated. Also, the **User Account Control** field will have values only if it was modified. Changed attributes will have new values, but it is hard to understand which attribute was really changed.

- **SAM Account Name** [Type = UnicodeString]: logon name for account used to support clients and servers from previous versions of Windows (pre-Windows 2000 logon name). If the value of **sAMAccountName** attribute of user object was changed, you will see the new value here. For example: ladmin. For local accounts, this field always has some value—if the account’s attribute was not changed it will contain the current value of the attribute.
- **Display Name** [Type = UnicodeString]: it is a name, displayed in the address book for a particular account. This is usually the combination of the user’s first name, middle initial, and last name. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. If the value of **displayName** attribute of user object was changed, you will see the new value here. For local accounts, this field always has some value—if the account’s attribute was not changed it will contain the current value of the attribute.

- **User Principal Name** [Type = UnicodeString]: internet-style login name for the account, based on the Internet standard RFC 822. By convention this should map to the account's email name. If the value of **userPrincipalName** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field is not applicable and always has “-” value.
- **Home Directory** [Type = UnicodeString]: user's home directory. If **homeDrive** attribute is set and specifies a drive letter, **homeDirectory** should be a UNC path. The path must be a network UNC of the form \\Server\Share\Directory. If the value of **homeDirectory** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **Home Drive** [Type = UnicodeString]: specifies the drive letter to which to map the UNC path specified by **homeDirectory** account's attribute. The drive letter must be specified in the form “**DRIVE LETTER:**”. For example – “H:”. If the value of **homeDrive** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **Script Path** [Type = UnicodeString]: specifies the path of the account's logon script. If the value of **scriptPath** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **Profile Path** [Type = UnicodeString]: specifies a path to the account's profile. This value can be a null string, a local absolute path, or a UNC path. If the value of **profilePath** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **User Workstations** [Type = UnicodeString]: contains the list of NetBIOS or DNS names of the computers from which the user can logon. Each computer name is separated by a comma. The name of a computer is the **sAMAccountName** property of a computer object. If the value of **userWorkstations** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field is not applicable and always appears as “**<value not set>**.”
- **Password Last Set** [Type = UnicodeString]: last time the account's password was modified. If the value of **pwdLastSet** attribute of user object was changed, you will see the new value here. For example: 8/12/2015 11:41:39 AM. This value will be changed, for example, after manual user account password reset. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **Account Expires** [Type = UnicodeString]: the date when the account expires. If the value of **accountExpires** attribute of user object was changed, you will see the new value here. For example, “9/21/2015 12:00:00 AM”. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. For local accounts, this field always has some value—if the account's attribute was not changed it will contain the current value of the attribute.
- **Primary Group ID** [Type = UnicodeString]: Relative Identifier (RID) of user's object primary group.

Relative identifier (RID) is a variable length number that is assigned to objects at creation and becomes part of the object's Security Identifier (SID) that uniquely identifies an account or group within a domain.

This field will contain some value if user's object primary group was changed. You can change user's primary group using Active Directory Users and Computers management console in the **Member Of** tab of user object properties. You will see a RID of new primary group as a field value. For example, RID 513 (Domain Users) is a default primary group for users.

Typical **Primary Group** values for user accounts:

- 513 (Domain Users. For local accounts this RID means Users) – for domain and local users.

See this article <https://support.microsoft.com/en-us/kb/243330> for more information. If the value of **primaryGroupID** attribute of user object was changed, you will see the new value here.

- **AllowedToDelegateTo** [Type = UnicodeString]: the list of SPNs to which this account can present delegated credentials. Can be changed using Active Directory Users and Computers management console in **Delegation** tab of user account, if at least one SPN is registered for user account. If the SPNs list on **Delegation** tab of a user account was changed, you will see the new SPNs list in **AllowedToDelegateTo** field (note that you will see the new list instead of changes) of this event. This is an example of **AllowedToDelegateTo**:

- dcom/WIN2012
- dcom/WIN2012.contoso.local

If the value of **msDS-AllowedToDelegateTo** attribute of user object was changed, you will see the new value here.

The value can be “<value not set>”, for example, if delegation was disabled.

For local accounts, this field is not applicable and always has “-” value.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Old UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user account. This parameter contains the previous value of **userAccountControl** attribute of user object.
- **New UAC Value** [Type = UnicodeString]: specifies flags that control password, lockout, disable/enable, script, and other behavior for the user account. If the value of **userAccountControl** attribute of user object was changed, you will see the new value here.

To decode this value, you can go through the property value definitions in the “Table 7. User’s or Computer’s account UAC flags.” from largest to smallest. Compare each property value to the flags value in the event. If the flags value in the event is greater than or equal to the property value, then the property is “set” and applies to that event. Subtract the property value from the flags value in the event and note that the flag applies and then go on to the next flag.

Here's an example: Flags value from event: 0x15

Decoding:

- PASSWD_NOTREQD 0x0020
- LOCKOUT 0x0010
- HOMEDIR_REQUIRED 0x0008
- (undeclared) 0x0004
- ACCOUNTDISABLE 0x0002
- SCRIPT 0x0001

0x0020 > 0x15, so PASSWD_NOTREQD does not apply to this event

0x10 < 0x15, so LOCKOUT applies to this event. 0x15 - 0x10 = 0x5

0x4 < 0x5, so the undeclared value is set. We'll pretend it doesn't mean anything. 0x5 - 0x4 = 0x1

0x2 > 0x1, so ACCOUNTDISABLE does not apply to this event

0x1 = 0x1, so SCRIPT applies to this event. 0x1 - 0x1 = 0x0, we're done.

So this UAC flags value decodes to: LOCKOUT and SCRIPT

- **User Account Control** [Type = UnicodeString]: shows the list of changes in **userAccountControl** attribute. You will see a line of text for each change. See possible values in here: “Table 7. User’s or Computer’s account UAC flags.”. In the “User Account Control field text” column, you can see the text that will be displayed in the **User Account Control** field in 4738 event.
- **User Parameters** [Type = UnicodeString]: if you change any setting using Active Directory Users and Computers management console in Dial-in tab of user’s account properties, then you will see **<value changed, but not displayed>** in this field. For local accounts, this field is not applicable and always has “**<value not set>**” value.
- **SID History** [Type = UnicodeString]: contains previous SIDs used for the object if the object was moved from another domain. Whenever an object is moved from one domain to another, a new SID is created and becomes the objectSID. The previous SID is added to the **sIDHistory** property. If the value of **sIDHistory** attribute of user object was changed, you will see the new value here.
- **Logon Hours** [Type = UnicodeString]: hours that the account is allowed to logon to the domain. If the value of **logonHours** attribute of user object was changed, you will see the new value here. You can change this attribute by using Active Directory Users and Computers, or through a script, for example. Here is an example of this field:

Sunday 12:00 AM - 7:00 PM

Sunday 9:00 PM -Monday 1:00 PM

Monday 2:00 PM -Tuesday 6:00 PM

Tuesday 8:00 PM -Wednesday 10:00 AM

For local accounts this field is not applicable and typically has value “**All**”.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”.

Security Monitoring Recommendations:

For 4738(S): A user account was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Some organizations monitor every [4738](#) event.
- If you have critical user computer accounts (for example, domain administrator accounts or service accounts) for which you need to monitor each change, monitor this event with the **“Target Account\Account Name”** that corresponds to the critical account or accounts.
- If you have user accounts for which any change in the services list on the **Delegation** tab should be monitored, monitor this event when **AllowedToDelegateTo** is not -. This value means the services list was changed.
- Consider whether to track the following fields:

Field to track	Reason to track
Display Name User Principal Name Home Directory Home Drive Script Path Profile Path User Workstations Password Last Set	We recommend monitoring all changes for these fields for critical domain and local accounts.

Account Expires	
Primary Group ID	
Logon Hours	
Primary Group ID is not 513	Typically, the Primary Group value is 513 for domain and local users. Other values should be monitored.
For user accounts for which the services list (on the Delegation tab) should not be empty: AllowedToDelegateTo is marked <value not set>	If AllowedToDelegateTo is marked <value not set> on user accounts that previously had a services list (on the Delegation tab), it means the list was cleared.
SID History is not -	This field will always be set to - unless the account was migrated from another domain.

- Consider whether to track the following user account control flags:

User account control flag to track	Information about the flag
'Normal Account' – Disabled	Should not be disabled for user accounts.
'Password Not Required' – Enabled	Should not typically be enabled for user accounts because it weakens security for the account.
'Encrypted Text Password Allowed' – Enabled	Should not typically be enabled for user accounts because it weakens security for the account.
'Server Trust Account' – Enabled	Should never be enabled for user accounts. Applies only to domain controller (computer) accounts.
'Don't Expire Password' – Enabled	Should be monitored for critical accounts, or all accounts if your organization does not allow this flag.
'Smartcard Required' – Enabled	Should be monitored for critical accounts.
'Password Not Required' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."
'Encrypted Text Password Allowed' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."
'Don't Expire Password' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."
'Smartcard Required' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."
'Trusted For Delegation' – Enabled	Means that Kerberos Constraint or Unconstraint delegation was enabled for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Trusted For Delegation' – Disabled	Means that Kerberos Constraint or Unconstraint delegation was disabled for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action. Also, if you have a list of user accounts for which delegation is critical and should not be disabled, monitor this for those accounts.
'Trusted To Authenticate For Delegation' – Enabled	Means that Protocol Transition delegation was enabled for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Trusted To Authenticate For Delegation' – Disabled	Means that Protocol Transition delegation was disabled for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.

	Also, if you have a list of user accounts for which delegation is critical and should not be disabled, monitor this for those accounts.
'Not Delegated' – Enabled	Means that Account is sensitive and cannot be delegated was checked for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Not Delegated' – Disabled	Should be monitored for all accounts where the setting should be "Enabled." Means that Account is sensitive and cannot be delegated was unchecked for the user account. We recommend monitoring this to discover whether it is an approved action (done by an administrator), a mistake, or a malicious action.
'Use DES Key Only' – Enabled	Should not typically be enabled for user accounts because it weakens security for the account's Kerberos authentication.
'Don't Require Preauth' – Enabled	Should not be enabled for user accounts because it weakens security for the account's Kerberos authentication.
'Use DES Key Only' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."
'Don't Require Preauth' – Disabled	Should be monitored for all accounts where the setting should be "Enabled."

4740(S): A user account was locked out.

Event Properties - Event 4740, Microsoft Windows security audit... X

General Details

Security ID: SYSTEM Account Name: DC01\$ Account Domain: CONTOSO Logon ID: 0x3E7	Account That Was Locked Out: Security ID: CONTOSO\Auditor Account Name: Auditor
Additional Information: Caller Computer Name: WIN81	
Log Name: Security Source: Microsoft Windows sev Event ID: 4740 Level: Information User: N/A OpCode: Info More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time a user account is locked out.

For user accounts, this event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4740</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-21T22:06:08.576887500Z" />
<EventRecordID>175703</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1112" />

```

<Channel>Security</Channel>

<Computer>DC01.contoso.local</Computer>

```
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">Auditor</Data>
<Data Name="TargetDomainName">WIN81</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that performed the lockout operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that performed the lockout operation.
- **Account Domain** [Type = UnicodeString]: domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Account That Was Locked Out:

- **Security ID** [Type = SID]: SID of account that was locked out. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that was locked out.

Additional Information:

- **Caller Computer Name** [Type = UnicodeString]: the name of computer account from which logon attempt was received and after which target account was locked out. For example: WIN81.

Security Monitoring Recommendations:

For 4740(S): A user account was locked out.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- If you have high-value domain or local accounts (for example, domain administrator accounts) for which you need to monitor every lockout, monitor all [4740](#) events with the “**Account That Was Locked Out \Security ID**” values that correspond to the accounts.
- If you have a high-value domain or local account for which you need to monitor every change, monitor all [4740](#) events with the “**Account That Was Locked Out \Security ID**” that corresponds to the account.
- If the user account “**Account That Was Locked Out\Security ID**” should not be used (for authentication attempts) from the **Additional Information\Caller Computer Name**, then trigger an alert.
- Monitor for all [4740](#) events where **Additional Information\Caller Computer Name** is not from your domain. However, be aware that even if the computer is not in your domain you will get the computer name instead of an IP address in the [4740](#) event.

4765(**S**): SID History was added to an account.

This event generates when [SID History](#) was added to an account.

See more information about SID History here: [https://technet.microsoft.com/en-us/library/cc779590\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc779590(v=ws.10).aspx).

There is no example of this event in this document.

Event Schema:

SID History was added to an account.

Subject:

*Security ID:%6
Account Name:%7
Account Domain:%8
Logon ID:%9*

Target Account:

*Security ID:%5
Account Name:%3
Account Domain:%4*

Source Account:

*Security ID:%2
Account Name:%1*

Additional Information:

Privileges:%10

SID List:%11

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4766(F): An attempt to add SID History to an account failed.

This event generates when an attempt to add [SID History](#) to an account failed.

See more information about SID History here: [https://technet.microsoft.com/en-us/library/cc779590\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc779590(v=ws.10).aspx).

There is no example of this event in this document.

Event Schema:

An attempt to add SID History to an account failed.

Subject:

Security ID:-

Account Name:%5

Account Domain:%6

Logon ID:%7

Target Account:

Security ID:%4

Account Name:%2

Account Domain:%3

Source Account:

Account Name:%1

Additional Information:

Privileges:%8

Required Server Roles: Active Directory domain controller.

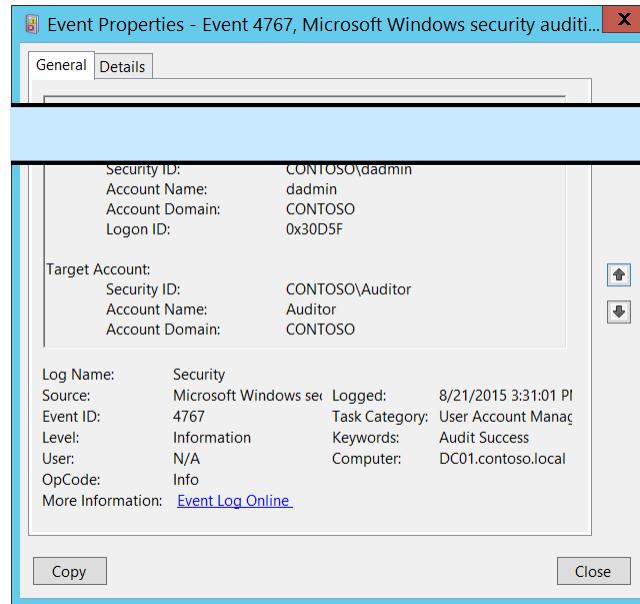
Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4767(S): A user account was unlocked.

 Event Properties - Event 4767, Microsoft Windows security audit... X

<input checked="" type="checkbox"/> General	<input type="checkbox"/> Details
---	----------------------------------

Event Description:
This event generates every time a user account is unlocked.
For user accounts, this event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4767</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-21T22:31:01.871931700Z" />
<EventRecordID>175705</EventRecordID>
<Correlation />

<Execution ProcessID="520" ThreadID="1520" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">Auditor</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d5f</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that performed the unlock operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that performed the unlock operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Target Account:

- **Security ID** [Type = SID]: SID of account that was unlocked. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that was unlocked.
- **Account Domain** [Type = UnicodeString]: target account's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

Security Monitoring Recommendations:

For 4767(S): A user account was unlocked.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. We recommend monitoring all [4767](#) events for local accounts.

4780(S): The ACL was set on accounts which are members of administrators groups.

Every hour, the domain controller that holds the primary domain controller (PDC) Flexible Single Master Operation (FSMO) role compares the ACL on all security principal accounts (users, groups, and machine accounts) present for its domain in Active Directory and that are in administrative or security-sensitive groups and which have AdminCount attribute = 1 against the ACL on the [AdminSDHolder](#) object. If the ACL on the principal account differs from the ACL on the AdminSDHolder object, then the ACL on the principal account is reset to match the ACL on the AdminSDHolder object and this event is generated.

For some reason, this event doesn't generate on some OS versions.

Event Schema:

The ACL was set on accounts which are members of administrators groups.

Subject:

Security ID:%4
Account Name:%5
Account Domain:%6
Logon ID:%7

Target Account:

Security ID:%3

Event Properties - Event 4781, Microsoft Windows security auditi... X

General Details

The name of an account was changed:

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x30D5F

Target Account:

Security ID:	CONTOSO\Admin
Account Domain:	CONTOSO
Old Account Name:	Admin
New Account Name:	MainAdmin

Additional Information:

Privileges: %8

Log Name: Security
Source: Microsoft Windows sec...
Event ID: 4781
Level: Information
User: N/A
OpCode: Info

Logged: 8/21/2015 7:41:09 PM
Task Category: User Account Mana...
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Account Name:%1
Account Domain:%2

Additional Information:
Privileges: %8

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

[Security Monitoring Recommendations:](#)

- Monitor for this event and investigate why the object's ACL was changed.

4781(S): The name of an account was changed.

Event Description:

This event generates every time a user or computer account name (**sAMAccountName** attribute) is changed.

For user accounts, this event generates on domain controllers, member servers, and workstations.

For computer accounts, this event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4781</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T02:41:09.737420900Z" />
<EventRecordID>175754</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1112" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="OldTargetUserName">Admin</Data>
<Data Name="NewTargetUserName">MainAdmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-6117</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d5f</Data>
<Data Name="PrivilegeList">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that performed the “change account name” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that performed the “change account name” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

- **Security ID** [Type = SID]: SID of account on which the name was changed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Domain** [Type = UnicodeString]: target account’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Old Account Name** [Type = UnicodeString]: old name of target account.
- **New Account Name** [Type = UnicodeString]: new name of target account.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in “Table 8. User Privileges.”

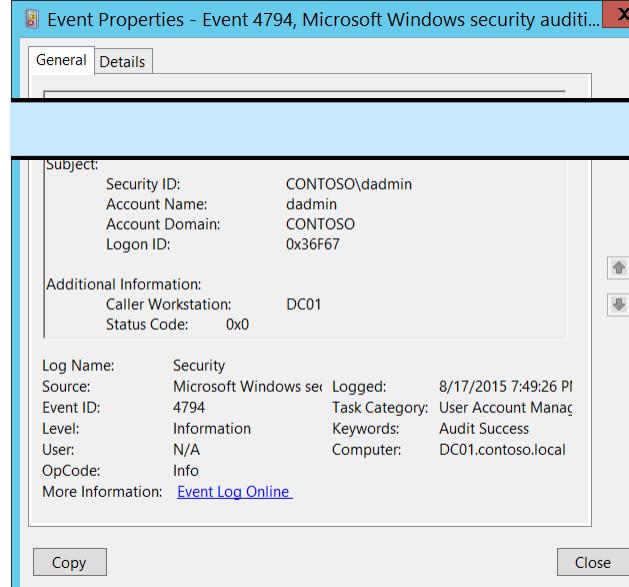
Security Monitoring Recommendations:

For 4781(S): The name of an account was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have high-value user or computer accounts (or local user accounts) for which you need to monitor each change to the accounts, monitor this event with the “**Target Account\Security ID**” that corresponds to the high-value accounts.

4794(S, F): An attempt was made to set the Directory Services Restore Mode administrator password.

 Event Properties - Event 4794, Microsoft Windows security auditi... X

General Details

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadadmin
Account Domain:	CONTOSO
Logon ID:	0x36F67

Additional Information:

Caller Workstation:	DC01
Status Code:	0x0

Log Name: Security
Source: Microsoft Windows sec... Logged: 8/17/2015 7:49:26 PM
Event ID: 4794 Task Category: User Account Mana...
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time Directory Services Restore Mode (DSRM) administrator password is changed.
This event generates only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4794</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-18T02:49:26.087748900Z" />
<EventRecordID>172348</EventRecordID>
<Correlation />
```

```
<Execution ProcessID="520" ThreadID="2964" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x36f67</Data>
<Data Name="Workstation">DC01</Data>
<Data Name="Status">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made an attempt to set Directory Services Restore Mode administrator password. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to set Directory Services Restore Mode administrator password.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Additional Information:

- **Caller Workstation** [Type = UnicodeString]: the name of computer account from which Directory Services Restore Mode (DSRM) administrator password change request was received. For example: "**DC01**". If the change request was sent locally (from the same server) this field will have the same name as the computer account.

- **Status Code** [Type = HexInt32]: for Success events it has "**0x0**" value.

Security Monitoring Recommendations:

For 4794(S, F): An attempt was made to set the Directory Services Restore Mode administrator password.

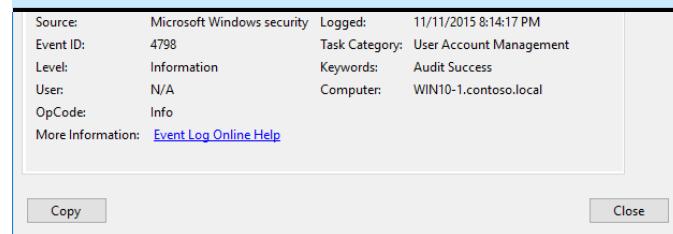
- Always monitor 4794 events and trigger alerts when they occur.

4798(S): A user's local group membership was enumerated.

Event Description:

This event generates when a process enumerates a user's security-enabled local groups on a computer or device.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



The screenshot shows the 'Event Properties' dialog box for Event 4798. The 'General' tab is selected. The event details are as follows:

Source:	Microsoft Windows security	Logged:	11/11/2015 8:14:17 PM
Event ID:	4798	Task Category:	User Account Management
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	WIN10-1.contoso.local
OpCode:	Info		
More Information: Event Log Online Help			

At the bottom left are 'Copy' and 'Close' buttons.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4798</EventID>
```

```
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T04:14:17.436787700Z" />
<EventRecordID>691</EventRecordID>
<Correlation ActivityID="{CBAEDE08-1CF0-0000-50DE-AECBF01CD101}" />
<Execution ProcessID="744" ThreadID="3928" />
<Channel>Security</Channel>
<Computer>WIN10-1.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserName">Administrator</Data>
<Data Name="TargetDomainName">WIN10-1</Data>
<Data Name="TargetSid">S-1-5-21-1694160624-234216347-2203645164-500</Data>
<Data Name="SubjectUserSid">S-1-5-21-1377283216-344919071-3415362939-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x72d9d</Data>
<Data Name="CallerProcessId">0xc80</Data>
<Data Name="CallerProcessName">C:\Windows\System32\mmc.exe</Data>
</EventData>
</Event>
```

Required Server Roles: none.

Minimum OS Version: Windows Server 2016, Windows 10.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “enumerate user's security-enabled local groups” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enumerate user's security-enabled local groups” operation.

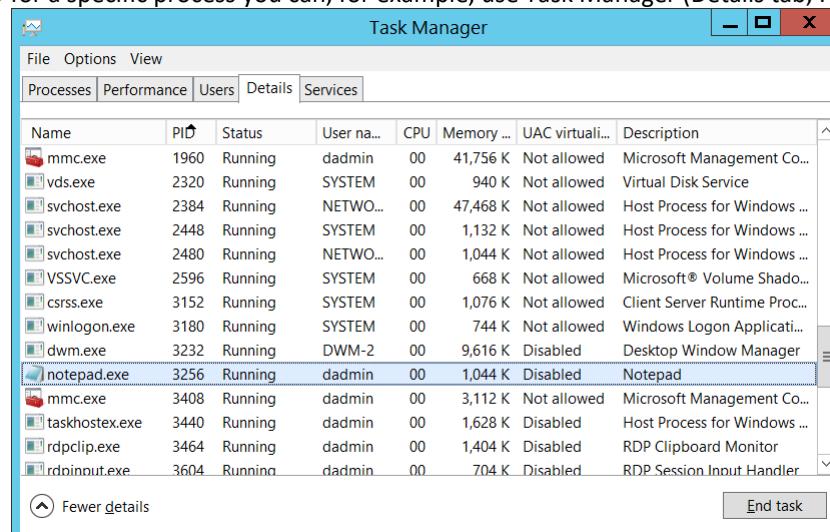
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

User:

- **Security ID** [Type = SID]: SID of the account whose groups were enumerated. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account whose groups were enumerated.
- **Account Domain** [Type = UnicodeString]: group's domain or computer name. Formats vary, and include the following:
 - For a local group, this field will contain the name of the computer to which this group belongs, for example: "Win81".
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that enumerated the members of the group. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

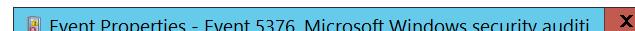
Security Monitoring Recommendations:

For 4798(S): A user's local group membership was enumerated.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have high value domain or local accounts for which you need to monitor each enumeration of their group membership, or any access attempt, monitor events with the “**Subject\Security ID**” that corresponds to the high value account or accounts.
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.

5376(S): Credential Manager credentials were backed up.

 Event Properties - Event 5376, Microsoft Windows security audit...

General Details

Credential Manager credentials were backed up.

Account Domain: CONTOSO
Logon ID: 0x30D7C

This event occurs when a user backs up their own Credential Manager credentials. A user (even an Administrator) cannot back up the credentials of an account other than his own.

Log Name: Security
Source: Microsoft Windows security
Event ID: 5376
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time the user (**Subject**) successfully backs up the [credential manager](#) database.

Typically this can be done by clicking “Back up Credentials” in Credential Manager in the Control Panel.

This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5376</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T03:28:02.200404700Z" />
```

```

<EventRecordID>175779</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="548" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d7c</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that performed the backup operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that performed the backup operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Security Monitoring Recommendations:

For 5376(S): Credential Manager credentials were backed up.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Every [5376](#) event should be recorded for all local and domain accounts, because this action (back up Credential Manager) is very rarely used by users and can indicate a virus, or other harmful or malicious activity.

5377(S): Credential Manager credentials were restored from a backup.

 Event Properties - Event 5377, Microsoft Windows security auditi... X

General Details

Credential Manager credentials were restored from a backup.

Account Domain:	CONTOSO
Logon ID:	0x30D7C

This event occurs when a user restores his Credential Manager credentials from a backup. A user (even an Administrator) cannot restore the credentials of an account other than his own.

Log Name: Security
Source: Microsoft Windows security
Event ID: 5377
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 8/21/2015 8:35:47 PM
Task Category: User Account Management
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Event Description:

This event generates every time the user (**Subject**) successfully restores the [credential manager](#) database. Typically this can be done by clicking “Restore Credentials” in Credential Manager in the Control Panel. This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5377</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13824</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-22T03:35:47.523266300Z" />
```

```
<EventRecordID>175780</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1236" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId" S-1-5-21-3457937927-2839227994-823803824-1104></Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d7c</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that performed the restore operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that performed the restore operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Security Monitoring Recommendations:

For 5377(S): Credential Manager credentials were restored from a backup.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Every [5377](#) event should be recorded for all local and domain accounts, because this action (restore Credential Manager credentials from a backup) is very rarely used by users, and can indicate a virus, or other harmful or malicious activity.

Detailed Tracking

Audit DPAPI Activity

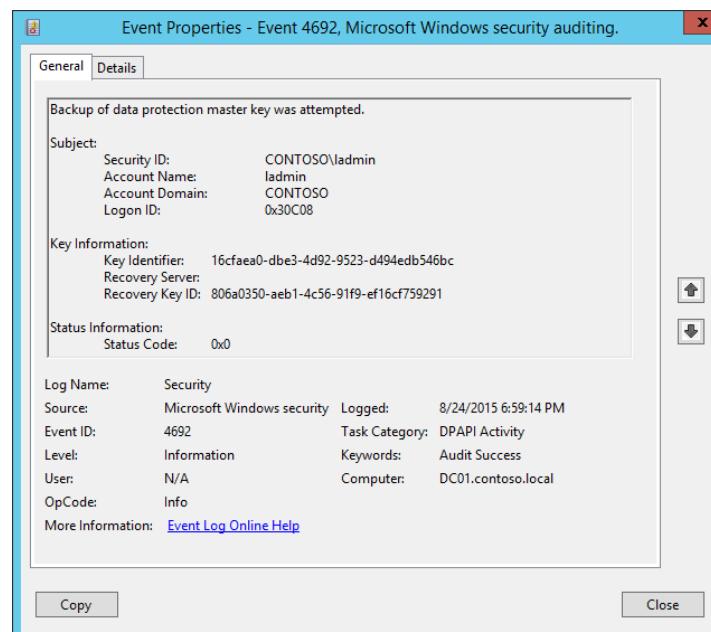
Audit [DPAPI](#) Activity determines whether the operating system generates audit events when encryption or decryption calls are made into the data protection application interface ([DPAPI](#)).

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF – Events in this subcategory typically have an informational purpose and it is difficult to detect any malicious activity using these events. It's mainly used for DPAPI troubleshooting.
Member Server	IF	IF	IF	IF	IF – Events in this subcategory typically have an informational purpose and it is difficult to detect any malicious activity using these events. It's mainly used for DPAPI troubleshooting.
Workstation	IF	IF	IF	IF	IF – Events in this subcategory typically have an informational purpose and it is difficult to detect any malicious activity using these events. It's mainly used for DPAPI troubleshooting.

Events List:

- [4692](#)(S, F): Backup of data protection master key was attempted.
- [4693](#)(S, F): Recovery of data protection master key was attempted.
- [4694](#)(S, F): Protection of auditable protected data was attempted.
- [4695](#)(S, F): Unprotection of auditable protected data was attempted.



4692(S, F): Backup of data protection master key was attempted.

Event Description:

This event generates every time that a backup is attempted for the [DPAPI](#) Master Key. When a computer is a member of a domain, DPAPI has a backup mechanism to allow unprotection of the data. When a Master Key is generated, DPAPI communicates with a domain controller. Domain controllers have a domain-wide public/private key pair, associated solely with DPAPI. The local DPAPI client gets the domain controller public key from a domain controller by using a mutually authenticated and privacy protected RPC call. The client encrypts the Master Key with the domain controller public key. It then stores this backup Master Key along with the Master Key protected by the user's password.

Periodically, a domain-joined machine will try to send an RPC request to a domain controller to back up the user's master key so that the user can recover secrets in case his or her password has to be reset. Although the user's keys are stored in the user profile, a domain controller must be contacted to encrypt the master key with a domain recovery key.

This event also generates every time a new DPAPI Master Key is generated, for example.

This event generates on domain controllers, member servers, and workstations.

Failure event generates when a Master Key backup operation fails for some reason.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4692</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13314</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-25T01:59:14.573672700Z" />
<EventRecordID>176964</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="540" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-500</Data>
<Data Name="SubjectUserName">ladmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30c08</Data>
<Data Name="MasterKeyId">16cfaea0-dbe3-4d92-9523-d494edb546bc</Data>
<Data Name="RecoveryServer" />
<Data Name="RecoveryKeyId">806a0350-aeb1-4c56-91f9-ef16cf759291</Data>
<Data Name="FailureReason">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested backup operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested backup operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

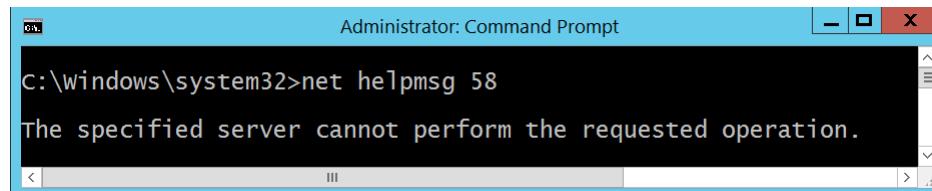
Key Information:

- **Key Identifier** [Type = UnicodeString]: unique identifier of a master key which backup was created. The Master Key is used, with some additional data, to generate an actual symmetric session key to encrypt\decrypt the data using DPAPI. All of user's Master Keys are located in user profile -> %APPDATA%\Roaming\Microsoft\Windows\Protect\%SID% folder. The name of every Master Key file is its ID.
- **Recovery Server** [Type = UnicodeString]: the name (typically – DNS name) of the computer that you contacted to back up your Master Key. For domain joined machines, it's typically a name of a domain controller. This parameter might not be captured in the event, and in that case will be empty.
- **Recovery Key ID** [Type = UnicodeString]: unique identifier of a recovery key. The recovery key is generated when a user chooses to create a Password Reset Disk (PRD) from the user's Control Panel or when first Master Key is generated. First, DPAPI generates a RSA public/private key pair, which is the recovery key. In this field you will see unique Recovery key ID which was used for Master key backup operation.

For Failure events this field is typically empty.

Status Information:

- **Status Code** [Type = HexInt32]: hexadecimal unique status code of performed operation. For Success events this field is typically "0x0". To see the meaning of status code you need to convert it to decimal value and use "**net helpmsg STATUS_CODE**" command to see the description for specific STATUS_CODE. Here is an example of "net helpmsg" command output for status code 0x3A:



```
Administrator: Command Prompt
C:\windows\system32>net helpmsg 58
The specified server cannot perform the requested operation.
```

Security Monitoring Recommendations:

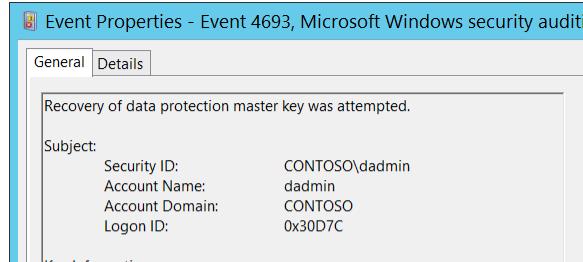
For 4692(S, F): Backup of data protection master key was attempted.

- This event is typically an informational event and it is difficult to detect any malicious activity using this event. It's mainly used for DPAPI troubleshooting.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

4693(S, F): Recovery of data protection master key was attempted.



Event Description:

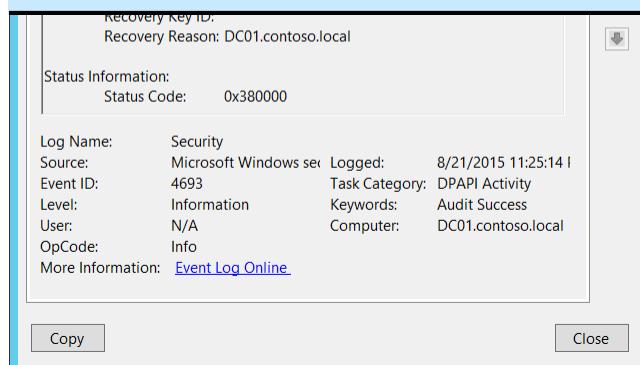
This event generates every time that recovery is attempted for a [DPAPI](#) Master Key.

While unprotecting data, if DPAPI cannot use the Master Key protected by the user's password, it sends the backup Master Key to a domain controller by using a mutually authenticated and privacy protected RPC call. The domain controller then decrypts the Master Key with its private key and sends it back to the client by using the same protected RPC call. This protected RPC call is used to ensure that no one listening on the network can get the Master Key.

This event generates on domain controllers, member servers, and workstations.

Failure event generates when a Master Key restore operation fails for some reason.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4693</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13314</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-22T06:25:14.589407700Z" />

```

```

<EventRecordID>175809</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="1340" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>

```

```
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x30d7c</Data>
<Data Name="MasterKeyId">0445c766-75f0-4de7-82ad-d9d97aad59f6</Data>
<Data Name="RecoveryReason">0x5c005c</Data>
<Data Name="RecoveryServer">DC01.contoso.local</Data>
<Data Name="RecoveryKeyId" />
<Data Name="FailureId">0x380000</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “recover” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “recover” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Key Information:

- **Key Identifier** [Type = UnicodeString]: unique identifier of a master key which was recovered. The Master Key is used, with some additional data, to generate an actual symmetric session key to encrypt\decrypt the data using DPAPI. All of user’s Master Keys are located in user profile -> %APPDATA%\Roaming\Microsoft\Windows\Protect\%SID% folder. The name of every Master Key file is it’s ID.
- **Recovery Server** [Type = UnicodeString]: the name (typically – DNS name) of the computer that you contacted to recover your Master Key. For domain joined machines, it’s typically a name of a domain controller.

In this event **Recovery Server** field contains information from **Recovery Reason** field.

- **Recovery Key ID** [Type = UnicodeString]: unique identifier of a recovery key. The recovery key is generated when a user chooses to create a Password Reset Disk (PRD) from the user's Control Panel or when first Master Key is generated. First, DPAPI generates a RSA public/private key pair, which is the recovery key. In this field you will see unique Recovery key ID which was used for Master key recovery operation. This parameter might not be captured in the event, and in that case will be empty.
- **Recovery Reason** [Type = HexInt32]: hexadecimal code of recovery reason.

In this event **Recovery Reason** field contains information from **Recovery Server** field.

Status Information:

- **Status Code** [Type = HexInt32]: hexadecimal unique status code. For Success events this field is typically “**0x380000**”.

Security Monitoring Recommendations:

For 4693(S, F): Recovery of data protection master key was attempted.

- This event is typically an informational event and it is difficult to detect any malicious activity using this event. It's mainly used for DPAPI troubleshooting.
- For domain joined computers, **Recovery Reason** should typically be a domain controller DNS name.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

4694(S, F): Protection of auditable protected data was attempted.

This event generates if [DPAPI CryptProtectData\(\)](#) function was used with **CRYPTPROTECT_AUDIT** flag (dwFlags) enabled.

There is no example of this event in this document.

Event Schema:

Protection of auditable protected data was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Protected Data:

*Data Description:%6
Key Identifier:%5
Protected Data Flags:%7
Protection Algorithms:%8*

Status Information:

Status Code:%9

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.
- This event is typically an informational event and it is difficult to detect any malicious activity using this event. It's mainly used for DPAPI troubleshooting.

4695(**S, F**): Unprotection of auditable protected data was attempted.

This event generates if [DPAPI CryptUnprotectData\(\)](#) function was used to unprotect “auditable” data that was encrypted using [CryptProtectData\(\)](#) function with **CRYPTPROTECT_AUDIT** flag (dwFlags) enabled.

There is no example of this event in this document.

Event Schema:

Unprotection of auditable protected data was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Protected Data:

*Data Description:%6
Key Identifier:%5
Protected Data Flags:%7
Protection Algorithms:%8*

Status Information:

Status Code:%9

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.
- This event is typically an informational event and it is difficult to detect any malicious activity using this event. It's mainly used for DPAPI troubleshooting.

Audit PNP Activity

Audit PNP Activity determines when Plug and Play detects an external device.

A PnP audit event can be used to track down changes in system hardware and will be logged on the machine where the change took place. For example, when a keyboard is plugged into a computer, a PnP event is triggered.

Event volume: Varies, depending on how the computer is used. Typically Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	This subcategory will help identify when and which Plug and Play device was attached, enabled, disabled or restricted by device installation policy. You can track, for example, whether a USB flash drive or stick was attached to a domain controller, which is typically not allowed. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	Yes	No	Yes	No	This subcategory will help identify when and which Plug and Play device was attached, enabled, disabled or restricted by device installation policy. You can track, for example, whether a USB flash drive or stick was attached to a critical server, which is typically not allowed. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	Yes	No	Yes	No	This subcategory will help identify when and which Plug and Play device was attached, enabled, disabled or restricted by device installation policy. You can track, for example, whether a USB flash drive or stick was attached to an administrative workstation or VIP workstation. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Events List:

- [6416\(S\)](#): A new external device was recognized by the System
- [6419\(S\)](#): A request was made to disable a device
- [6420\(S\)](#): A device was disabled.
- [6421\(S\)](#): A request was made to enable a device.
- [6422\(S\)](#): A device was enabled.
- [6423\(S\)](#): The installation of this device is forbidden by system policy.
- [6424\(S\)](#): The installation of this device was allowed, after having previously been forbidden by policy.

[6416\(S\)](#): A new external device was recognized by the System.

Event Properties - Event 6416, Microsoft Windows security auditing.

General Details

A new external device was recognized by the system.

Account Domain:	WORKGROUP
Logon ID:	0x3e7
Device ID:	SCSI\Disk&Ven_Seagate&Prod_Expansion\000000
Device Name:	Seagate Expansion SCSI Disk Device
Class ID:	{4d36e967-e325-11ce-bfc1-08002be10318}
Class Name:	DiskDrive
Vendor IDs:	SCSI\DiskSeagate_Expansion_____0636 SCSI\DiskSeagate_Expansion_____ SCSI\DiskSeagate_ SCSI\Seagate_Expansion_____0 Seagate_Expansion_____0 GenDisk
Compatible IDs:	SCSI\Disk SCSI\RAW
Location Information:	Bus Number 0, Target Id 0, LUN 0
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	6416
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online Help

Copy **Close**

Event Description:

This event generates every time a new external device is recognized by a system.
 This event generates, for example, when a new external device is connected or enabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>6416</EventID>
  <Version>1</Version>
  <Level>0</Level>
  <Task>13316</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-11-13T18:20:16.818569900Z" />
  <EventRecordID>436</EventRecordID>
  <Correlation />
  <Execution ProcessID="4" ThreadID="308" />
  <Channel>Security</Channel>
  <Computer>DESKTOP-NFC0HVN</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">DESKTOP-NFC0HVN$</Data>
  <Data Name="SubjectDomainName">WORKGROUP</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="DeviceId">SCSI\Disk&Ven_Seagate&Prod_Expansion\000000</Data>
<Data Name="DeviceDescription">Seagate Expansion SCSI Disk Device</Data>
<Data Name="ClassId">{4D36E967-E325-11CE-BFC1-08002BE10318}</Data>
<Data Name="ClassName">DiskDrive</Data>
<Data Name="VendorIds">SCSI\DiskSeagate_Expansion_____0636 SCSI\DiskSeagate_Expansion_____ SCSI\DiskSeagate_SCSI\Seagate_Expansion_____0  
Seagate_Expansion_____0 GenDisk</Data>
<Data Name="CompatibleIds">SCSI\Disk SCSI\RAW</Data>
<Data Name="LocationInformation">Bus Number 0, Target Id 0, LUN 0</Data>
</EventData>
```

</Event>

Required Server Roles: None.**Minimum OS Version:** Windows Server 2016, Windows 10.**Event Versions:**

- 0 - Windows 10.
- 1 - Windows 10 [Version 1511].
 - Added "Device ID" field.
 - Added "Device Name" field.
 - Added "Class Name" field.

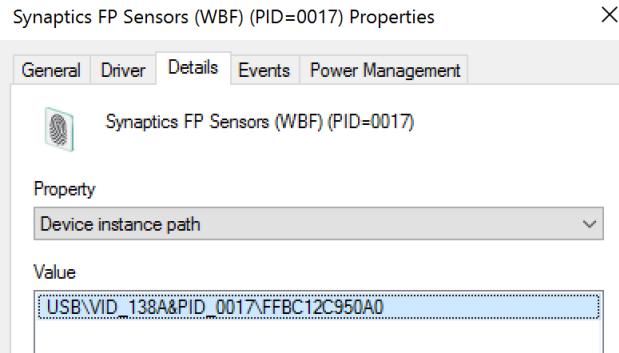
Field Descriptions:**Subject:**

- **Security ID** [Type = SID]: SID of account that registered the new device. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

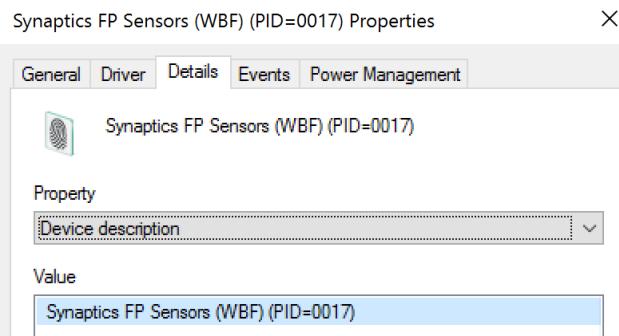
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that registered the new device.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

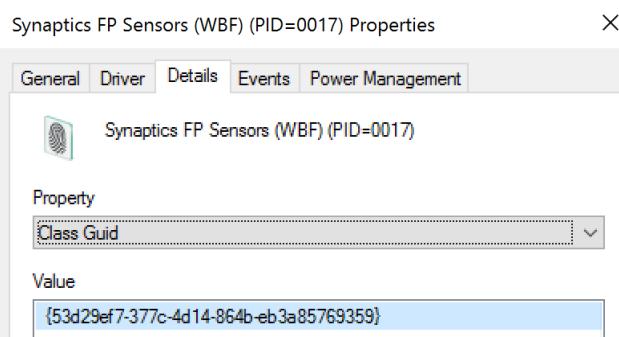
Device ID [Type = UnicodeString] [Version 1]: "**Device instance path**" attribute of device. To see device properties, start Device Manager, open specific device properties, and click "Details":



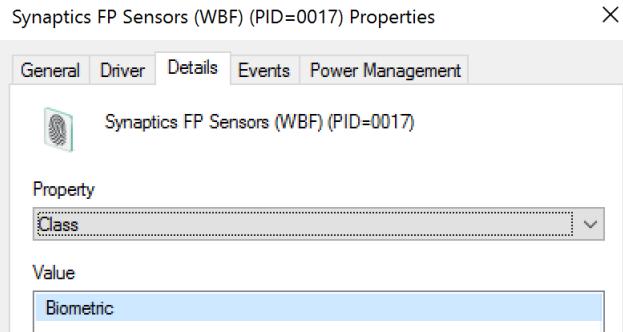
Device Name [Type = UnicodeString] [Version 1]: “**Device description**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



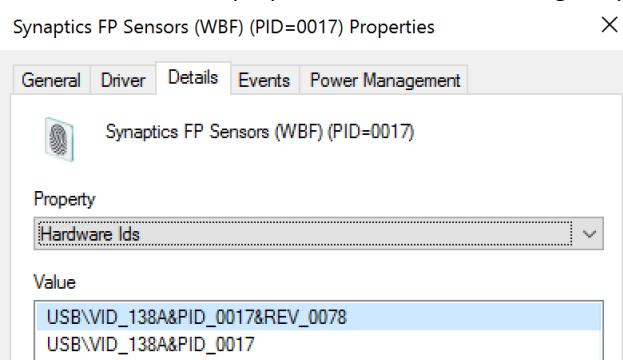
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



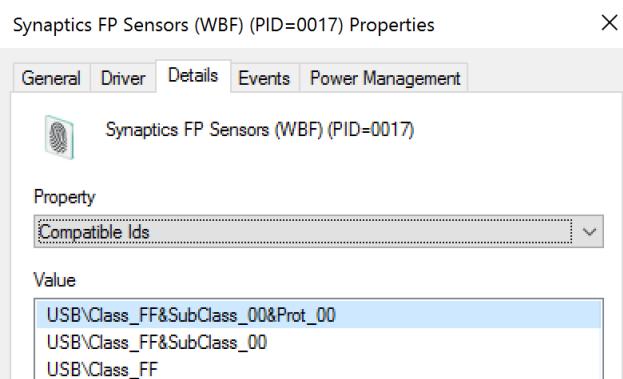
Class Name [Type = UnicodeString] [Version 1]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



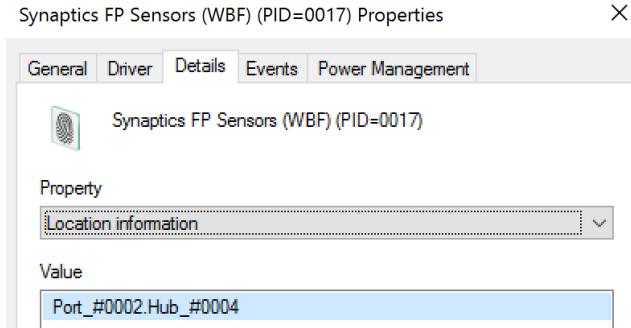
Vendor IDs [Type = UnicodeString]: “**Hardware Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6416(S): A new external device was recognized by the System.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- You can use this event to track the events and event information shown in the following table by using the listed fields:

Event and event information to monitor	Field to use
Device recognition events, Device Instance Path	“ Device ID ”
Device recognition events, Device Description	“ Device Name ”
Device recognition events, Class GUID	“ Class ID ”
Device recognition events, Hardware IDs	“ Vendor IDs ”
Device recognition events, Compatible IDs	“ Compatible IDs ”
Device recognition events, Location information	“ Location Information ”

6419(S): A request was made to disable a device.

Event Properties - Event 6419, Microsoft Windows security auditing.

[General](#) [Details](#)

Security ID:	DESKTOP-NFC0HV\ladmin
Account Name:	ladmin
Account Domain:	DESKTOP-NFC0HV
Logon ID:	0x3FCC7
Device ID:	USB\VID_138A&PID_0017\FFBC12C950A0
Device Name:	Synaptics FP Sensors (WBF) (PID=0017)
Class ID:	{53d29ef7-377c-4d14-864b-eb3a85769359}
Class Name:	Biometric
Hardware IDs:	USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017
Compatible IDs:	USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF
Location Information:	Port_#0002.Hub_#0004
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	6419
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online Help

[Copy](#) [Close](#)

Event Description:

This event generates every time when someone made a request to disable a device.
This event doesn't mean that device was disabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6419</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13316</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-14T22:23:26.789591400Z" />
<EventRecordID>483</EventRecordID>
<Correlation />
<Execution ProcessID="2192" ThreadID="1392" />
<Channel>Security</Channel>
<Computer>DESKTOP-NFC0HV</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-2695983153-1310895815-1903476278-1001</Data>
<Data Name="SubjectUserName">ladmin</Data>
<Data Name="SubjectDomainName">DESKTOP-NFC0HV</Data>
<Data Name="SubjectLogonId">0x3fcc7</Data>
<Data Name="DeviceId">USB\VID_138A&PID_0017\FFBC12C950A0</Data>
<Data Name="DeviceDescription">Synaptics FP Sensors (WBF) (PID=0017)</Data>

```

```

<Data Name="ClassId">{53D29EF7-377C-4D14-864B-EB3A85769359}</Data>
<Data Name="ClassName">Biometric</Data>
<Data Name="HardwareIds">USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017</Data>
<Data Name="CompatibleIds">USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF</Data>
<Data Name="LocationInformation">Port_#0002.Hub_#0004</Data>
</EventData>

```

</Event>

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Field Descriptions:

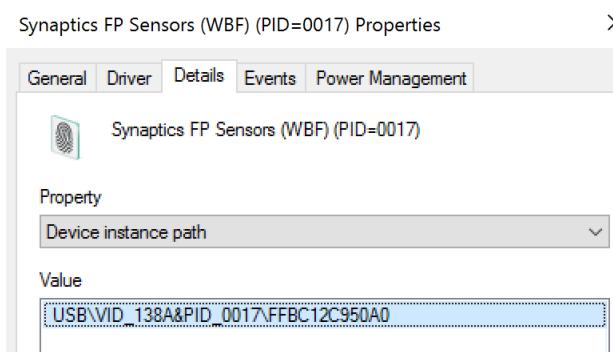
Subject:

- **Security ID** [Type = SID]: SID of account that made the request. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

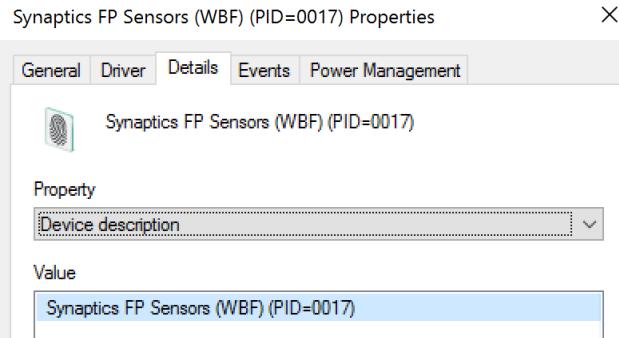
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made the request.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

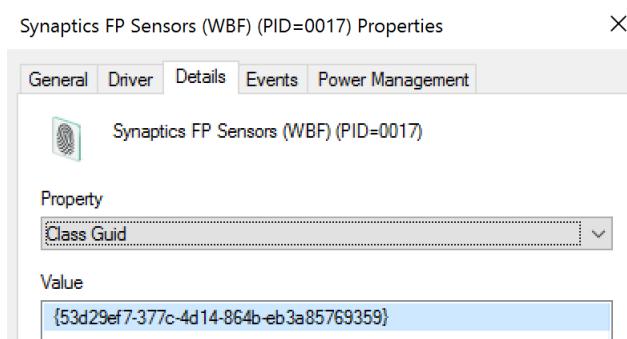
Device ID [Type = UnicodeString]: "Device instance path" attribute of device. To see device properties, start Device Manager, open specific device properties, and click "Details":



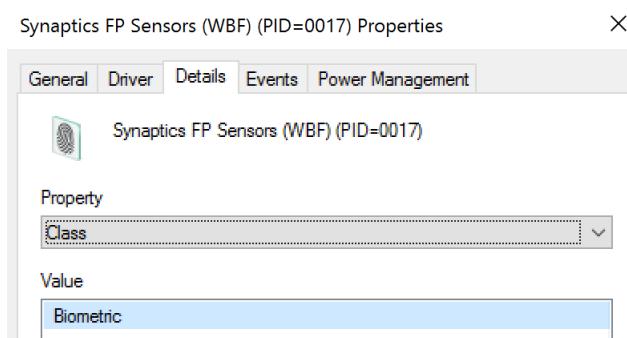
Device Name [Type = UnicodeString]: "Device description" attribute of device. To see device properties, start Device Manager, open specific device properties, and click "Details":



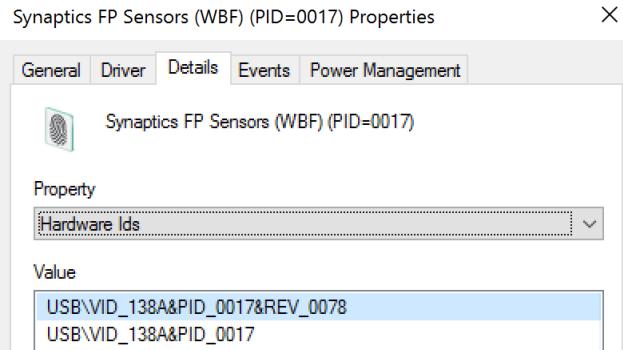
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



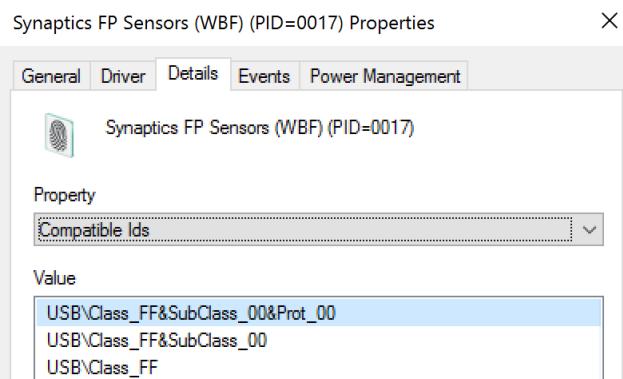
Class Name [Type = UnicodeString]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



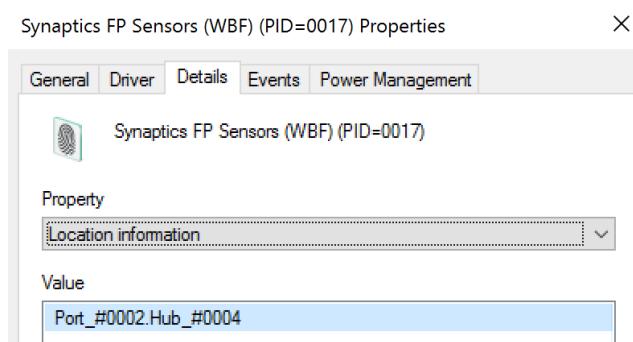
Hardware IDs [Type = UnicodeString]: “**Hardware Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6419(S): A request was made to disable a device.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. You can use this event to track the events and event information shown in the following table by using the listed fields:

Event and event information to monitor	Field to use
--	--------------

Device disable requests, Device Instance Path	“Device ID”
Device disable requests, Device Description	“Device Name”
Device disable requests, Class GUID	“Class ID”
Device disable requests, Hardware IDs	“Hardware IDs”
Device disable requests, Compatible IDs	“Compatible IDs”
Device disable requests, Location information	“Location Information”

6420(S): A device was disabled.

Event Description:

This event generates every time specific device was disabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event Properties - Event 6419, Microsoft Windows security auditing.

General Details

Security ID:	DESKTOP-NFC0HVN\ladmin
Account Name:	ladmin
Account Domain:	DESKTOP-NFC0HVN
Logon ID:	0x3FCC7
Device ID:	USB\VID_138A&PID_0017\FFBC12C950A0
Device Name:	Synaptics FP Sensors (WBF) (PID=0017)
Class ID:	{53d29ef7-377c-4d14-864b-eb3a85769359}
Class Name:	Biometric
Hardware IDs:	USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017
Compatible IDs:	USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF
Location Information:	Port_#0002.Hub_#0004
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	6419
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online Help

Copy **Close**

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6420</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13316</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-14T22:23:29.137398300Z" />
<EventRecordID>484</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="88" />
<Channel>Security</Channel>
<Computer>DESKTOP-NFC0HVN</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DESKTOP-NFC0HVN$</Data>
<Data Name="SubjectDomainName">WORKGROUP</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="DeviceId">USB\VID_138A&PID_0017\ffbc12c950a0</Data>
```

```
<Data Name="DeviceDescription">Synaptics FP Sensors (WBF) (PID=0017)</Data>
<Data Name="ClassId">{53D29EF7-377C-4D14-864B-EB3A85769359}</Data>
<Data Name="ClassName">Biometric</Data>
<Data Name="HardwareIds">USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017</Data>
<Data Name="CompatibleIds">USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF</Data>
<Data Name="LocationInformation">Port_#0002.Hub_#0004</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Field Descriptions:

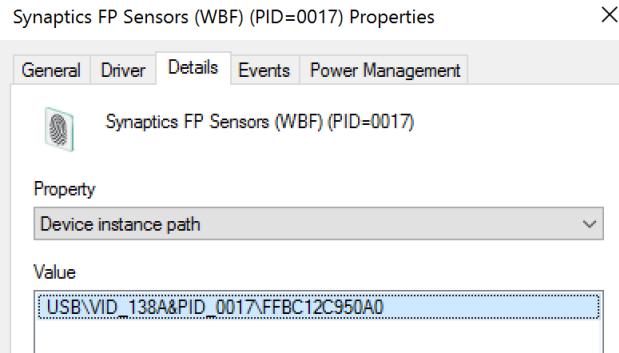
Subject:

- **Security ID** [Type = SID]: SID of account that disabled the device. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

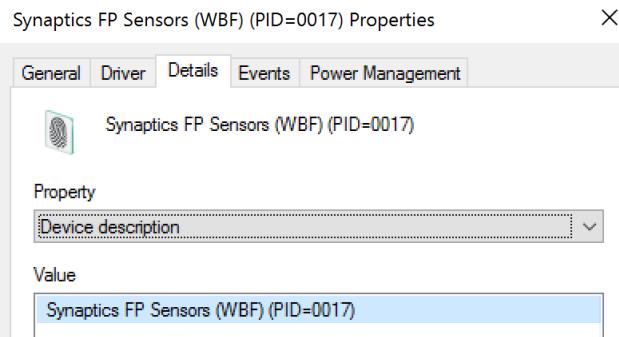
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that disabled the device.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

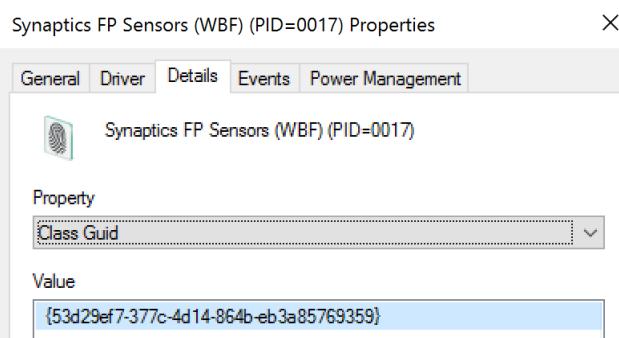
Device ID [Type = UnicodeString]: **“Device instance path”** attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



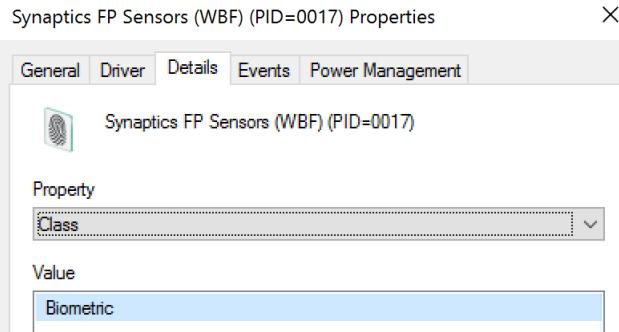
Device Name [Type = UnicodeString]: “**Device description**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



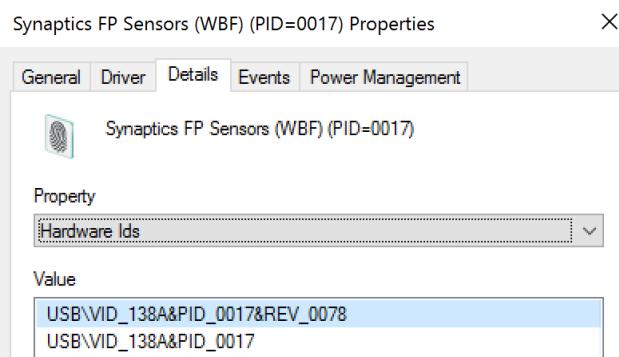
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



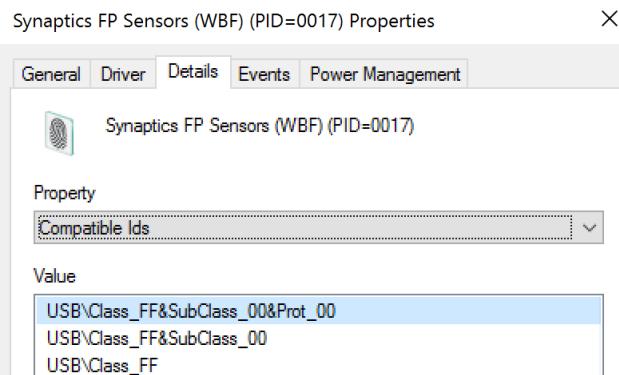
Class Name [Type = UnicodeString]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



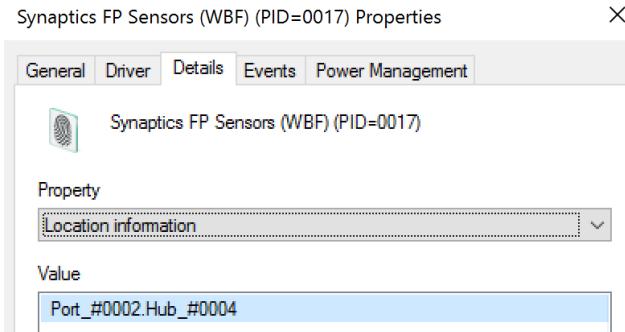
Hardware IDs [Type = UnicodeString]: “**Hardware Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6420(S): A device was disabled.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. You can use this event to track the events and event information shown in the following table by using the listed fields:

Event and event information to monitor	Field to use
Device disable events, Device Instance Path	“Device ID”
Device disable events, Device Description	“Device Name”
Device disable events, Class GUID	“Class ID”
Device disable events, Hardware IDs	“Hardware IDs”
Device disable events, Compatible IDs	“Compatible IDs”
Device disable events, Location information	“Location Information”

6421(S): A request was made to enable a device.

Event Properties - Event 6421, Microsoft Windows security auditing.

[General](#) [Details](#)

Security ID:	DESKTOP-NFC0HV\ladmin
Account Name:	ladmin
Account Domain:	DESKTOP-NFC0HV
Logon ID:	0x3FCC7
Device ID:	USB\VID_138A&PID_0017\FFBC12C950A0
Device Name:	Synaptics FP Sensors (WBF) (PID=0017)
Class ID:	{53d29ef7-377c-4d14-864b-eb3a85769359}
Class Name:	Biometric
Hardware IDs:	USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017
Compatible IDs:	USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF
Location Information:	Port_#0002.Hub_#0004
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	6421
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online Help

[Copy](#) [Close](#)

Event Description:

This event generates every time when someone made a request to enable a device.
This event doesn't mean that device was enabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6421</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13316</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-14T22:37:50.034918700Z" />
<EventRecordID>485</EventRecordID>
<Correlation />
<Execution ProcessID="2192" ThreadID="1392" />
<Channel>Security</Channel>
<Computer>DESKTOP-NFC0HV</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-2695983153-1310895815-1903476278-1001</Data>
<Data Name="SubjectUserName">ladmin</Data>
<Data Name="SubjectDomainName">DESKTOP-NFC0HV</Data>
<Data Name="SubjectLogonId">0x3fcc7</Data>
<Data Name="DeviceId">USB\VID_138A&PID_0017\FFBC12C950A0</Data>
<Data Name="DeviceDescription">Synaptics FP Sensors (WBF) (PID=0017)</Data>
<Data Name="ClassId">{53D29EF7-377C-4D14-864B-EB3A85769359}</Data>
<Data Name="ClassName">Biometric</Data>
<Data Name="HardwareIds">USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017</Data>
<Data Name="CompatibleIds">USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF</Data>
<Data Name="LocationInformation">Port_#0002.Hub_#0004</Data>
</EventData>
```

</Event>

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Field Descriptions:

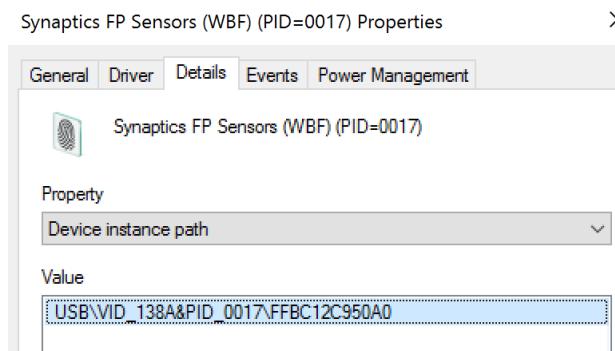
Subject:

- **Security ID** [Type = SID]: SID of account that made the request. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

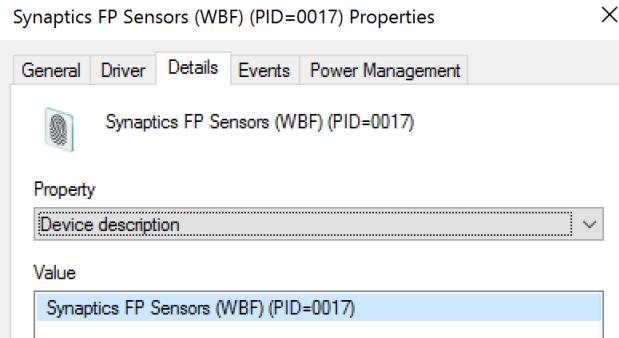
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made the request.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

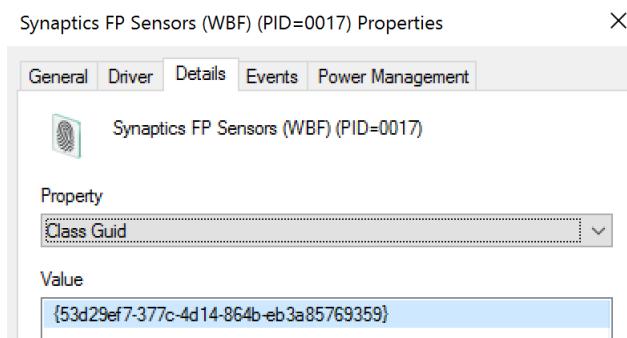
Device ID [Type = UnicodeString]: "Device instance path" attribute of device. To see device properties, start Device Manager, open specific device properties, and click "Details":



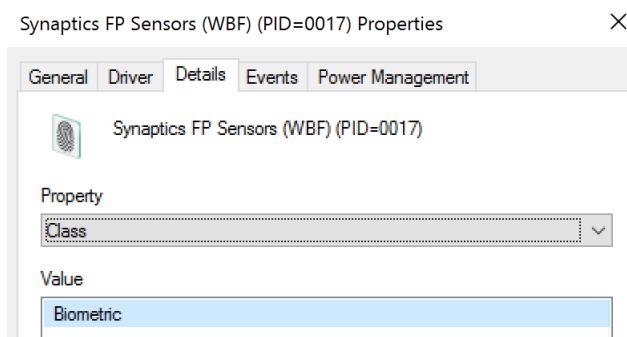
Device Name [Type = UnicodeString]: "Device description" attribute of device. To see device properties, start Device Manager, open specific device properties, and click "Details":



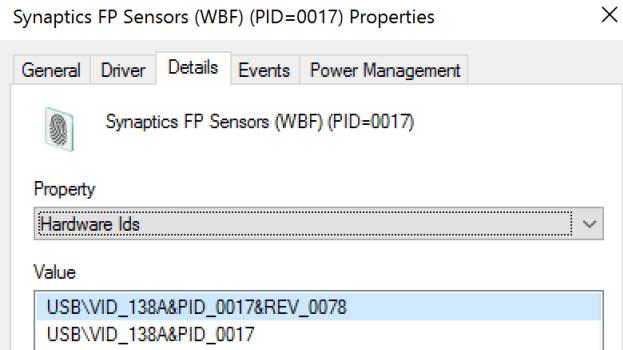
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



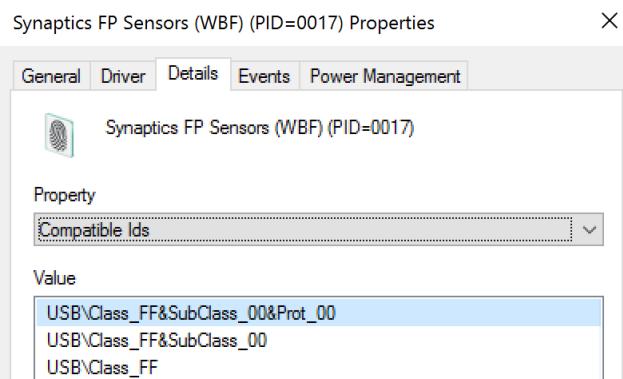
Class Name [Type = UnicodeString]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



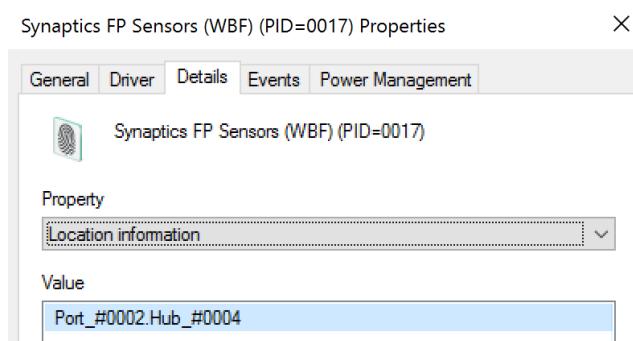
Hardware IDs [Type = UnicodeString]: “**Hardware Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6421(S): A request was made to enable a device.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. You can use this event to track the events and event information shown in the following table by using the listed fields:

Event and event information to monitor	Field to use
--	--------------

Device enable requests, Device Instance Path	“Device ID”
Device enable requests, Device Description	“Device Name”
Device enable requests, Class GUID	“Class ID”
Device enable requests, Hardware IDs	“Hardware IDs”
Device enable requests, Compatible IDs	“Compatible IDs”
Device enable requests, Location information	“Location Information”

Event Properties - Event 6422, Microsoft Windows security auditing.

General Details

Security ID: SYSTEM
 Account Name: DESKTOP-NFC0HVNS
 Account Domain: WORKGROUP
 Logon ID: 0x3E7

Device ID: USB\VID_138A&PID_0017\ffbc12c950a0

Device Name: Synaptics FP Sensors (WBF) (PID=0017)

Class ID: {53d29ef7-377c-4d14-864b-eb3a85769359}

Class Name: Biometric

Hardware IDs:
 USB\VID_138A&PID_0017&REV_0078
 USB\VID_138A&PID_0017

Compatible IDs:
 USB\Class_FF&SubClass_00&Prot_00
 USB\Class_FF&SubClass_00
 USB\Class_FF

Location Information:
 Port_#0002.Hub_#0004

Log Name: Security
 Source: Microsoft Windows security Logged: 11/14/2015 2:37:50 PM
 Event ID: 6422 Task Category: Plug and Play Events
 Level: Information Keywords: Audit Success
 User: N/A Computer: DESKTOP-NFC0HVN
 OpCode: Info
 More Information: [Event Log Online Help](#)

Copy **Close**

6422(S): A device was enabled.

Event Description:

This event generates every time specific device was enabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6422</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13316</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-14T22:37:50.036050900Z" />
<EventRecordID>486</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="408" />
<Channel>Security</Channel>
<Computer>DESKTOP-NFC0HVN</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DESKTOP-NFC0HVN$</Data>
<Data Name="SubjectDomainName">WORKGROUP</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="DeviceId">USB\VID_138A&PID_0017\ffbc12c950a0</Data>
```

```
<Data Name="DeviceDescription">Synaptics FP Sensors (WBF) (PID=0017)</Data>
<Data Name="ClassId">{53D29EF7-377C-4D14-864B-EB3A85769359}</Data>
<Data Name="ClassName">Biometric</Data>
<Data Name="HardwareIds">USB\VID_138A&PID_0017&REV_0078 USB\VID_138A&PID_0017</Data>
<Data Name="CompatibleIds">USB\Class_FF&SubClass_00&Prot_00 USB\Class_FF&SubClass_00 USB\Class_FF</Data>
<Data Name="LocationInformation">Port_#0002.Hub_#0004</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Field Descriptions:

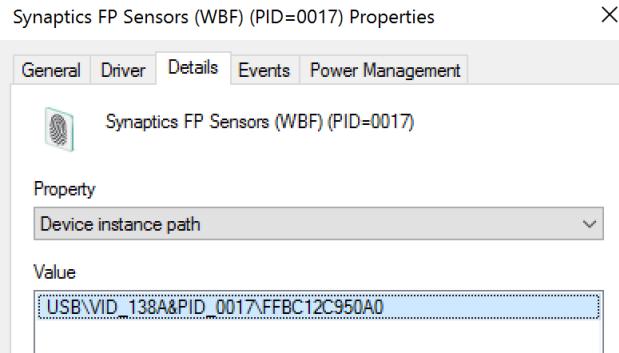
Subject:

- **Security ID** [Type = SID]: SID of account that enabled the device. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

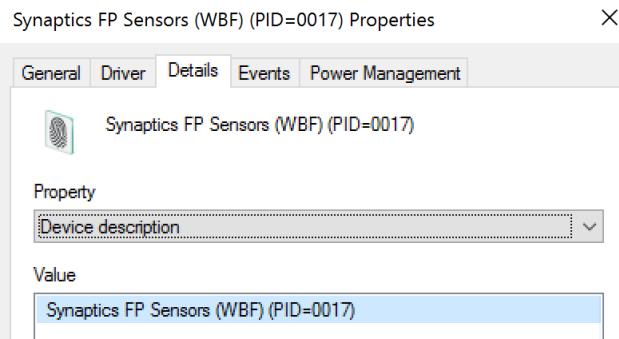
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that enabled the device.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

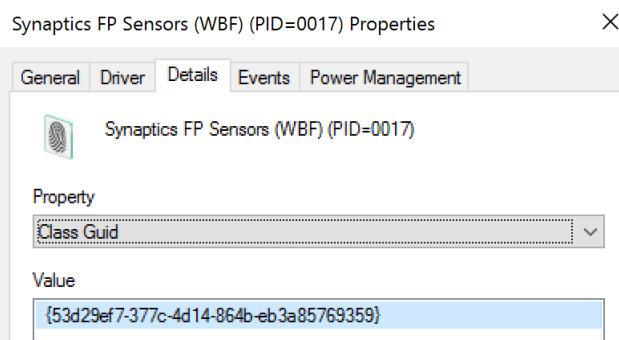
Device ID [Type = UnicodeString]: **“Device instance path”** attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



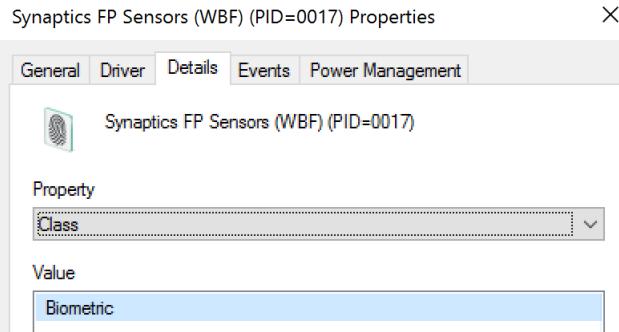
Device Name [Type = UnicodeString]: “**Device description**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



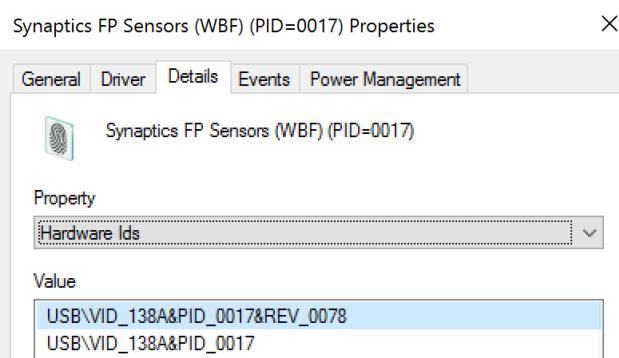
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



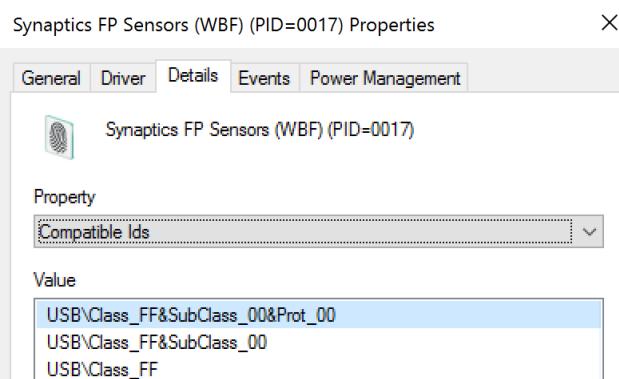
Class Name [Type = UnicodeString]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



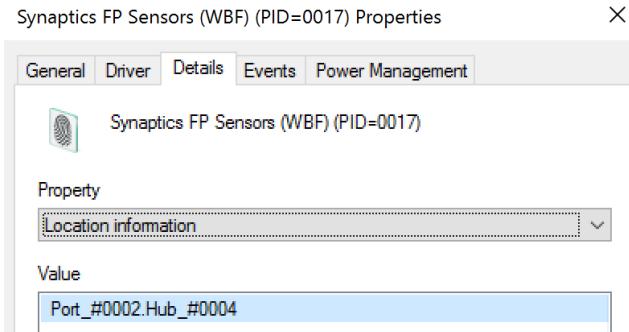
Hardware IDs [Type = UnicodeString]: “**Hardware IDs**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6422(S): A device was enabled.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- You can use this event to track the events and event information shown in the following table by using the listed fields:

Event and event information to monitor	Field to use
Device enable events, Device Instance Path	“ Device ID ”
Device enable events, Device Description	“ Device Name ”
Device enable events, Class GUID	“ Class ID ”
Device enable events, Hardware IDs	“ Hardware IDs ”
Device enable events, Compatible IDs	“ Compatible IDs ”
Device enable events, Location information	“ Location Information ”

6423(S): The installation of this device is forbidden by system policy.

Event Properties - Event 6423, Microsoft Windows security auditing.

General Details

The installation of this device is forbidden by system policy.

Subject: Security ID: SYSTEM

Device ID: USB\VID_04F3&PID_012D\7&1E3A8971&0&2

Device Name: Touchscreen

Class ID: {00000000-0000-0000-0000-000000000000}

Class Name:

Hardware IDs:

- USB\VID_04F3&PID_012D&REV_0013
- USB\VID_04F3&PID_012D

Compatible IDs:

- USB\Class_03&SubClass_00&Prot_00
- USB\Class_03&SubClass_00
- USB\Class_03

Location Information:

Port_#0002.Hub_#0004

Log Name: Security

Source: Microsoft Windows security Logged: 11/14/2015 2:49:34 PM

Event ID: 6423 Task Category: Plug and Play Events

Level: Information Keywords: Audit Success

User: N/A Computer: DESKTOP-NFC0HVN

OpCode: Info

More Information: [Event Log Online Help](#)

Copy **Close**

Event Description:

This event generates every time installation of this device is forbidden by system policy.

Device installation restriction group policies are located here: \Computer Configuration\Administrative Templates\System\Device Installation\Device Installation Restrictions. If one of the policies restricts installation of a specific device, this event will be generated.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6423</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13316</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-14T22:49:34.647975900Z" />
<EventRecordID>488</EventRecordID>
<Correlation />
<Execution ProcessID="828" ThreadID="1924" />
<Channel>Security</Channel>
<Computer>DESKTOP-NFC0HVN</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DESKTOP-NFC0HVN$</Data>
<Data Name="SubjectDomainName">WORKGROUP</Data>

<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="DeviceId">USB\VID_04F3&PID_012D\7&1E3A8971&0&2</Data>
<Data Name="DeviceDescription">Touchscreen</Data>
<Data Name="ClassId">{00000000-0000-0000-0000-000000000000}</Data>
<Data Name="ClassName" />
<Data Name="HardwareIds">USB\VID_04F3&PID_012D&REV_0013 USB\VID_04F3&PID_012D</Data>
<Data Name="CompatibleIds">USB\Class_03&SubClass_00&Prot_00 USB\Class_03&SubClass_00 USB\Class_03</Data>
```

```
<Data Name="LocationInformation">Port_#0002.Hub_#0004</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Field Descriptions:

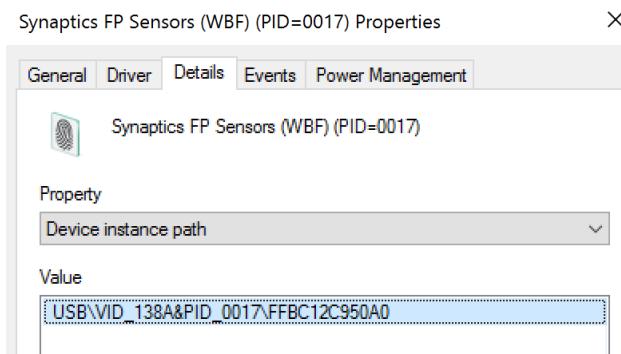
Subject:

- **Security ID** [Type = SID]: SID of account that forbids the device installation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

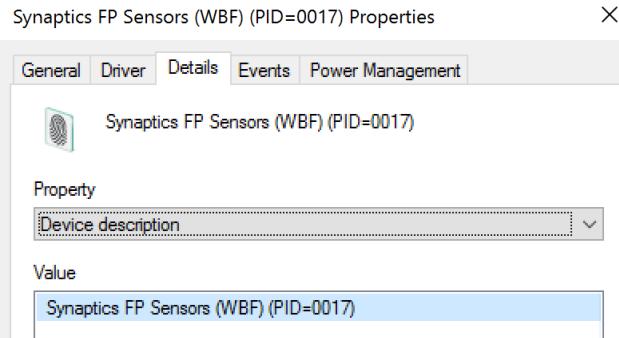
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that forbids the device installation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

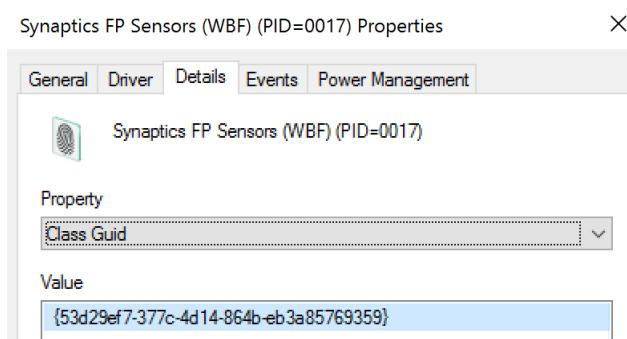
Device ID [Type = UnicodeString]: “**Device instance path**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



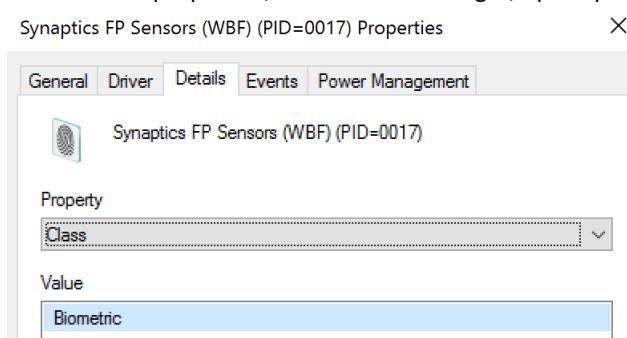
Device Name [Type = UnicodeString]: “**Device description**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



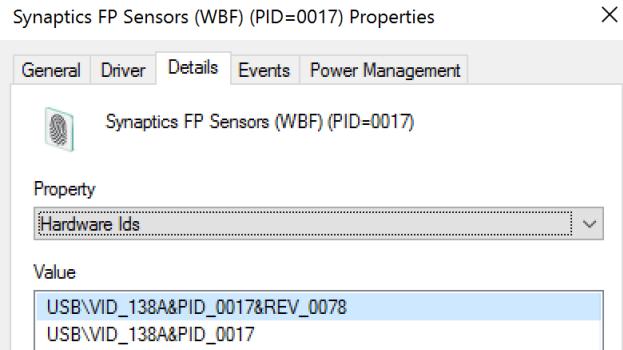
Class ID [Type = UnicodeString]: “**Class Guid**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



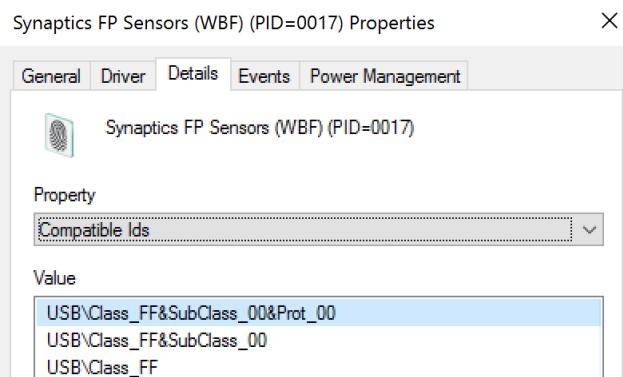
Class Name [Type = UnicodeString]: “**Class**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



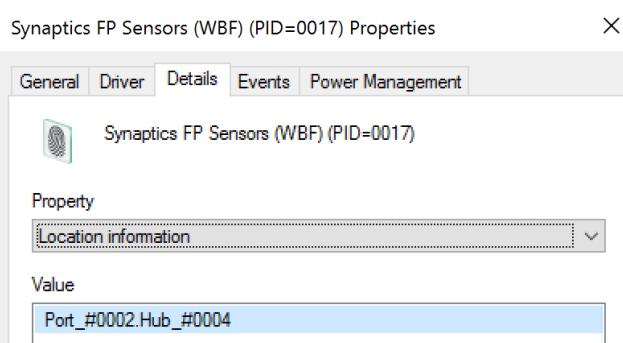
Hardware IDs [Type = UnicodeString]: “**Hardware Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Compatible IDs [Type = UnicodeString]: “**Compatible Ids**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Location Information [Type = UnicodeString]: “**Location information**” attribute of device. To see device properties, start Device Manager, open specific device properties, and click “Details”:



Security Monitoring Recommendations:

For 6423(S): The installation of this device is forbidden by system policy.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you want to track device installation policy violations then you need to track every event of this type.
- Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- You can use this event to track the policy violations and related information shown in the following table by using the listed fields:

Policy violation and related information to monitor	Field to use
Device installation policy violations, Device Instance Path	“ Device ID ”
Device installation policy violations, Device Description	“ Device Name ”
Device installation policy violations, Class GUID	“ Class ID ”
Device installation policy violations, Hardware IDs	“ Hardware IDs ”
Device installation policy violations, Compatible IDs	“ Compatible IDs ”
Device installation policy violations, Location information	“ Location Information ”

6424(S): The installation of this device was allowed, after having previously been forbidden by policy.

This event occurs rarely, and in some situations may be difficult to reproduce.

Required Server Roles: None.

Minimum OS Version: Windows 10 [Version 1511].

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

Audit Process Creation

Audit Process Creation determines whether the operating system generates audit events when a process is created (starts).

These audit events can help you track user activity and understand how a computer is being used. Information includes the name of the program or the user that created the process.

Event volume: Low to Medium, depending on system usage.

This subcategory allows you to audit events generated when a process is created or starts. The name of the application and user that created the process is also audited.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>It is typically useful to collect Success auditing information for this subcategory for forensic investigations, to find information who, when and with which options\parameters ran specific process.</p> <p>Additionally, you can analyse process creation events for elevated credentials use, potential malicious process names and so on.</p> <p>The event volume is typically medium-high level, depending on the process activity on the computer.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	Yes	No	Yes	No	<p>It is typically useful to collect Success auditing information for this subcategory for forensic investigations, to find information who, when and with which options\parameters ran specific process.</p> <p>Additionally, you can analyse process creation events for elevated credentials use, potential malicious process names and so on.</p> <p>The event volume is typically medium-high level, depending on the process activity on the computer.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	Yes	No	Yes	No	<p>It is typically useful to collect Success auditing information for this subcategory for forensic investigations, to find information who, when and with which options\parameters ran specific process.</p> <p>Additionally, you can analyse process creation events for elevated credentials use, potential malicious process names and so on.</p> <p>The event volume is typically medium-high level, depending on the process activity on the computer.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4688\(S\)](#): A new process has been created.
- [4696\(S\)](#): A primary token was assigned to process.

4688(S): A new process has been created.

Event Properties - Event 4688, Microsoft Windows security auditing.

A new process has been created.

Creator Subject:	SYSTEM
Security ID:	WIN-GG82ULGC9GO\$
Account Name:	CONTOSO
Account Domain:	
Logon ID:	0x3E7
Target Subject:	CONTOSO\dadmin
Security ID:	dadmin
Account Name:	CONTOSO
Account Domain:	
Logon ID:	0x4ASAFO
Process Information:	
New Process ID:	0x2bc
New Process Name:	C:\Windows\System32\rundll32.exe
Token Elevation Type:	%%1938
Mandatory Label:	Mandatory Label\Medium Mandatory Level
Creator Process ID:	0xe74
Creator Process Name:	C:\Windows\explorer.exe
Process Command Line:	

Token Elevation Type indicates the type of token that was assigned to the new process in accordance with User Account Control policy.

Type 1 is a full token with no privileges removed or groups disabled. A full token is only used if User Account Control is disabled or if the user is the built-in Administrator account or a service account.

Type 2 is an elevated token with no privileges removed or groups disabled. An elevated token is used when User Account Control is enabled and the user chooses to start the program using Run as administrator. An elevated token is also used when an application is configured to always require administrative privilege or to always require maximum privilege, and the user is a member of the Administrators group.

Type 3 is a limited token with administrative privileges removed and administrative groups disabled. The limited token is used when User Account Control is enabled, the application does not require administrative privilege, and the user does not choose to start the program using Run as administrator.

Event Description:
This event generates every time a new process starts.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4688</EventID>
  <Version>2</Version>
  <Level>0</Level>
  <Task>13312</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-11-12T02:24:52.377352500Z" />
  <EventRecordID>2814</EventRecordID>
  <Correlation />
  <Execution ProcessID="4" ThreadID="400" />
  <Channel>Security</Channel>
  <Computer>WIN-GG82ULGC9GO.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">WIN-GG82ULGC9GO$</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x3e7</Data>
  <Data Name="NewProcessId">0x2bc</Data>

```

```

<Data Name="NewProcessName">C:\Windows\System32\rundll32.exe</Data>
<Data Name="TokenElevationType">%%1938</Data>
<Data Name="ProcessId">0xe74</Data>
<Data Name="CommandLine" />
<Data Name="TargetUserSid">S-1-5-21-1377283216-344919071-3415362939-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>

```

```
<Data Name="TargetLogonId">0x4a5af0</Data>
<Data Name="ParentProcessName">C:\Windows\explorer.exe</Data>
<Data Name="MandatoryLabel">S-1-16-8192</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.
- 1 - Windows Server 2012 R2, Windows 8.1.
 - Added “Process Command Line” field.
- 2 - Windows 10.
 - **Subject** renamed to **Creator Subject**.
 - Added “**Target Subject**” section.
 - Added “**Mandatory Label**” field.
 - Added “**Creator Process Name**” field.

Field Descriptions:

Creator Subject [Value for versions 0 and 1 – **Subject**]:

- **Security ID** [Type = SID]: SID of account that requested the “create process” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create process” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624: An account was successfully logged on.](#)”

Target Subject [Version 2]:

This event includes the principal of the process creator, but this is not always sufficient if the target context is different from the creator context. In that situation, the subject specified in the process termination event does not match the subject in the process creation event even though both events refer to the same process ID. Therefore, in addition to including the creator of the process, we will also include the target principal when the creator and target do not share the same logon.

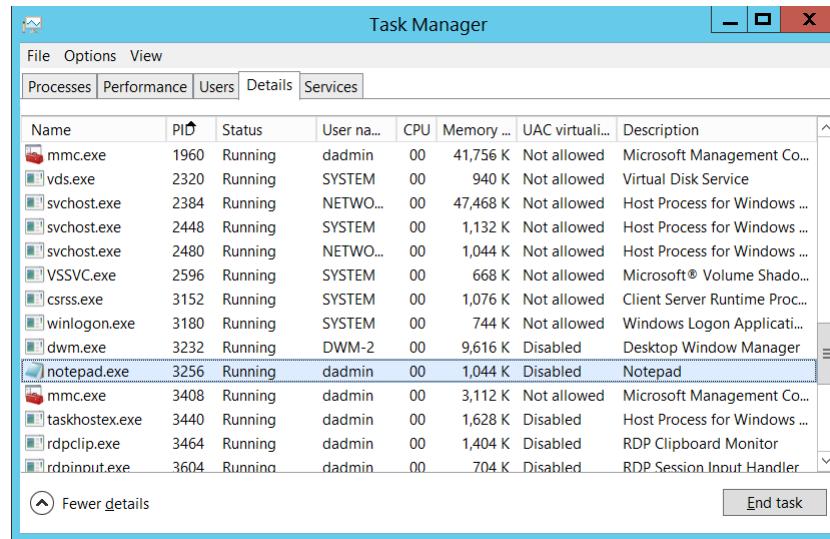
- **Security ID [Type = SID] [Version 2]:** SID of target account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString] [Version 2]:** the name of the target account.
- **Account Domain [Type = UnicodeString] [Version 2]:** target account's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID [Type = HexInt64] [Version 2]:** hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Process Information:

- **New Process ID [Type = Pointer]:** hexadecimal Process ID of the new process. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



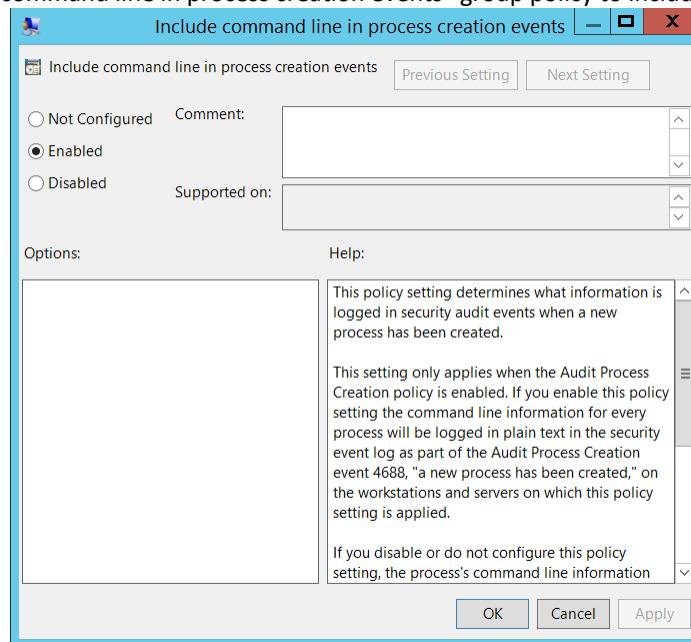
If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

- **New Process Name** [Type = UnicodeString]: full path and the name of the executable for the new process.
- **Token Elevation Type** [Type = UnicodeString]:
 - **TokenElevationTypeDefault (1)**: Type 1 is a full token with no privileges removed or groups disabled. A full token is only used if User Account Control is disabled or if the user is the built-in Administrator account (for which UAC disabled by default), service account or local system account.
 - **TokenElevationTypeFull (2)**: Type 2 is an elevated token with no privileges removed or groups disabled. An elevated token is used when User Account Control is enabled and the user chooses to start the program using Run as administrator. An elevated token is also used when an application is configured to always require administrative privilege or to always require maximum privilege, and the user is a member of the Administrators group.
 - **TokenElevationTypeLimited (3)**: Type 3 is a limited token with administrative privileges removed and administrative groups disabled. The limited token is used when User Account Control is enabled, the application does not require administrative privilege, and the user does not choose to start the program using Run as administrator.

- **Mandatory Label** [Version 2] [Type = SID]: SID of [integrity label](#) which was assigned to the new process. Can have one of the following values:

SID	RID	RID label	Meaning
S-1-16-0	0x00000000	SECURITY_MANDATORY_UNTRUSTED RID	Untrusted.
S-1-16-4096	0x00001000	SECURITY_MANDATORY_LOW RID	Low integrity.
S-1-16-8192	0x00002000	SECURITY_MANDATORY_MEDIUM RID	Medium integrity.
S-1-16-8448	0x00002100	SECURITY_MANDATORY_MEDIUM_PLUS RID	Medium high integrity.
S-1-16-12288	0x00003000	SECURITY_MANDATORY_HIGH RID	High integrity.
S-1-16-16384	0x00004000	SECURITY_MANDATORY_SYSTEM RID	System integrity.
S-1-16-20480	0x00005000	SECURITY_MANDATORY_PROTECTED_PROCESS RID	Protected process.

- **Creator Process ID** [Type = Pointer]: hexadecimal Process ID of the process which ran the new process. If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.
You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.
- **Creator Process Name** [Version 2] [Type = UnicodeString]: full path and the name of the executable for the process.
- **Process Command Line** [Version 1, 2] [Type = UnicodeString]: contains the name of executable and arguments which were passed to it. You must enable “Administrative Templates\System\Audit Process Creation\Include command line in process creation events” group policy to include command line in process creation events:



By default **Process Command Line** field is empty.

Security Monitoring Recommendations:

For 4688(S): A new process has been created.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor all events with the “ Creator Subject\Security ID ” or “ Target Subject\Security ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting	When you monitor for anomalies or malicious actions, use the “ Creator Subject\Security

anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.

Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.

Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.

Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.

External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).

Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.

Account naming conventions: Your organization might have specific naming conventions for account names.

ID” or “Target Subject\Security ID (with other information) to monitor how or when a particular account is being used.

Monitor all events with the **“Creator Subject\Security ID”** or **“Target Subject\Security ID”** that corresponds to the accounts that should never be used.

If this event corresponds to a “whitelist-only” action, review the **“Creator Subject\Security ID”** and **“Target Subject\Security ID”** for accounts that are outside the whitelist.

If this event corresponds to an action you want to monitor for certain account types, review the **“Creator Subject\Security ID”** or **“Target Subject\Security ID”** to see whether the account type is as expected.

Monitor the specific events for the **“Creator Subject\Security ID”** or **“Target Subject\Security ID”** corresponding to accounts from another domain or “external” accounts.

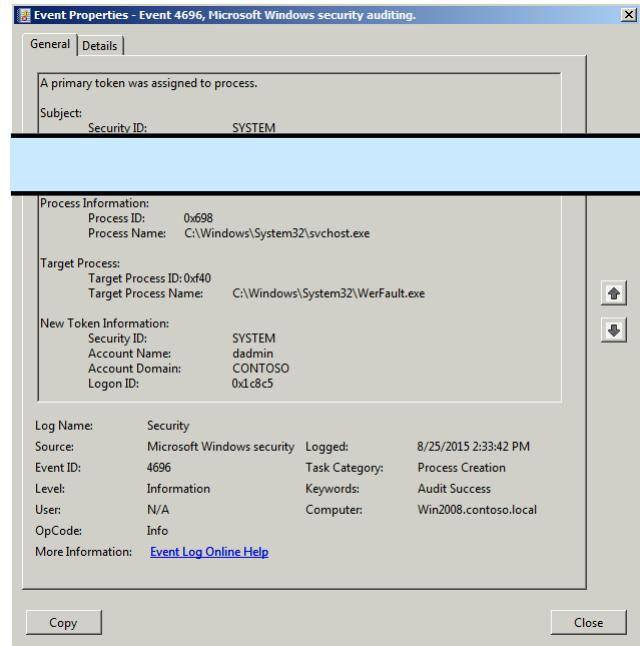
Monitor the target **Computer:** (or other target device) for actions performed by the **“Creator Subject\Security ID”** or **“Target Subject\Security ID”** that you are concerned about.

Monitor **“Creator Subject\Security ID”** or **“Target Subject\Security ID”** for names that don’t comply with naming conventions.

- If you have a pre-defined **“New Process Name”** or **“Creator Process Name”** for the process reported in this event, monitor all events with **“New Process Name”** or **“Creator Process Name”** not equal to your defined value.
- You can monitor to see if **“New Process Name”** or **“Creator Process Name”** is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in process names (for example **“mimikatz”** or **“cain.exe”**), check for these substrings in **“New Process Name”** or **“Creator Process Name.”**
- It can be unusual for a process to run using a local account in either **Creator Subject\Security ID** or in **Target Subject\Security ID**.
- Monitor for **Token Elevation Type** with value **TokenElevationTypeDefault (1)** when **Subject\Security ID** lists a real user account, for example when **Account Name** doesn’t contain the \$ symbol. Typically this means that UAC is disabled for this account for some reason.
- Monitor for **Token Elevation Type** with value **TokenElevationTypeDefault (2)** on standard workstations, when **Subject\Security ID** lists a real user account, for example when **Account Name** doesn’t contain the \$ symbol. This means that a user ran a program using administrative privileges.
- You can also monitor for **Token Elevation Type** with value **TokenElevationTypeDefault (2)** on standard workstations, when a computer object was used to run the process, but that computer object is not the same computer where the event occurs.

- If you need to monitor all new processes with a specific Mandatory Label, for example S-1-16-20480 (Protected process), check the “**Mandatory Label**” in this event.

4696(S): A primary token was assigned to process.



Event Description:

This event generates every time a process runs using the non-current access token, for example, UAC elevated token, RUN AS different user actions, scheduled task with defined user, services, and so on.

IMPORTANT: this event is deprecated starting from Windows 7 and Windows 2008 R2.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-a5ba-3e3b0328c30d}" />
  <EventID>4696</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13312</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-25T21:33:42.401Z" />
  <EventRecordID>561</EventRecordID>
  <Correlation />

```

```

<Execution ProcessID="4" ThreadID="88" />
<Channel>Security</Channel>
<Computer>Win2008.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">WIN2008$</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x3e7</Data>
  <Data Name="TargetUserSid">S-1-5-18</Data>
  <Data Name="TargetUserName">dadmin</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetLogonId">0x1c8c5</Data>
  <Data Name="TargetProcessId">0xf40</Data>

```

```
<Data Name="TargetProcessName">C:\Windows\System32\WerFault.exe</Data>
<Data Name="ProcessId">0x698</Data>
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: this event is deprecated starting from Windows 7 and Windows 2008 R2.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

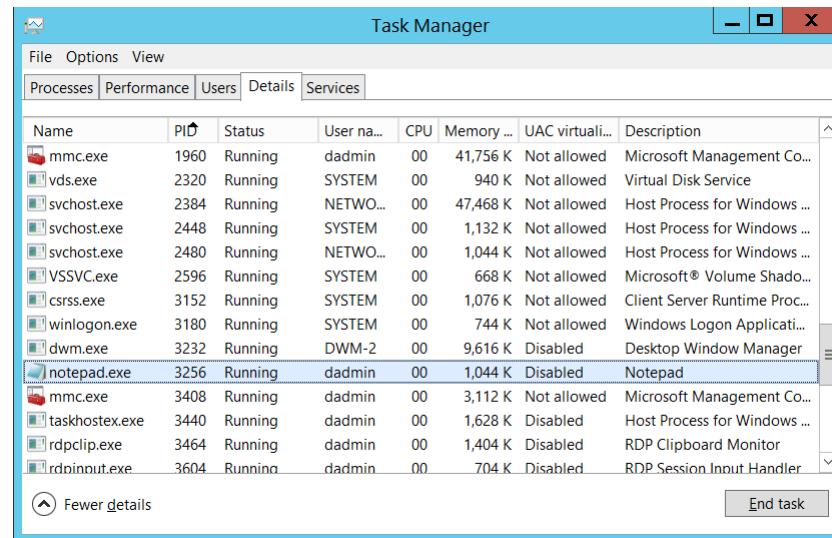
- **Security ID [Type = SID]:** SID of account that requested the “assign token to process” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]:** the name of the account that requested the “assign token to process” operation.
- **Account Domain [Type = UnicodeString]:** subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID [Type = HexInt64]:** hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Process Information:

- **Process ID [Type = Pointer]:** hexadecimal Process ID of the process which started the new process with the new security token. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process which ran the new process with new security token.

Target Process:

- **Target Process ID** [Type = Pointer]: hexadecimal Process ID of the new process with new security token. If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Target Process Name** [Type = UnicodeString]: full path and the name of the executable for the new process.

New Token Information:

- **Security ID** [Type = SID]: SID of account through which the security token will be assigned to the new process. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account through which the security token will be assigned to the new process.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL

- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Security Monitoring Recommendations:

For 4696(S): A primary token was assigned to process.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Subject\Security ID ” or “ New Token Information\Security ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” or “ New Token Information\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Security ID ” or “ New Token Information\Security ID ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” and “ New Token Information\Security ID ” for accounts that are outside the whitelist.
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” or “ New Token Information\Security ID ” to see whether the account type is as expected.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Subject\Security ID ” or “ New Token Information\Security ID ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.	Monitor the target Computer : (or other target device) for actions performed by the “ Subject\Security ID ” or “ New Token Information\Security ID ” that you are concerned about.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Security ID**” or “**New Token Information\Security ID**” for names that don’t comply with naming conventions.

- If you have a pre-defined “**Process Name**” or “**Target Process Name**” for the process reported in this event, monitor all events with “**Process Name**” or “**Target Process Name**” not equal to your defined value.
- You can monitor to see if “**Process Name**” or “**Target Process Name**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in process names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Process Name**” or “**Target Process Name**”.
- It can be uncommon if process runs using local account.

Audit Process Termination

Audit Process Termination determines whether the operating system generates audit events when process has exited.

Success audits record successful attempts and Failure audits record unsuccessful attempts.

This policy setting can help you track user activity and understand how the computer is used.

Event volume: Low to Medium, depending on system usage.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	No	No	IF	No	<p>IF - This subcategory typically is not as important as Audit Process Creation subcategory. Using this subcategory you can, for example get information about for how long process was run in correlation with 4688 event. If you have a list of critical processes that run on some computers, you can enable this subcategory to monitor for termination of these critical processes.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	No	No	IF	No	<p>IF - This subcategory typically is not as important as Audit Process Creation subcategory. Using this subcategory you can, for example get information about for how long process was run in correlation with 4688 event. If you have a list of critical processes that run on some computers, you can enable this subcategory to monitor for termination of these critical processes.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	No	No	IF	No	<p>IF - This subcategory typically is not as important as Audit Process Creation subcategory. Using this subcategory you can, for example get information about for how long process was run in correlation with 4688 event. If you have a list of critical processes that run on some computers, you can enable this subcategory to monitor for termination of these critical processes.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

Event Properties - Event 4689, Microsoft Windows security audit... X

<input checked="" type="checkbox"/>	General	Details
A process has exited.		
Subject:	Security ID:	CONTOSO\dadmin

- [4689\(S\)](#): A process has exited.

4689(S): A process has exited.

Event Description:

This event generates every time a process has exited.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Process Information:	Process ID: 0xfb0
Process Name:	C:\Windows\System32\notepad.exe
Exit Status:	0x0
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4689
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Up Down

Copy Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4689</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13313</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

```
<TimeCreated SystemTime="2015-08-27T17:13:01.826339500Z" />
<EventRecordID>187030</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="144" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x31365</Data>
<Data Name="Status">0x0</Data>
<Data Name="ProcessId">0xfb0</Data>
<Data Name="ProcessName">C:\Windows\System32\notepad.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “terminate process” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

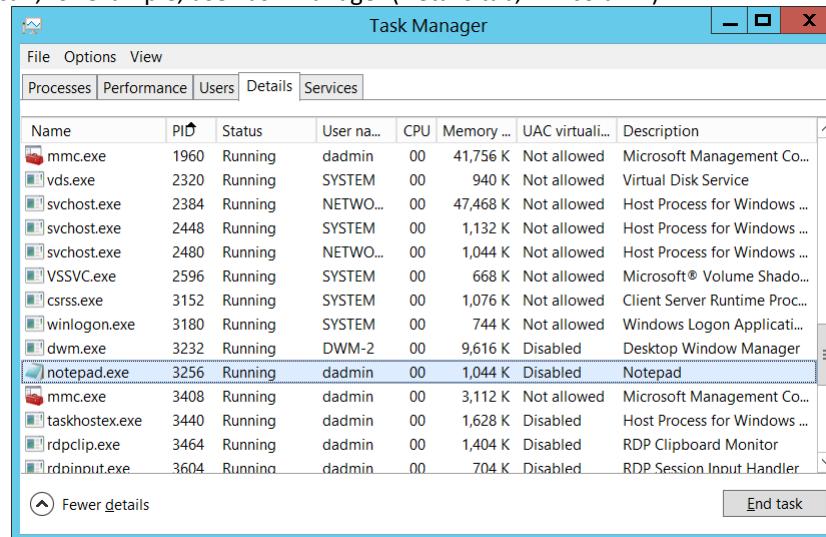
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “terminate process” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the ended/terminated process. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688\(S\)](#): A new process has been created” **New Process ID** on this computer.

- **Process Name** [Type = UnicodeString]: full path and the executable name of the exited/terminated process.
- **Exit Status** [Type = HexInt32]: hexadecimal exit code of exited/terminated process. This exit code is unique for every application, check application documentation for more details. The exit code value for a process reflects the specific convention implemented by the application developer for that process.

Security Monitoring Recommendations:

For 4689(S): A process has exited.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If you have a critical processes list for the computer, with the requirement that these processes must always run and not stop, you can monitor **Process Name** field in [4689](#) events for these process names.

Audit RPC Events

Audit RPC Events determines whether the operating system generates audit events when inbound remote procedure call (RPC) connections are made.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	No	No	Events in this subcategory occur rarely.
Member Server	No	No	No	No	Events in this subcategory occur rarely.
Workstation	No	No	No	No	Events in this subcategory occur rarely.

Events List:

- [5712\(S\)](#): A Remote Procedure Call (RPC) was attempted.

5712(S): A Remote Procedure Call (RPC) was attempted.

It appears that this event never occurs.

Event Schema:

A Remote Procedure Call (RPC) was attempted.

Subject:

*SID:%1
 Name:%2
 Account Domain:%3
 LogonId:%4*

Process Information:

*PID:%5
 Name:%6*

Network Information:

*Remote IP Address:%7
 Remote Port:%8*

RPC Attributes:

*Interface UUID:%9
 Protocol Sequence:%10
 Authentication Service:%11
 Authentication Level:%12*

Required Server Roles: no information.

Minimum OS Version: no information.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

DS Access

Audit Detailed Directory Service Replication

Audit Detailed Directory Service Replication determines whether the operating system generates audit events that contain detailed tracking information about data that is replicated between domain controllers.

This audit subcategory can be useful to diagnose replication issues.

Event volume: These events can create a very high volume of event data on domain controllers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	IF	IF	IF - Events in this subcategory typically have an informational purpose and it is difficult to detect any malicious activity using these events. It's mainly used for Active Directory replication troubleshooting.
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Events List:

- [4928](#)(S, F): An Active Directory replica source naming context was established.
- [4929](#)(S, F): An Active Directory replica source naming context was removed.
- [4930](#)(S, F): An Active Directory replica source naming context was modified.
- [4931](#)(S, F): An Active Directory replica destination naming context was modified.
- [4934](#)(S): Attributes of an Active Directory object were replicated.
- [4935](#)(F): Replication failure begins.
- [4936](#)(S): Replication failure ends.
- [4937](#)(S): A lingering object was removed from a replica.

4928(S, F): An Active Directory replica source naming context was established.

Event Description:

This event generates every time a new Active Directory replica source naming context is established. Failure event generates if an error occurs (**Status Code != 0**).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event Properties - Event 4928, Microsoft Windows security audit...

General Details

An Active Directory replica source naming context was established.

Destination DRA: CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local
Source DRA: CN=NTDS Settings,CN=WIN2012R2,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local
Source Address: ddec0cff-6ceb-4a59-b13f-1724c38a0970.
.msdcos.contoso.local
Naming Context: DC=ForestDnsZones,DC=contoso,DC=local
Options: 368

Log Name: Security
Source: Microsoft Windows security audit
Event ID: 4928
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 8/27/2015 12:15:30 I
Task Category: Detailed Directory S
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4928</EventID>
<Version>0</Version>
```

```
<Level>0</Level>
<Task>14083</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-27T19:15:30.067319300Z" />
<EventRecordID>227065</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="1236" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="DestinationDRA">CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="SourceDRA">CN=NTDS Settings,CN=WIN2012R2,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="SourceAddr">ddec0cff-6ceb-4a59-b13f-1724c38a0970._msdcs.contoso.local</Data>
  <Data Name="NamingContext">DC=ForestDnsZones,DC=contoso,DC=local</Data>
  <Data Name="Options">368</Data>
  <Data Name="StatusCode">0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name.

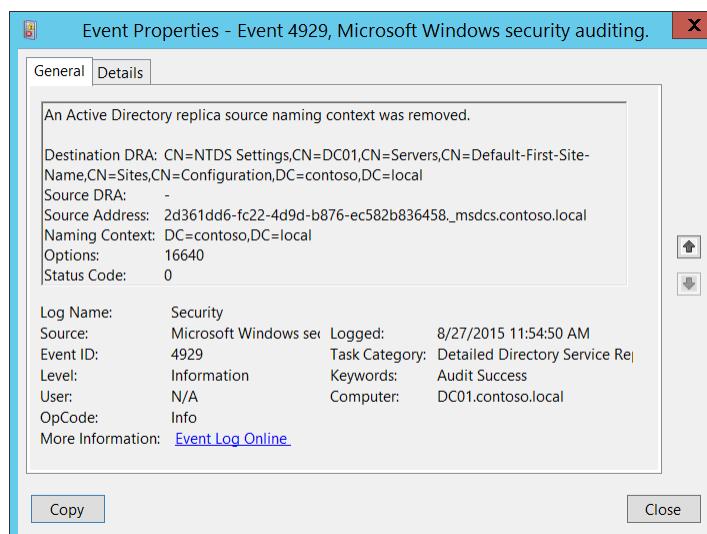
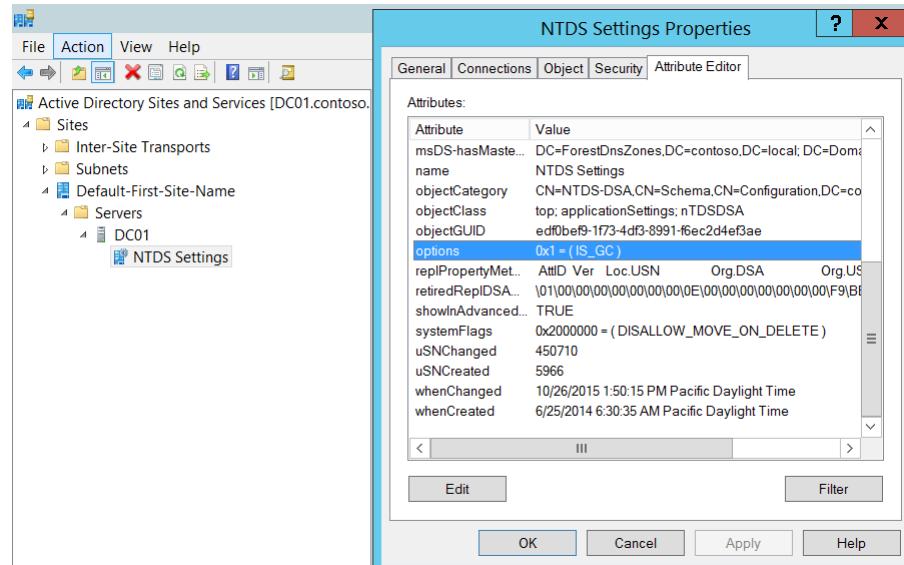
The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Source Address** [Type = UnicodeString]: DNS record of the server from which information or an update was received.
- **Naming Context** [Type = UnicodeString]: naming context to replicate.

The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.

- **Options** [Type = UInt32]: decimal value of [DRS Options](#).



- **Status Code** [Type = UInt32]: if there are no issues or errors, the status code will be 0. If an error happened, you will receive Failure event and Status Code will not be equal to "0". You can check error code meaning here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx)

Security Monitoring Recommendations:

For 4928(S, F): An Active Directory replica source naming context was established.

- Monitor for **Source Address** field, because the source of new replication (new DRA) must be authorized for this action. If you find any unauthorized DRA you should trigger an event.
- This event is typically used for Active Directory replication troubleshooting.

4929(S, F): An Active Directory replica source naming context was removed.

Event Description:

This event generates every time Active Directory replica source naming context was removed. Failure event generates if an error occurs (**Status Code != 0**).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4929</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14083</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-27T18:54:50.446211200Z" />
<EventRecordID>227013</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="2636" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="DestinationDRA">CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="SourceDRA">-</Data>
<Data Name="SourceAddr">2d361dd6-fc22-4d9d-b876-ec582b836458._msdcs.contoso.local</Data>
<Data Name="NamingContext">DC=contoso,DC=local</Data>
<Data Name="Options">16640</Data>
<Data Name="StatusCode">0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

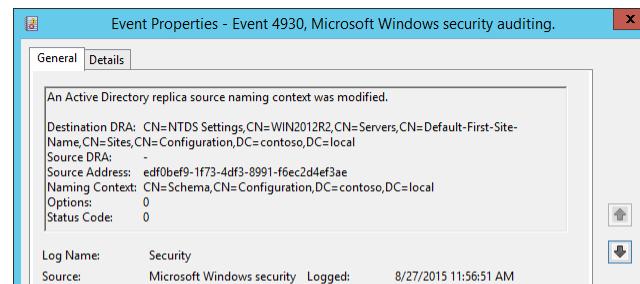
- **Source Address** [Type = UnicodeString]: DNS record of the server from which the “remove” request was received.
- **Naming Context** [Type = UnicodeString]: naming context which was removed.

The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.

- **Options** [Type = UInt32]: decimal value of [DRS Options](#).
- **Status Code** [Type = UInt32]: if there are no issues or errors, the status code will be 0. If an error happened, you will receive Failure event and Status Code will not be equal to “0”. You can check error code meaning here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx)

Security Monitoring Recommendations:

For 4929(S, F): An Active Directory replica source naming context was removed.



4930(S, F): An Active Directory replica source naming context was modified.

Event Description:

This event generates every time Active Directory replica source naming context was modified.

Failure event generates if an error occurs (**Status Code** != 0).

It is not possible to understand what exactly was modified from this event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">

```
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4930</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14083</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-27T18:56:51.474057400Z" />
<EventRecordID>1564</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="1280" />
<Channel>Security</Channel>
<Computer>Win2012r2.corp.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="DestinationDRA">CN=NTDS Settings,CN=WIN2012R2,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="SourceDRA">-</Data>
<Data Name="SourceAddr">edf0bef9-1f73-4df3-8991-f6ec2d4ef3ae</Data>
<Data Name="NamingContext">CN=Schema,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="Options">0</Data>
<Data Name="StatusCode">0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name. Typically equals “-” for this event.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

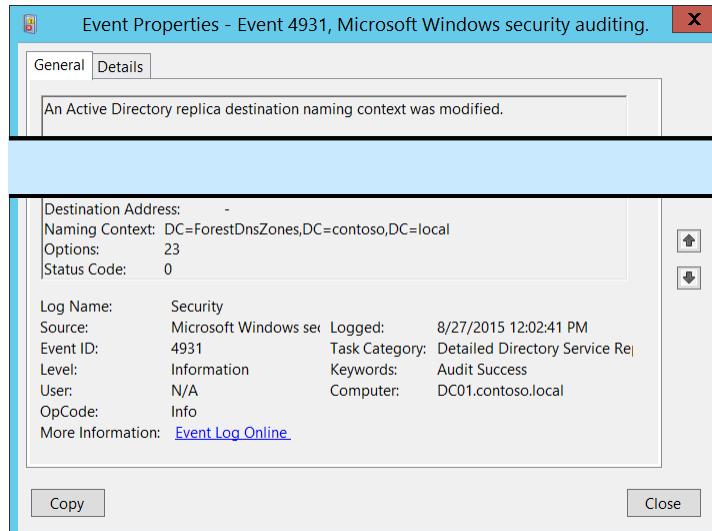
- DC - domainComponent
 - CN - commonName
 - OU - organizationalUnitName
 - O - organizationName
- **Source Address** [Type = UnicodeString]: DNS record of computer from which the modification request was received.
- **Naming Context** [Type = UnicodeString]: naming context which was modified.
- The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.
- **Options** [Type = UInt32]: decimal value of [DRS Options](#).
 - **Status Code** [Type = UInt32]: if there are no issues or errors, the status code will be 0. If an error happened, you will receive Failure event and Status Code will not be equal to "0". You can check error code meaning here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx)

Security Monitoring Recommendations:

For 4930(S, F): An Active Directory replica source naming context was modified.

- Monitor for **Source Address** field, because the source of the request must be authorized for this action. If you find any unauthorized DRA you should trigger an event.
- This event is typically used for Active Directory replication troubleshooting.

4931(S, F): An Active Directory replica destination naming context was modified.

 Event Properties - Event 4931, Microsoft Windows security auditing.

General **Details**

An Active Directory replica destination naming context was modified.

Destination Address:	-
Naming Context:	DC=ForestDnsZones,DC=contoso,DC=local
Options:	23
Status Code:	0
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4931
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

Event Description:

This event generates every time Active Directory replica destination naming context was modified. Failure event generates if an error occurs (**Status Code** != 0). It is not possible to understand what exactly was modified from this event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4931</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14083</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

```
<TimeCreated SystemTime="2015-08-27T19:02:41.563619400Z" />
<EventRecordID>227058</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="2936" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="DestinationDRA">ddec0cff-6ceb-4a59-b13f-1724c38a0970._msdcs.contoso.local</Data>
  <Data Name="SourceDRA">CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="SourceAddr">-</Data>
  <Data Name="NamingContext">DC=ForestDnsZones,DC=contoso,DC=local</Data>
  <Data Name="Options">23</Data>
  <Data Name="StatusCode">0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Destination Address** [Type = UnicodeString]: DNS record of computer to which the modification request was sent.

- **Naming Context** [Type = UnicodeString]: naming context which was modified.

The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.

- **Options** [Type = UInt32]: decimal value of [DRS Options](#).
- **Status Code** [Type = UInt32]: if there are no issues or errors, the status code will be 0. If an error happened, you will receive Failure event and Status Code will not be equal to “0”. You can check error code meaning here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx)

Security Monitoring Recommendations:

For 4931(S, F): An Active Directory replica destination naming context was modified.

- This event is typically used for Active Directory replication troubleshooting.

4934(S): Attributes of an Active Directory object were replicated.

This event generates when attributes of an Active Directory object were replicated.

There is no example of this event in this document.

Event Schema:

Attributes of an Active Directory object were replicated.

Session ID:%1

Object:%2

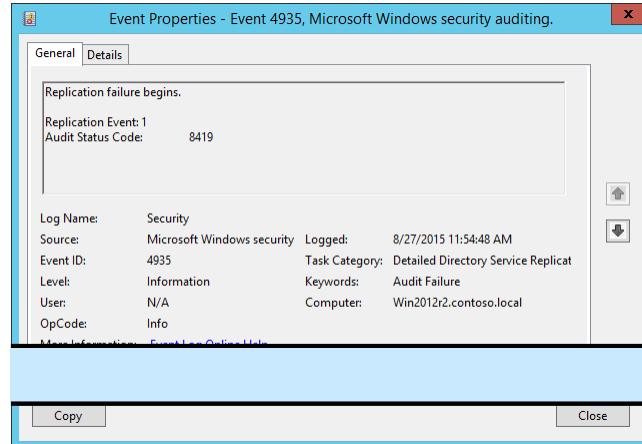
Attribute:%3

Type of change:%4

New Value:%5

USN:%6

Status Code:%7



Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- This event is typically used for Active Directory replication troubleshooting.

4935(F): Replication failure begins.

Event Description:

This event generates when Active Directory replication failure begins.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4935</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14083</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-08-27T18:54:48.758149800Z" />
<EventRecordID>1552</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="524" />
<Channel>Security</Channel>
<Computer>Win2012r2.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ReplicationEvent">1</Data>
<Data Name="AuditStatusCode">8419</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Replication Event [Type = UInt32]: there is no detailed information about this field in this document.

Audit Status Code [Type = UInt32]: there is no detailed information about this field in this document.

Security Monitoring Recommendations:

For 4935(F): Replication failure begins.

- This event is typically used for Active Directory replication troubleshooting.

4936(S): Replication failure ends.

This event generates when Active Directory replication failure ends.

There is no example of this event in this document.

Event Schema:

Replication failure ends.

Replication Event:%1

Audit Status Code:%2

Replication Status Code:%3

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- This event is typically used for Active Directory replication troubleshooting.

4937(S): A lingering object was removed from a replica.

This event generates when a [lingering object](#) was removed from a replica.

There is no example of this event in this document.

Event Schema:

A lingering object was removed from a replica.

Destination DRA:%1

Source DRA:%2

Object:%3

Options:%4

Status Code:%5

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

Audit Directory Service Access

Audit Directory Service Access determines whether the operating system generates audit events when an Active Directory Domain Services (AD DS) object is accessed.

Event volume: High on servers running AD DS role services.

This subcategory allows you to audit when an Active Directory Domain Services (AD DS) object is accessed. It also generates Failure events if access was not granted.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	No	Yes	No	Yes	<p>It is better to track changes to Active Directory objects through the Audit Directory Service Changes subcategory. However, Audit Directory Service Changes doesn't give you information about failed access attempts, so we recommend Failure auditing in this subcategory to track failed access attempts to Active Directory objects.</p> <p>For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. Also, develop an Active Directory auditing policy (SACL design for specific classes, operation types which need to be monitored for specific Organizational Units, and so on) so you can audit only the access attempts that are made to specific important objects.</p>
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Event Properties - Event 4662, Microsoft Windows security auditing. X

[General](#) [Details](#)

An operation was performed on an object.

Subject:

Security ID:	CONTOSO\damain
Account Name:	damain
Account Domain:	CONTOSO
Logon ID:	0x35867

Object:

Object Server:	DS
Object Type:	computer
Object Name:	CN=MyComputer,CN=Users,DC=contoso,DC=local
Handle ID:	0x0

Accesses: DELETE

Access Mask: 0x10000

Properties: DELETE
(bf967a86-0de6-11d0-a285-00aa003049e2)

Additional Information:

Parameter 1:	-
Parameter 2:	-

Log Name: Security

Source: Microsoft Windows security

Event ID: 4662

Level: Information

User: N/A

OpCode: Info

More Information: [Event Log Online](#)

[Copy](#) [Close](#)

Events List:

- [4662\(S, F\)](#): An operation was performed on an object.
- [4661\(S, F\)](#): A handle to an object was requested.

4662(S, F): An operation was performed on an object.

Event Description:

This event generates every time when an operation was performed on an Active Directory object.

This event generates only if appropriate [SACL](#) was set for Active Directory object and performed operation meets this SACL.

If operation failed then Failure event will be generated.

You will get one 4662 for each operation type which was performed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4662</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14080</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-28T01:58:36.894922400Z" />
<EventRecordID>407230</EventRecordID>
<Correlation />
```

```
<Execution ProcessID="520" ThreadID="600" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x35867</Data>
<Data Name="ObjectServer">DS</Data>
<Data Name="ObjectType">%{bf967a86-0de6-11d0-a285-00aa003049e2}</Data>
<Data Name="ObjectName">%{38b3d2e6-9948-4dc1-ae90-1605d5eab9a2}</Data>
<Data Name="OperationType">Object Access</Data>
<Data Name="HandleId">0x0</Data>
<Data Name="AccessList">%%1537</Data>
<Data Name="AccessMask">0x10000</Data>
<Data Name="Properties">%%1537 {bf967a86-0de6-11d0-a285-00aa003049e2}</Data>
<Data Name="AdditionalInfo">-</Data>
<Data Name="AdditionalInfo2" />
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local

- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “DS” value for this event.
- **Object Type** [Type = UnicodeString]: type or class of the object that was accessed. Some of the common Active Directory object types and classes are:
 - container – for containers.
 - user – for users.
 - group – for groups.
 - domainDNS – for domain object.
 - groupPolicyContainer – for group policy objects.

For all possible values of **Object Type** open Active Directory Schema snap-in (see how to enable this snap-in:

[https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

- **Object Name** [Type = UnicodeString]: distinguished name of the object that was accessed.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4661](#): A handle to an object was requested.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Operation:

- **Operation Type** [Type = UnicodeString]: the type of operation which was performed on an object. Typically has “**Object Access**” value for this event.
- **Accesses** [Type = UnicodeString]: the type of access used for the operation. See “Table 9. Active Directory Access Codes and Rights.” for more information.
- **Access Mask** [Type = HexInt32]: hexadecimal mask for the type of access used for the operation. See “Table 9. Active Directory Access Codes and Rights.” for more information.

Access Mask	Access Name	Description
0x1	Create Child	The right to create child objects of the object.
0x2	Delete Child	The right to delete child objects of the object.
0x4	List Contents	The right to list child objects of this object.
0x8	SELF	The right to perform an operation controlled by a validated write access right.
0x10	Read Property	The right to read properties of the object.

0x20	Write Property	The right to write properties of the object.
0x40	Delete Tree	Delete all children of this object, regardless of the permissions of the children. It indicates that "Use Delete Subtree server control" check box was checked during deletion. This operation means that all objects within the subtree, including all delete-protected objects, will be deleted.
0x80	List Object	The right to list a particular object.
0x100	Control Access	Access allowed only after extended rights checks supported by the object are performed. The right to perform an operation controlled by an extended access right.
0x10000	DELETE	The right to delete the object. DELETE also generated when object was moved.
0x20000	READ_CONTROL	The right to read data from the security descriptor of the object, not including the data in the SACL.
0x40000	WRITE_DAC	The right to modify the discretionary access-control list (DACL) in the object security descriptor.
0x80000	WRITE_OWNER	The right to assume ownership of the object. The user must be an object trustee. The user cannot transfer the ownership to other users.
0x100000	SYNCHRONIZE	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state.
0x1000000	ADS_RIGHT_ACCESS_SYS TEM_SECURITY	The right to get or set the SACL in the object security descriptor.
0x80000000	ADS_RIGHT_GENERIC_R EAD	The right to read permissions on this object, read all the properties on this object, list this object name when the parent container is listed, and list the contents of this object if it is a container.
0x40000000	ADS_RIGHT_GENERIC_W RITE	The right to read permissions on this object, write all the properties on this object, and perform all validated writes to this object.
0x20000000	ADS_RIGHT_GENERIC_E XECUTE	The right to read permissions on, and list the contents of, a container object.
0x10000000	ADS_RIGHT_GENERIC_A LL	The right to create or delete child objects, delete a subtree, read and write properties, examine child objects and the object itself, add and remove the object from the directory, and read or write with an extended right.

Table 9. Active Directory Access Codes and Rights.

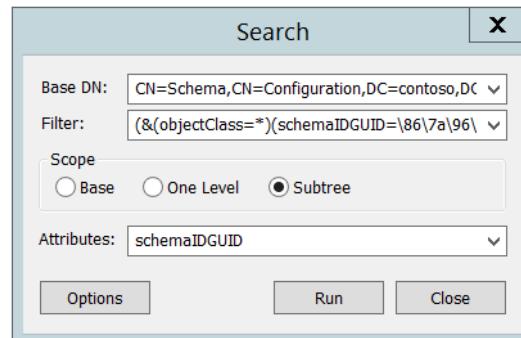
- Properties [Type = UnicodeString]: first part is the type of access that was used. Typically has the same value as **Accesses** field. Second part is a tree of **GUID** values of Active Directory classes or property sets, for which operation was performed.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

To translate this GUID, use the following procedure:

- Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(schemaIDGUID=**GUID**))
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: bf967a86-0de6-11d0-a285-00aa003049e2
 - Take first 3 sections **bf967a86-0de6-11d0**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **867a96bf-e60d-d011**

- Add the last 2 sections without transformation: **867a96bf-e60d-d011-a285-00aa003049e2**
- Delete - : **867a96bfe60dd011a28500aa003049e2**
- Divide bytes with backslashes: **\86\7a\96\bf\e6\0d\d0\11\aa\285\00\aa\00\30\49\ee2**
- Filter example: **(&(objectClass=*)(schemaIDGUID=\86\7a\96\bf\e6\0d\d0\11\aa\285\00\aa\00\30\49\ee2))**
- Scope: Subtree
- Attributes: schemaIDGUID



Sometimes GUID refers to pre-defined Active Directory Property Sets, you can find GUID (**Rights-GUID** field), “property set name” and details here:
[https://msdn.microsoft.com/en-us/library/ms683990\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/ms683990(v=vs.85).aspx).

Here is an example of decoding of **Properties** field:

Properties	Translation
{bf967a86-0de6-11d0-a285-00aa003049e2}	Computer
{91e647de-d96f-4b70-9557-d63ff4f3ccd8}	Private-Information property set
{6617e4ac-a2f1-43ab-b60c-11fbfd1facf05}	ms-PKI-RoamingTimeStamp
{b3f93023-9239-4f7c-b99c-6745d87adbc2}	ms-PKI-DPAPIMasterKeys
{b8dfa744-31dc-4ef1-ac7c-84baf7ef9da7}	ms-PKI-AccountCredentials

Additional Information:

- **Parameter 1** [Type = UnicodeString]: there is no information about this field in this document.
- **Parameter 2** [Type = UnicodeString]: there is no information about this field in this document.

Security Monitoring Recommendations:

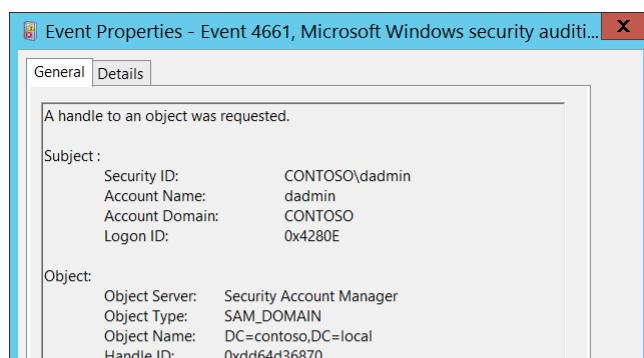
For 4662(S, F): An operation was performed on an object.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor operations attempts to specific Active Directory classes, monitor for **Object Type** field with specific class name. For example, we recommend that you monitor all operations attempts to **domainDNS** class.

- If you need to monitor operations attempts to specific Active Directory objects, monitor for **Object Name** field with specific object name. For example, we recommend that you monitor all operations attempts to “**CN=AdminSDHolder,CN=System,DC=domain,DC=com**” object.
- Some access types are more important to monitor, for example:
 - Write Property
 - Control Access
 - DELETE
 - WRITE_DAC
 - WRITE_OWNER

You can decide to monitor these (or one of these) access types for specific Active Directory objects. To do so, monitor for **Accesses** field with specific access type.

 Event Properties - Event 4661, Microsoft Windows security auditi... X

General Details

A handle to an object was requested.

Subject :

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x4280E

Object:

Object Server:	Security Account Manager
Object Type:	SAM_DOMAIN
Object Name:	DC=contoso,DC=local
Handle ID:	0xdd64d36870

- If you need to monitor operations attempts to specific Active Directory properties, monitor for **Properties** field with specific property GUID.
- Do not forget that **Failure** attempts are also very important to audit. Decide where you want to monitor Failure attempts based on previous recommendations.

4661(S, F): A handle to an object was requested.

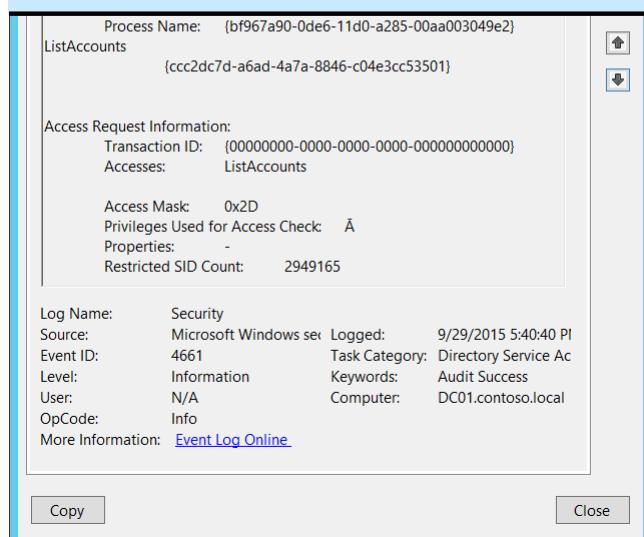
Event Description:

This event indicates that a handle was requested for either an Active Directory object or a Security Account Manager (SAM) object.

If access was declined, then Failure event is generated.

This event generates only if Success auditing is enabled for the [Audit Handle Manipulation](#) subcategory.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Process Name: {bf967a90-0de6-11d0-a285-00aa003049e2}
ListAccounts
(ccc2dc7d-a6ad-4a7a-8846-c04e3cc53501)

Access Request Information:
Transaction ID: {00000000-0000-0000-0000-000000000000}
Accesses: ListAccounts

Access Mask: 0x2D
Privileges Used for Access Check: A
Properties: -
Restricted SID Count: 2949165

Log Name: Security
Source: Microsoft Windows sev Logged: 9/29/2015 5:40:40 PI
Event ID: 4661 Task Category: Directory Service Ac
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

[Copy](#) [Close](#)

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4661</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14080</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-30T00:11:56.547696700Z" />
<EventRecordID>1048009</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="528" />
<Channel>Security</Channel>

```

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x4280e</Data>
<Data Name="ObjectServer">Security Account Manager</Data>
<Data Name="ObjectType">SAM_DOMAIN</Data>
<Data Name="ObjectName">DC=contoso,DC=local</Data>
<Data Name="HandleId">0xdd64d36870</Data>
<Data Name="TransactionId">{00000000-0000-0000-0000-000000000000}</Data>
<Data Name="AccessList">%%5400</Data>
<Data Name="AccessMask">0x2d</Data>
<Data Name="PrivilegeList">Ã</Data>
<Data Name="Properties">-</Data>
<Data Name="RestrictedSidCount">2949165</Data>
<Data Name="ProcessId">0x9000a000d002d</Data>
<Data Name="ProcessName">{bf967a90-0de6-11d0-a285-00aa003049e2} %%5400 {ccc2dc7d-a6ad-4a7a-8846-c04e3cc53501}</Data>
</EventData>
</Event>
```

Required Server Roles: For an Active Directory object, the domain controller role is required. For a SAM object, there is no required role.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested a handle to an object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested a handle to an object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local

- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “**Security Account Manager**” value for this event.
- **Object Type** [Type = UnicodeString]: the type or class of the object that was accessed. The following list contains possible values for this field:
 - SAM_ALIAS - a local group.
 - SAM_GROUP - a group that is not a local group.
 - SAM_USER - a user account.
 - SAM_DOMAIN - a domain. For Active Directory events, this is the typical value.
 - SAM_SERVER - a computer account.
- **Object Name** [Type = UnicodeString]: the name of an object for which access was requested. Depends on **Object Type**. This event can have the following format:
 - SAM_ALIAS – SID of the group.
 - SAM_GROUP – SID of the group.
 - SAM_USER – SID of the account.
 - SAM_DOMAIN – distinguished name of the accessed object.
 - SAM_SERVER – distinguished name of the accessed object.

Process Information: The LDAP API references an LDAP object by its **distinguished name (DN)**. A DN is a sequence of relative distinguished names (RDN) connected by commas.

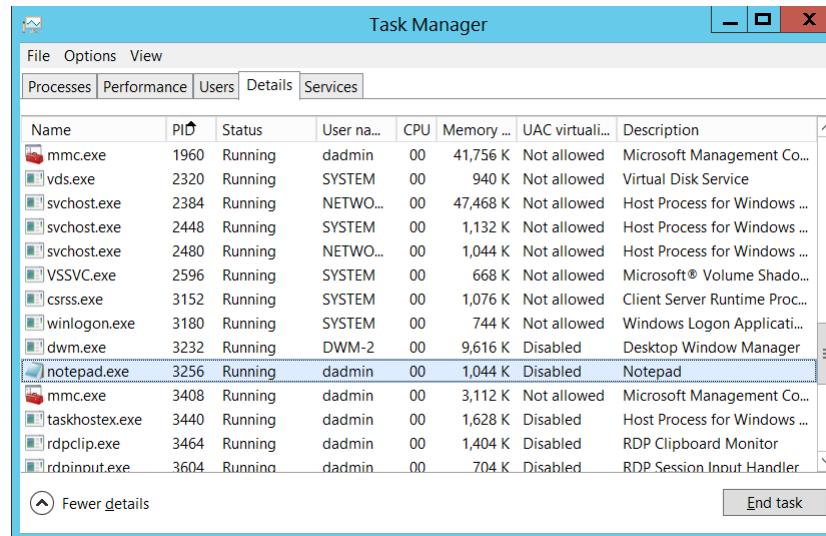
An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4662](#): An operation was performed on an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that requested the handle. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Access Request Information:

- **Transaction ID** [Type = GUID]: unique GUID of the transaction. This field can help you correlate this event with other events that might contain the same the **Transaction ID**, such as “[4660\(S\): An object was deleted.](#)”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Accesses** [Type = UnicodeString]: the list of access rights which were requested by **Subject\Security ID**. These access rights depend on **Object Type**. See “Table 13. File access codes.” for more information about file access rights. For information about SAM object access right use <https://technet.microsoft.com/> or other informational resources.
- **Access Mask** [Type = HexInt32]: hexadecimal mask for the operation that was requested or performed. See “Table 13. File access codes.” for more information about file access rights. For information about SAM object access right use <https://technet.microsoft.com/> or other informational resources.
- **Privileges Used for Access Check** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as “-”. See full list of user privileges in the table below:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	Required to perform backup operations.

		<p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object.</p> <p>The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated</p>

		credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
SeMachineAccountPrivilege	Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE

		<ul style="list-style-type: none"> • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	<p>This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers.</p> <p>With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.</p>
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	<p>Required to gather profiling information for the entire system.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.</p>
SeSystemtimePrivilege	Change the system time	<p>Required to modify the system time.</p> <p>With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.</p>
SeTakeOwnershipPrivilege	Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>
SeTcbPrivilege	Act as part of the operating system	<p>This privilege identifies its holder as part of the trusted computer base.</p> <p>This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.</p>
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivilege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from	Required to undock a laptop.

	docking station	With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

- **Properties** [Type = UnicodeString]: depends on **Object Type**. This field can be empty or contain the list of the object properties that were accessed. See more detailed information in "[4661: A handle to an object was requested](#)" from [Audit SAM](#) subcategory.
- **Restricted SID Count** [Type = UInt32]: Number of [restricted SIDs](#) in the token. Applicable to only specific **Object Types**.

Security Monitoring Recommendations:

For 4661(S, F): A handle to an object was requested.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. You can get almost the same information from "[4662: An operation was performed on an object](#)." There are no additional recommendations for this event in this document.

Audit Directory Service Changes

Audit Directory Service Changes determines whether the operating system generates audit events when changes are made to objects in Active Directory Domain Services (AD DS).

Auditing of directory service objects can provide information about the old and new properties of the objects that were changed.

Audit events are generated only for objects with configured system access control lists ([SACLs](#)), and only when they are accessed in a manner that matches their [SACL](#) settings. Some objects and properties do not cause audit events to be generated due to settings on the object class in the schema.

This subcategory only logs events on domain controllers.

Event volume: High on domain controllers.

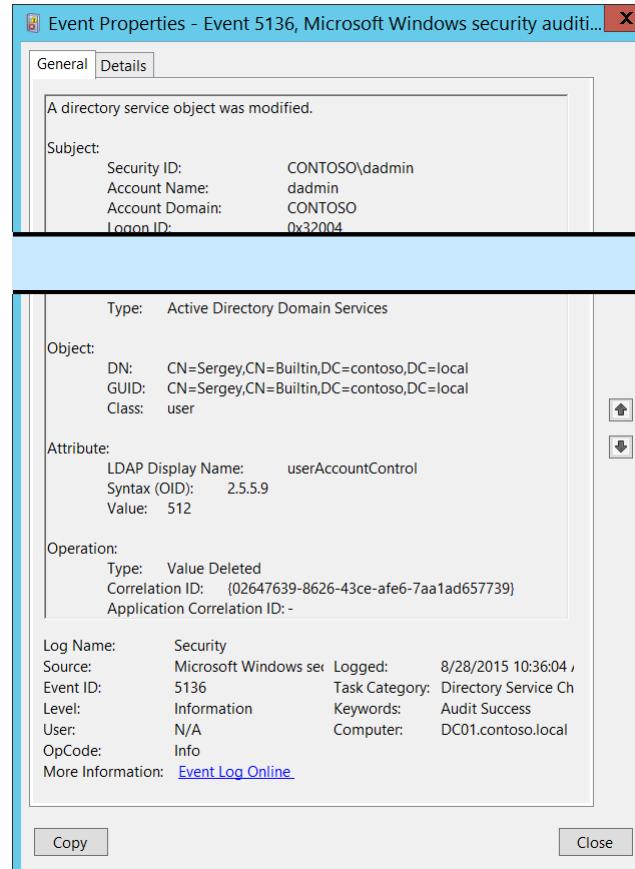
This subcategory triggers events when an Active Directory object was modified, created, undeleted, moved, or deleted.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>It is important to track actions related to high value or critical Active Directory objects, for example, changes to AdminSDHolder container or Domain Admins group objects.</p> <p>This subcategory shows you what actions were performed. If you want to track failed access attempts for Active Directory objects you need to take a look at Audit Directory Service Access subcategory.</p> <p>For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. Also, develop an Active Directory auditing policy (SACL) design for specific classes, operation types which need to be monitored for specific Organizational Units, and so on) so you can audit only the access attempts that are made to specific important objects.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Events List:

- [5136\(S\)](#): A directory service object was modified.
- [5137\(S\)](#): A directory service object was created.
- [5138\(S\)](#): A directory service object was undeleted.
- [5139\(S\)](#): A directory service object was moved.
- [5141\(S\)](#): A directory service object was deleted.

5136(S): A directory service object was modified.

 Event Properties - Event 5136, Microsoft Windows security audit...

General Details

A directory service object was modified.

Subject:

Security ID:	CONTOSO\dam
Account Name:	dam
Account Domain:	CONTOSO
Logon ID:	0x32004

Type: Active Directory Domain Services

Object:

DN:	CN=Sergey,CN=Builtin,DC=contoso,DC=local
GUID:	CN=Sergey,CN=Builtin,DC=contoso,DC=local
Class:	user

Attribute:

LDAP Display Name:	userAccountControl
Syntax (OID):	2.5.5.9
Value:	512

Operation:

Type:	Value Deleted
Correlation ID:	{02647639-8626-43ce-afe6-7aa1ad657739}
Application Correlation ID:	-

Log Name: Security
Source: Microsoft Windows se
Event ID: 5136
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time an Active Directory object is modified.

To generate this event, the modified object must have an appropriate entry in [SACL](#): the “Write” action auditing for specific attributes.

For a change operation you will typically see two 5136 events for one action, with different **Operation\Type** fields: “Value Deleted” and then “Value Added”. “Value Deleted” event typically contains previous value and “Value Added” event contains new value.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5136</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14081</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-28T17:36:04.129472600Z" />
<EventRecordID>410204</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="4020" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```

</System>
- <EventData>
<Data Name="OpCorrelationID">{02647639-8626-43CE-AFE6-7AA1AD657739}</Data>
<Data Name="AppCorrelationID">-</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dam</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x32004</Data>
<Data Name="DSName">contoso.local</Data>
<Data Name="DSType">%14676</Data>
```

```
<Data Name="ObjectDN">CN=Sergey,CN=Builtin,DC=contoso,DC=local</Data>
<Data Name="ObjectGUID">{4FE80A66-5F93-4F73-B215-68678058E613}</Data>
<Data Name="ObjectClass">user</Data>
<Data Name="AttributeLDAPDisplayName">userAccountControl</Data>
<Data Name="AttributeSyntaxOID">2.5.5.9</Data>
<Data Name="AttributeValue">512</Data>
<Data Name="OperationType">%&14675</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “modify object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “modify object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Directory Service:

- **Name** [Type = UnicodeString]: the name of the Active Directory domain where the modified object is located.
- **Type** [Type = UnicodeString]: has “**Active Directory Domain Services**” value for this event.

Object:

- **DN** [Type = UnicodeString]: distinguished name of the object that was modified.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
 - CN - commonName
 - OU - organizationalUnitName
 - O - organizationName
- **GUID** [Type = GUID]: each Active Directory object has globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise but also across the world. GUIDs are assigned to every object created by Active Directory. Each object's GUID is stored in its Object-GUID (**objectGUID**) property.
Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object's GUID will yield results if the user has an account somewhere in the enterprise. In fact, searching for any object by Object-GUID might be the most reliable way of finding the object you want to find. The values of other object properties can change, but the Object-GUID never changes. When an object is assigned a GUID, it keeps that value for life.
Event Viewer automatically resolves **GUID** field to real object.
To translate this GUID, use the following procedure:
 - Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(objectGUID=**GUID**))
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: a6b34ab5-551b-4626-b8ee-2b36b3ee6672
 - Take first 3 sections **a6b34ab5-551b-4626**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **b54ab3a6-1b55-2646**
 - Add the last 2 sections without transformation: **b54ab3a6-1b55-2646-b8ee-2b36b3ee6672**
 - Delete - : **b54ab3a61b552646b8ee2b36b3ee6672**
 - Divide bytes with backslashes: \b5\4a\b3\61\b55\26\46\b8\ee\2b\36\b3\ee\66\72
 - Filter example: (&(objectClass=*)(objectGUID = \b5\4a\b3\61\b55\26\46\b8\ee\2b\36\b3\ee\66\72))
 - Scope: Subtree
 - Attributes: objectGUID
- **Class** [Type = UnicodeString]: class of the object that was modified. Some of the common Active Directory object classes:
 - container – for containers.
 - user – for users.
 - group – for groups.
 - domainDNS – for domain object.
 - groupPolicyContainer – for group policy objects.

For all possible values of this field open Active Directory Schema snap-in (see how to enable this snap-in: [https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx)) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

Attribute:

- **LDAP Display Name** [Type = UnicodeString]: the object attribute that was modified.

[LDAP Display Name](#) is the name used by LDAP clients, such as the ADSI LDAP provider, to read and write the attribute by using the LDAP protocol.

- **Syntax (OID)** [Type = UnicodeString]: The syntax for an attribute defines the storage representation, byte ordering, and matching rules for comparisons of property types. Whether the attribute value must be a string, a number, or a unit of time is also defined. Every attribute of every object is associated with exactly one syntax. The syntaxes are not represented as objects in the schema, but they are programmed to be understood by Active Directory. The allowable syntaxes in Active Directory are predefined.

OID	Syntax Name	Description
2.5.5.0	Undefined	Not a legal syntax.
2.5.5.1	Object(DN-DN)	The fully qualified name of an object in the directory.
2.5.5.2	String(Object-Identifier)	The object identifier.
2.5.5.3	Case-Sensitive String	General String.
2.5.5.4	CaselgnoreString(Teletex)	Differentiates uppercase and lowercase.
2.5.5.5	String(Printable), String(IA5)	Teletex. Does not differentiate uppercase and lowercase.
2.5.5.6	String(Numeric)	Printable string or IA5-String.
2.5.5.7	Object(DN-Binary)	Both character sets are case-sensitive.
2.5.5.8	Boolean	A sequence of digits.
2.5.5.9	Integer, Enumeration	A distinguished name plus a binary large object.
2.5.5.10	String(Octet)	TRUE or FALSE values.
2.5.5.11	String(UTC-Time), String(Generalized-Time)	A 32-bit number or enumeration.
2.5.5.12	String(Unicode)	A string of bytes.
2.5.5.13	Object(Presentation-Address)	UTC Time or Generalized-Time.
2.5.5.14	Object(DN-String)	Unicode string.
2.5.5.15	String(NT-Sec-Desc)	Presentation address.
2.5.5.16	LargeInteger	A DN-String plus a Unicode string.
2.5.5.17	String(Sid)	A Microsoft® Windows NT® Security descriptor.

Table 10. LDAP Attribute Syntax OIDs.

- **Value** [Type = UnicodeString]: the value which was added or deleted, depending on the **Operation\Type** field.

Operation:

- **Type** [Type = UnicodeString]: type of performed operation.
 - **Value Added** – new value added.
 - **Value Deleted** – value deleted (typically “Value Deleted” is a part of change operation).
- **Correlation ID** [Type = GUID]: multiple modifications are often executed as one operation via LDAP. This value allows you to correlate all the modification events that comprise the operation. Just look for other events from current subcategory with the same **Correlation ID**, for example “[5137](#): A directory service object was created.” and “[5139](#): A directory service object was moved.”

GUID is an acronym for ‘Globally Unique Identifier’. It is a 128-bit integer number used to identify resources, activities or instances.

- **Application Correlation ID** [Type = UnicodeString]: always has “-“ value. Not in use.

Security Monitoring Recommendations:

For 5136(S): A directory service object was modified.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor modifications to specific Active Directory objects, monitor for **DN** field with specific object name. For example, we recommend that you monitor all modifications to “**CN=AdminSDHolder,CN=System,DC=domain,DC=com**” object.
- If you need to monitor modifications to specific Active Directory classes, monitor for **Class** field with specific class name. For example, we recommend that you monitor all modifications to **domainDNS** class.
- If you need to monitor modifications to specific Active Directory attributes, monitor for **LDAP Display Name** field with specific attribute name.
- It is better to monitor **Operation\Type = Value Added** events, because you will see the new value of attribute. At the same time you can correlate to previous **Operation\Type = Value Deleted** event with the same **Correlation ID** to see the previous value.

5137(S): A directory service object was created.

Event Properties - Event 5137, Microsoft Windows security audit...

General **Details**

A directory service object was created.

Account Domain: CONTOSO
Logon ID: 0x32004

Directory Service:
Name: contoso.local
Type: Active Directory Domain Services

Object:
DN: cn=Win2000,CN=Users,DC=contoso,DC=local
GUID: CN=Win2000,CN=Users,DC=contoso,DC=local
Class: computer

Operation:
Correlation ID: {4ead68ff-7229-42a4-8c73-aab57169858b}
Application Correlation ID: -

Log Name: Security
Source: Microsoft Windows sec... Logged: 8/28/2015 11:36:26
Event ID: 5137 Task Category: Directory Service Ch
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time an Active Directory object is created.

This event only generates if the parent object has a particular entry in its **SACL**: the “**Create**” action, auditing for specific classes or objects. An example is the “**Create Computer objects**” action auditing for the organizational unit.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>5137</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>14081</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-28T18:36:26.048167500Z" />
  <EventRecordID>410737</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="3156" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />

```

```
</System>
- <EventData>
<Data Name="OpCorrelationID">{4EAD68FF-7229-42A4-8C73-AAB57169858B}</Data>
<Data Name="AppCorrelationID"></Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x32004</Data>
<Data Name="DSName">contoso.local</Data>
<Data Name="DSType">%%14676</Data>
<Data Name="ObjectDN">cn=Win2000,CN=Users,DC=contoso,DC=local</Data>
<Data Name="ObjectGUID">{41D5F7AF-64A2-4985-9A4B-70DAAFC7CCE6}</Data>
<Data Name="ObjectClass">computer</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “create object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Directory Service:

- **Name** [Type = UnicodeString]: the name of an Active Directory domain, where new object is created.

- **Type** [Type = UnicodeString]: has “**Active Directory Domain Services**” value for this event.

Object:

- **DN** [Type = UnicodeString]: distinguished name of the object that was created.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **GUID** [Type = GUID]: each Active Directory object has globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise but also across the world. GUIDs are assigned to every object created by Active Directory. Each object's GUID is stored in its Object-GUID (**objectGUID**) property.

Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object's GUID will yield results if the user has an account somewhere in the enterprise. In fact, searching for any object by Object-GUID might be the most reliable way of finding the object you want to find. The values of other object properties can change, but the Object-GUID never changes. When an object is assigned a GUID, it keeps that value for life.

Event Viewer automatically resolves **GUID** field to real object.

To translate this GUID, use the following procedure:

- Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(objectGUID=**GUID**))
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: a6b34ab5-551b-4626-b8ee-2b36b3ee6672
 - Take first 3 sections a6**b34ab5-551b-4626**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **b54ab3a6-1b55-2646**
 - Add the last 2 sections without transformation: **b54ab3a6-1b55-2646**-b8ee-2b36b3ee6672
 - Delete - : **b54ab3a61b552646b8ee2b36b3ee6672**
 - Divide bytes with backslashes: \b5\4a\b3\46\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72
 - Filter example: (&(objectClass=*)(objectGUID = \b5\4a\b3\46\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72))
 - Scope: Subtree
 - Attributes: objectGUID
- **Class** [Type = UnicodeString]: class of the object that was created. Some of the common Active Directory object classes:
 - container – for containers.
 - user – for users.
 - group – for groups.
 - domainDNS – for domain object.
 - groupPolicyContainer – for group policy objects.

For all possible values of this field open Active Directory Schema snap-in (see how to enable this snap-in: [https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx)) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

Operation:

- **Correlation ID** [Type = GUID]: multiple modifications are often executed as one operation via LDAP. This value allows you to correlate all the modification events that comprise the operation. Just look for other events from current subcategory with the same **Correlation ID**, for example “[5136](#): A directory service object was modified.” and “[5139](#): A directory service object was moved.”

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Application Correlation ID** [Type = UnicodeString]: always has “-“ value. Not in use.

Security Monitoring Recommendations:

For 5137(S): A directory service object was created.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor creation of Active Directory objects with specific classes, monitor for **Class** field with specific class name. For example, we recommend that you monitor all new group policy objects creations: **groupPolicyContainer** class.
- You must set correct auditing access lists (SACLs) for specific classes within Active Directory container to get [5137](#). There is no reason to audit all creation events for all types of Active Directory objects; find the most important locations (organizational units, folders, etc.) and monitor for creation of specific classes only (user, computer, group, etc.).

5138(S): A directory service object was undeleted.

Event Properties - Event 5138, Microsoft Windows security auditing.

General Details

A directory service object was undeleted.

Subject:
Security ID: CONTOSO\dadmin
Account Name: dadmin

Directory Service:
Name: contoso.local
Type: Active Directory Domain Services

Object:
Old DN: CN=Andrei\0ADEL:53511188-bc98-4995-9d78-2d40143c9711,CN=Deleted Objects,DC=contoso,DC=local
New DN:CN=Andrei,CN=Users,DC=contoso,DC=local
GUID: CN=Andrei\0ADEL:53511188-bc98-4995-9d78-2d40143c9711,CN=Deleted Objects,DC=contoso,DC=local
Class: user

Operation:
Correlation ID: {3e2b5ecf-4c35-4c3f-8d82-b8d6f477d846}
Application Correlation ID: -

Log Name: Security
Source: Microsoft Windows sev
Event ID: 5138
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time an Active Directory object is undeleted. It happens, for example, when an Active Directory object was restored from the [Active Directory Recycle Bin](#). This event only generates if the container to which the Active Directory object was restored has a particular entry in its [SACL](#): the “**Create**” action, auditing for specific classes or objects. An example is the “**Create User objects**” action.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5138</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14081</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-02T04:34:20.611082300Z" />
<EventRecordID>229336</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="544" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="OpCorrelationID">{3E2B5ECF-4C35-4C3F-8D82-B8D6F477D846}</Data>
<Data Name="AppCorrelationID"></Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3be49</Data>
<Data Name="DSName">contoso.local</Data>
<Data Name="DSType">%14676</Data>
```

```
<Data Name="OldObjectDN">CN=Andrei\0ADEL:53511188-bc98-4995-9d78-2d40143c9711,CN=Deleted Objects,DC=contoso,DC=local</Data>
<Data Name="NewObjectDN">CN=Andrei,CN=Users,DC=contoso,DC=local</Data>
<Data Name="ObjectGUID">{53511188-BC98-4995-9D78-2D40143C9711}</Data>
<Data Name="ObjectClass">user</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested that the object be undeleted or restored. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: name of account that requested that the object be undeleted or restored.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Directory Service:

- **Name** [Type = UnicodeString]: the name of an Active Directory domain, where the object was undeleted.
- **Type** [Type = UnicodeString]: has "**Active Directory Domain Services**" value for this event.

Object:

- **Old DN** [Type = UnicodeString]: Old distinguished name of undeleted object. It will point to [Active Directory Recycle Bin](#) folder, in case if it was restored from it.

The LDAP API references an LDAP object by its **distinguished name (DN)**. A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName

- O - organizationName
- New DN [Type = UnicodeString]: New distinguished name of undeleted object. The Active Directory container to which the object was restored.
 - GUID [Type = GUID]: each Active Directory object has globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise but also across the world. GUIDs are assigned to every object created by Active Directory. Each object's GUID is stored in its Object-GUID (**objectGUID**) property.
Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object's GUID will yield results if the user has an account somewhere in the enterprise. In fact, searching for any object by Object-GUID might be the most reliable way of finding the object you want to find. The values of other object properties can change, but the Object-GUID never changes. When an object is assigned a GUID, it keeps that value for life.
Event Viewer automatically resolves **GUID** field to real object.
To translate this GUID, use the following procedure:
 - Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(objectGUID=**GUID**))
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: a6b34ab5-551b-4626-b8ee-2b36b3ee6672
 - Take first 3 sections a6**b34ab5-551b-4626**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **b54ab3a6-1b55-2646**
 - Add the last 2 sections without transformation: **b54ab3a6-1b55-2646**-b8ee-2b36b3ee6672
 - Delete - : **b54ab3a61b552646**b8ee2b36b3ee6672
 - Divide bytes with backslashes: \b5\4a\b3\6\b1\55\26\46\b8\ee\2b\36\b3\ee\66\72
 - Filter example: (&(objectClass=*)(objectGUID = \b5\4a\b3\6\b1\55\26\46\b8\ee\2b\36\b3\ee\66\72))
 - Scope: Subtree
 - Attributes: objectGUID
 - Class [Type = UnicodeString]: class of the object that was undeleted. Some of the common Active Directory object classes:
 - container – for containers.
 - user – for users.
 - group – for groups.
 - domainDNS – for domain object.
 - groupPolicyContainer – for group policy objects.

For all possible values of this field open Active Directory Schema snap-in (see how to enable this snap-in: [https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx)) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

Operation:

- Correlation ID [Type = GUID]: multiple modifications are often executed as one operation via LDAP. This value allows you to correlate all the modification events that comprise the operation. Just look for other events from current subcategory with the same **Correlation ID**, for example “[5137](#): A directory service object was created.” and “[5139](#): A directory service object was moved.”

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Application Correlation ID** [Type = UnicodeString]: always has “-” value. Not in use.

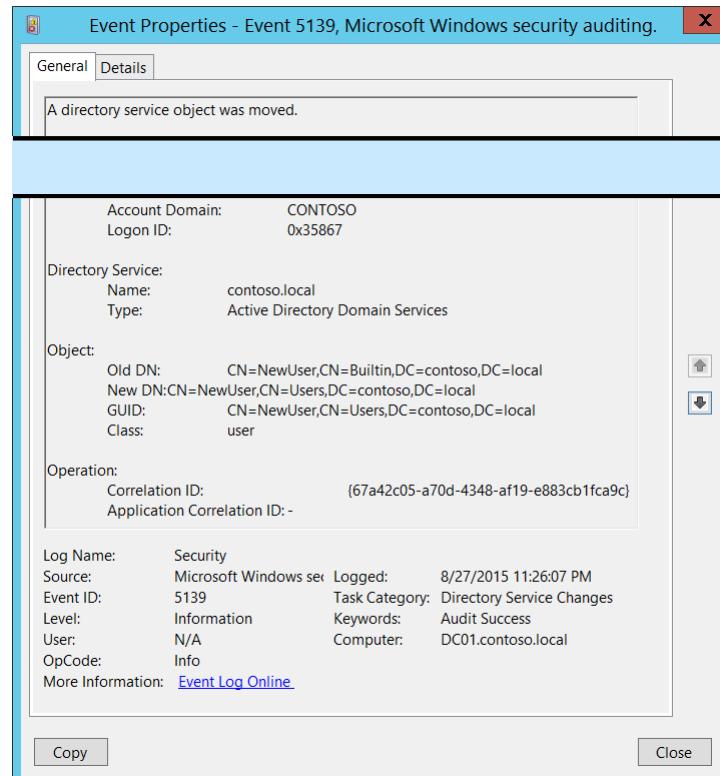
Security Monitoring Recommendations:

For 5138(S): A directory service object was undeleted.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor undelete operations (restoration) of Active Directory objects with specific classes, monitor for **Class** field with specific class name.
- It may be a good idea to monitor all undelete events, because the operation is not performed very often. Confirm that there is a reason for the object to be undeleted.

5139(S): A directory service object was moved.



Event Description:

This event generates every time an Active Directory object is moved.

This event only generates if the destination object has a particular entry in its **SACL**: the “**Create**” action, auditing for specific classes or objects. An example is the “**Create Computer objects**” action, auditing for the organizational unit.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>5139</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>14081</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-08-28T06:26:07.019116600Z" />
  <EventRecordID>409532</EventRecordID>
  <Correlation />
  <Execution ProcessID="520" ThreadID="600" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
```

```
- <EventData>
<Data Name="OpCorrelationID">{67A42C05-A70D-4348-AF19-E883CB1FCA9C}</Data>
<Data Name="AppCorrelationID">-</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x35867</Data>
<Data Name="DSName">contoso.local</Data>
<Data Name="DSType">%%14676</Data>
<Data Name="OldObjectDN">CN>NewUser,CN=Builtin,DC=contoso,DC=local</Data>
<Data Name="NewObjectDN">CN>NewUser,CN=Users,DC=contoso,DC=local</Data>
<Data Name="ObjectGUID">{06713960-9CC3-4B5D-A594-35883A04F934}</Data>
<Data Name="ObjectClass">user</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “move object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “move object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Directory Service:

- **Name** [Type = UnicodeString]: the name of an Active Directory domain, where the object was moved.

- **Type** [Type = UnicodeString]: has “**Active Directory Domain Services**” value for this event.

Object:

- **Old DN** [Type = UnicodeString]: Old distinguished name of moved object.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **New DN** [Type = UnicodeString]: New distinguished name of moved object. The Active Directory container to which the object was moved.

- **GUID** [Type = GUID]: each Active Directory object has globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise but also across the world.

GUIDs are assigned to every object created by Active Directory. Each object's GUID is stored in its Object-GUID (**objectGUID**) property.

Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object's GUID will yield results if the user has an account somewhere in the enterprise. In fact, searching for any object by Object-GUID might be the most reliable way of finding the object you want to find. The values of other object properties can change, but the Object-GUID never changes. When an object is assigned a GUID, it keeps that value for life.

Event Viewer automatically resolves **GUID** field to real object.

To translate this GUID, use the following procedure:

- Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(objectGUID=**GUID**)
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: a6b34ab5-551b-4626-b8ee-2b36b3ee6672
 - Take first 3 sections **a6b34ab5-551b-4626**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **b54ab3a6-1b55-2646**
 - Add the last 2 sections without transformation: **b54ab3a6-1b55-2646-b8ee-2b36b3ee6672**
 - Delete - : **b54ab3a61b552646b8ee2b36b3ee6672**
 - Divide bytes with backslashes: **\b5\4a\b3\6a\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72**
 - Filter example: (&(objectClass=*)(objectGUID = \b5\4a\b3\6a\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72))
 - Scope: Subtree
 - Attributes: objectGUID

- **Class** [Type = UnicodeString]: class of the object that was moved. Some of the common Active Directory object classes:

- container – for containers.
- user – for users.
- group – for groups.
- domainDNS – for domain object.

- groupPolicyContainer – for group policy objects.

For all possible values of this field open Active Directory Schema snap-in (see how to enable this snap-in: [https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx)) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

Operation:

- **Correlation ID** [Type = GUID]: multiple modifications are often executed as one operation via LDAP. This value allows you to correlate all the modification events that comprise the operation. Just look for other events from current subcategory with the same **Correlation ID**, for example “[5137](#): A directory service object was created.” and “[5141](#): A directory service object was deleted.”

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Application Correlation ID** [Type = UnicodeString]: always has “-“ value. Not in use.

Security Monitoring Recommendations:

For 5139(S): A directory service object was moved.

[Appendix A: Security monitoring recommendations for many audit events](#)



The screenshot shows the 'Event Properties' window for Event 5141. The 'General' tab is selected. The subject of the event is a directory service object named 'dadmin' in the 'CONTOSO\domain' account. The event occurred on '0x32004'. The object deleted was 'CN=WIN2003,CN=Users,DC=contoso,DC=local'. The type of the deleted object was 'Active Directory Domain Services'.

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor movement of Active Directory objects with specific classes, monitor for **Class** field with specific class name.
- You must set correct auditing access lists (SACLs) for specific classes within Active Directory container to get [5139](#). There is no reason to audit all movement events for all types of Active Directory objects, you need to find the most important locations (organizational units, folders, etc.) and monitor for movement of specific classes only to these locations (user, computer, group, etc.).

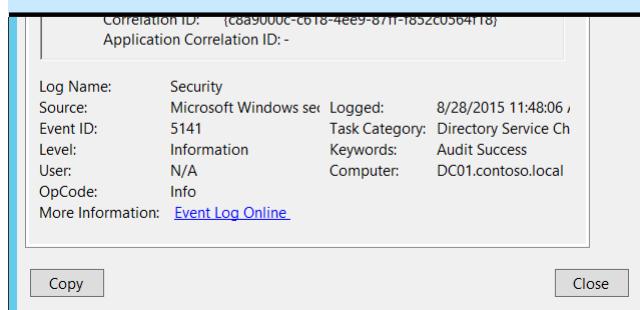
5141(S): A directory service object was deleted.

Event Description:

This event generates every time an Active Directory object is deleted.

This event only generates if the deleted object has a particular entry in its **SACL**: the “**Delete**” action, auditing for specific objects.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



The screenshot shows the 'Event Properties' window for Event 5141. The 'Details' tab is selected. The event was generated by 'Security' on '8/28/2015 11:48:06' with 'Event ID: 5141'. The 'Task Category' is 'Directory Service Ch' and 'Keywords' are 'Audit Success'. The 'Computer' is 'DC01.contoso.local'. The 'Log Name' is 'Security' and the 'Source' is 'Microsoft Windows security audit'. The 'Level' is 'Information' and 'User' is 'N/A'. The 'OpCode' is 'Info'. The 'More Information' link points to 'Event Log Online'.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5141</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14081</Task>
```

```
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-08-28T18:48:06.792762900Z" />
<EventRecordID>411118</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="4092" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="OpCorrelationID">{C8A9000C-C618-4EE9-87FF-F852C0564F18}</Data>
  <Data Name="AppCorrelationID">-</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x32004</Data>
  <Data Name="DSName">contoso.local</Data>
  <Data Name="DSType">%%14676</Data>
  <Data Name="ObjectDN">CN=WIN2003,CN=Users,DC=contoso,DC=local</Data>
  <Data Name="ObjectGUID">{CA15B875-AFB1-4E5A-86B2-96E61DE09110}</Data>
  <Data Name="ObjectClass">computer</Data>
  <Data Name="TreeDelete">%%14679</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete object” operation.

- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Directory Service:

- **Name** [Type = UnicodeString]: the name of an Active Directory domain, where the object was deleted.
- **Type** [Type = UnicodeString]: has "Active Directory Domain Services" value for this event.

Object:

- **DN** [Type = UnicodeString]: distinguished name of the object that was deleted.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas.

An RDN is an attribute with an associated value in the form attribute=value; . These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **GUID** [Type = GUID]: each Active Directory object has globally unique identifier (GUID), which is a 128-bit value that is unique not only in the enterprise but also across the world. GUIDs are assigned to every object created by Active Directory. Each object's GUID is stored in its Object-GUID (**objectGUID**) property.

Active Directory uses GUIDs internally to identify objects. For example, the GUID is one of an object's properties that is published in the global catalog. Searching the global catalog for a User object's GUID will yield results if the user has an account somewhere in the enterprise. In fact, searching for any object by Object-GUID might be the most reliable way of finding the object you want to find. The values of other object properties can change, but the Object-GUID never changes. When an object is assigned a GUID, it keeps that value for life.

Event Viewer automatically resolves **GUID** field to real object. For deleted objects **GUID** will be resolved to new destination of object, for example: OU=My\0ADEL:cc94c0d7-dd53-4061-9791-e53478dbbc3b,CN=Deleted Objects,DC=contoso,DC=local.

To translate this GUID, use the following procedure:

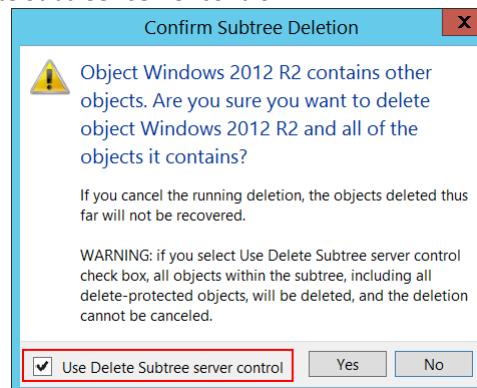
- Perform the following LDAP search using LDP.exe tool:
 - Base DN: CN=Schema,CN=Configuration,DC=XXX,DC=XXX
 - Filter: (&(objectClass=*)(objectGUID=**GUID**))
 - Perform the following operations with the GUID before using it in a search request:
 - We have this GUID to search for: a6b34ab5-551b-4626-b8ee-2b36b3ee6672
 - Take first 3 sections a6**b34ab5-551b-4626**.
 - For each of these 3 sections you need to change (Invert) the order of bytes, like this **b54ab3a6-1b55-2646**
 - Add the last 2 sections without transformation: **b54ab3a6-1b55-2646**-b8ee-2b36b3ee6672

- Delete - : **b54ab3a61b552646b8ee2b36b3ee6672**
 - Divide bytes with backslashes: \b5\4a\b3\6\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72
- Filter example: (&(objectClass=*)(objectGUID = \b5\4a\b3\6\1b\55\26\46\b8\ee\2b\36\b3\ee\66\72))
 - Scope: Subtree
 - Attributes: objectGUID
- **Class** [Type = UnicodeString]: class of the object that was deleted. Some of the common Active Directory object classes:
 - container – for containers.
 - user – for users.
 - group – for groups.
 - domainDNS – for domain object.
 - groupPolicyContainer – for group policy objects.

For all possible values of this field open Active Directory Schema snap-in (see how to enable this snap-in: [https://technet.microsoft.com/en-us/library/Cc755885\(v=WS.10\).aspx](https://technet.microsoft.com/en-us/library/Cc755885(v=WS.10).aspx)) and navigate to **Active Directory Schema\Classes**. Or use this document: <https://msdn.microsoft.com/en-us/library/cc221630.aspx>

Operation:

- **Tree Delete** [Type = UnicodeString]:
 - **Yes** – “Delete Subtree” operation was performed. It happens, for example, if “Use Delete Subtree server control” check box was checked during delete operation using Active Directory Users and Computers management console.
 - **No** – delete operation was performed without “Delete Subtree” server control.



- **Correlation ID** [Type = GUID]: multiple modifications are often executed as one operation via LDAP. This value allows you to correlate all the modification events that comprise the operation. Just look for other events from current subcategory with the same **Correlation ID**, for example “[5137](#): A directory service object was created.” and “[5139](#): A directory service object was moved.”

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Application Correlation ID** [Type = UnicodeString]: always has “–” value. Not in use.

Security Monitoring Recommendations:

For 5141(S): A directory service object was deleted.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor deletion of Active Directory objects with specific classes, monitor for **Class** field with specific class name. For example, we recommend that you monitor for group policy objects deletions: **groupPolicyContainer** class.
- If you need to monitor deletion of specific Active Directory objects, monitor for **DN** field with specific object name. For example, if you have critical Active Directory objects which should not be deleted, monitor for their deletion.

Audit Directory Service Replication

Audit Directory Service Replication determines whether the operating system generates audit events when replication between two domain controllers begins and ends.

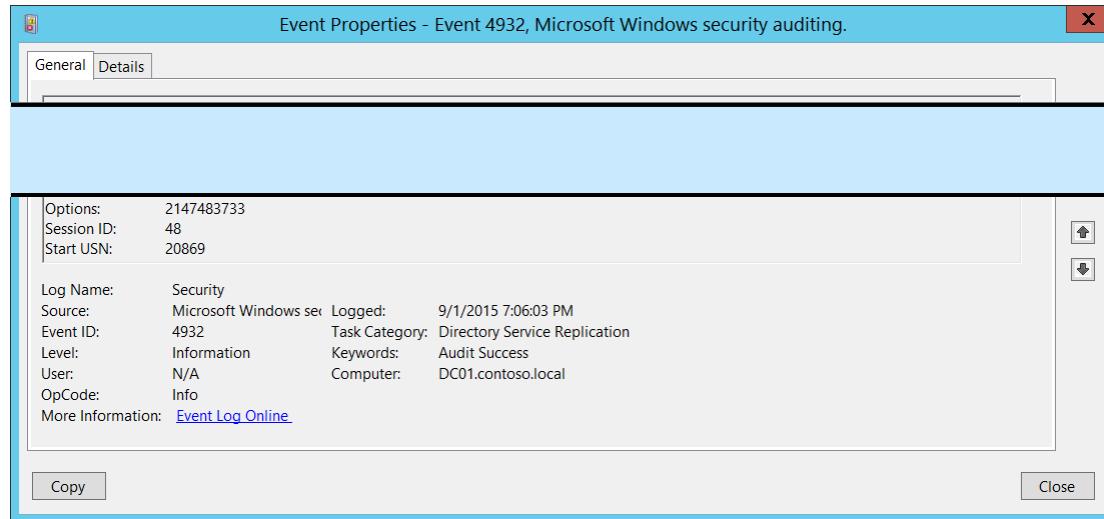
Event volume: Medium on domain controllers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	IF	IF	IF - Events in this subcategory typically have an informational purpose and it is difficult to detect any malicious activity using these events. It's mainly used for Active Directory replication troubleshooting.
Member Server	No	No	No	No	This subcategory makes sense only on domain controllers.
Workstation	No	No	No	No	This subcategory makes sense only on domain controllers.

Events List:

- [4932\(S\)](#): Synchronization of a replica of an Active Directory naming context has begun.
- [4933\(S, F\)](#): Synchronization of a replica of an Active Directory naming context has ended.

4932(S): Synchronization of a replica of an Active Directory naming context has begun.



Event Properties - Event 4932, Microsoft Windows security auditing.

General Details

Options: 2147483733
Session ID: 48
Start USN: 20869

Log Name: Security
Source: Microsoft Windows security auditing
Event ID: 4932
Level: Information
User: N/A
OpCode: Info

Logged: 9/1/2015 7:06:03 PM
Task Category: Directory Service Replication
Keywords: Audit Success
Computer: DC01.contoso.local

More Information: [Event Log Online](#)

Copy **Close**

Event Description:
This event generates every time synchronization of a replica of an Active Directory naming context has begun.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4932</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>14082</Task>
```

```
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-02T02:06:03.814642100Z" />
<EventRecordID>413689</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="276" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="DestinationDRA">CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="SourceDRA">CN=NTDS Settings,CN=WIN2012R2,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="NamingContext">CN=Schema,CN=Configuration,DC=contoso,DC=local</Data>
  <Data Name="Options">2147483733</Data>
  <Data Name="SessionID">48</Data>
  <Data Name="StartUSN">20869</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name.

The LDAP API references an LDAP object by its **distinguished name** (DN). A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Naming Context** [Type = UnicodeString]: naming context to replicate.

The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.

- **Options** [Type = UInt32]: decimal value of [DRS Options](#).
- **Session ID** [Type = UInt32]: unique identifier of replication session. Using this field you can find "[4932](#): Synchronization of a replica of an Active Directory naming context has begun." and "[4933](#): Synchronization of a replica of an Active Directory naming context has ended." events for the same session.

- Start USN [Type = UnicodeString]: Naming Context's USN number before replication begins.

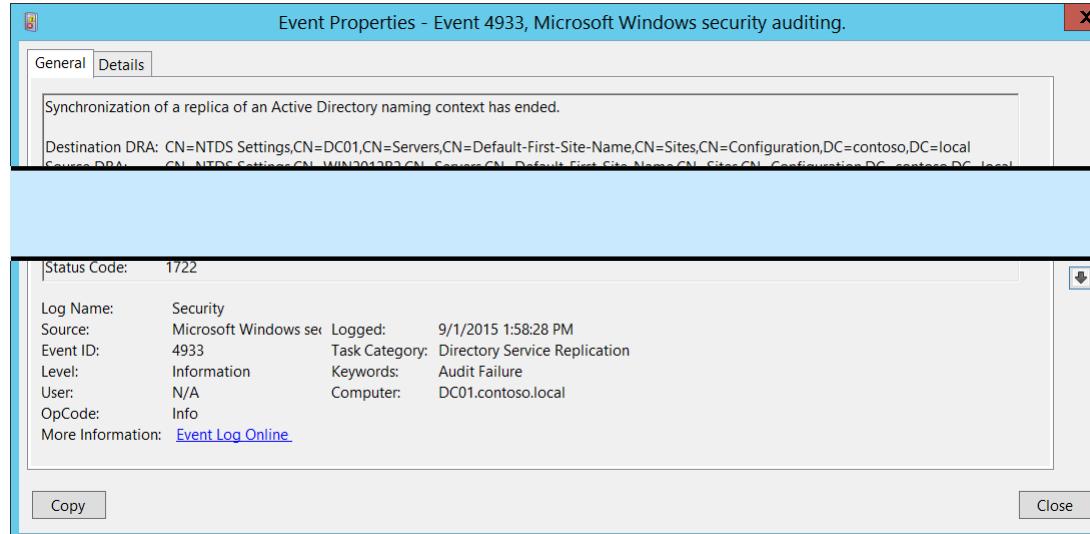
Active Directory replication does not depend on time to determine what changes need to be propagated. It relies instead on the use of **update sequence numbers (USNs)** that are assigned by a counter that is local to each domain controller. Because these USN counters are local, it is easy to ensure that they are reliable and never "run backward" (that is, decrease in value). The trade-off is that it is meaningless to compare a USN assigned on one domain controller to a USN assigned on a different domain controller. The replication system is designed with this restriction in mind.

Security Monitoring Recommendations:

For 4932(S): Synchronization of a replica of an Active Directory naming context has begun.

- Monitor for **Source Address** field, because the source of replication (DRA) must be authorized for this action. If you find any unauthorized DRA you should trigger an event.
- This event is typically used for Active Directory replication troubleshooting.

4933(S, F): Synchronization of a replica of an Active Directory naming context has ended.

 Event Properties - Event 4933, Microsoft Windows security auditing.

General Details

Synchronization of a replica of an Active Directory naming context has ended.

Destination DRA: CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local
Source DRA: CN=NTDS Settings,CN=WIND2013P,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local

Status Code: 1722

Log Name: Security
Source: Microsoft Windows security
Event ID: 4933
Level: Information
User: N/A
OpCode: Info

Logged: 9/1/2015 1:58:28 PM
Task Category: Directory Service Replication
Keywords: Audit Failure
Computer: DC01.contoso.local

More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates every time synchronization of a replica of an Active Directory naming context has ended.

Failure event occurs when synchronization of a replica of an Active Directory naming context failed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4933</EventID>
<Version>0</Version>
<Level>0</Level>
```

```
<Task>14082</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-01T20:58:28.854735700Z" />
<EventRecordID>413644</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="2288" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="DestinationDRA">CN=NTDS Settings,CN=DC01,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="SourceDRA">CN=NTDS Settings,CN=WIN2012R2,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="NamingContext">CN=Schema,CN=Configuration,DC=contoso,DC=local</Data>
<Data Name="Options">2147483733</Data>
<Data Name="SessionID">40</Data>
<Data Name="EndUSN">20869</Data>
<Data Name="StatusCode">1722</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

- **Destination DRA** [Type = UnicodeString]: destination directory replication agent distinguished name.

The **Directory Replication Agent (DRA)** handles replication between domain controllers. The Directory Replication Agent uses the connection objects in the topology map to find out those partners that are relevant when replicating changes to directory partitions. The DRA sends a replication request to the partners of a domain controller when the domain controller needs to update its copy of Active Directory.

- **Source DRA** [Type = UnicodeString]: source directory replication agent distinguished name.

The LDAP API references an LDAP object by its **distinguished name (DN)**. A DN is a sequence of relative distinguished names (RDN) connected by commas. An RDN is an attribute with an associated value in the form attribute=value;. These are examples of RDNs attributes:

- DC - domainComponent
- CN - commonName
- OU - organizationalUnitName
- O - organizationName

- **Naming Context** [Type = UnicodeString]: naming context to replicate.

The Directory Tree of Active Directory tree is partitioned to allow sections to be distributed (replicated) to domain controllers in different domains within the forest. Each domain controller stores a copy of a specific part of the directory tree, called a **Naming Context** also known as Directory Partition. **Naming Context** is replicated as a unit to other domain controllers in the forest that contain a replica of the same sub tree. A **Naming Context** is also called a Directory Partition.

- **Options** [Type = UInt32]: decimal value of [DRS Options](#).
- **Session ID** [Type = UInt32]: unique identifier of replication session. Using this field you can find "[4932](#): Synchronization of a replica of an Active Directory naming context has begun." and "[4933](#): Synchronization of a replica of an Active Directory naming context has ended." events for the same session.
- **End USN** [Type = UInt32]: **Naming Context's** USN number after replication ends.

Active Directory replication does not depend on time to determine what changes need to be propagated. It relies instead on the use of **update sequence numbers (USNs)** that are assigned by a counter that is local to each domain controller. Because these USN counters are local, it is easy to ensure that they are reliable and never "run backward" (that is, decrease in value). The trade-off is that it is meaningless to compare a USN assigned on one domain controller to a USN assigned on a different domain controller. The replication system is designed with this restriction in mind.

- **Status Code** [Type = UInt32]: if there are no issues or errors, the status code will be "**0**". If an error happened, you will receive Failure event and Status Code will not be equal to "**0**". You can check error code meaning here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx)

Security Monitoring Recommendations:

For 4933(S, F): Synchronization of a replica of an Active Directory naming context has ended.

- Monitor for **Source Address** field, because the source of replication (DRA) must be authorized for this action. If you find any unauthorized DRA you should trigger an event.
- This event is typically used for Active Directory replication troubleshooting.

Logon and Logoff

Audit Account Lockout

Audit Account Lockout enables you to audit security events that are generated by a failed attempt to log on to an account that is locked out.

If you configure this policy setting, an audit event is generated when an account cannot log on to a computer because the account is locked out. Success audits record successful attempts and failure audits record unsuccessful attempts.

Account lockout events are essential for understanding user activity and detecting potential attacks.

Event volume: Low.

This subcategory failure logon attempts, when account was already locked out.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	Yes	No	Yes	We recommend tracking account lockouts, especially for high value domain or local accounts (database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts, and so on). This subcategory doesn't have Success events, so there is no recommendation to enable Success auditing for this subcategory.
Member Server	No	Yes	No	Yes	We recommend tracking account lockouts, especially for high value domain or local accounts (database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts, and so on). This subcategory doesn't have Success events, so there is no recommendation to enable Success auditing for this subcategory.
Workstation	No	Yes	No	Yes	We recommend tracking account lockouts, especially for high value domain or local accounts (database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts, and so on). This subcategory doesn't have Success events, so there is no recommendation to enable Success auditing for this subcategory.

Events List:

- [4625\(F\)](#): An account failed to log on.

4625(F): An account failed to log on.

 Event Properties - Event 4625, Microsoft Windows security auditi... X

General Details

An account failed to log on.

Subject:
Security ID: SYSTEM
Account Name: DC01\$

Logon Type: 2

Account For Which Logon Failed:
Security ID: NULL SID
Account Name: Auditor
Account Domain: CONTOSO

Failure Information:
Failure Reason: Account locked out.
Status: 0xC0000234
Sub Status: 0x0

Process Information:
Caller Process ID: 0x1bc
Caller Process Name: C:\Windows\System32\winlogon.exe

Network Information:
Workstation Name: DC01
Source Network Address: 127.0.0.1
Source Port: 0

Detailed Authentication Information:
Logon Process: User32
Authentication Package: Negotiate
Transited Services: -
Package Name (NTLM only): -
Key Length: 0

This event is generated when a logon request fails. It is generated on the computer where access was attempted.

Log Name:	Security		
Source:	Microsoft Windows se	Logged:	9/8/2015 3:54:54 PM
Event ID:	4625	Task Category:	Account Lockout
Level:	Information	Keywords:	Audit Failure
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info		
More Information: Event Log Online			

Copy Close

Event Description:

This event generates if an account logon attempt failed when the account was already locked out. It also generates for a logon attempt after which the account was locked out.

It generates on the computer where logon attempt was made, for example, if logon attempt was made on user's workstation, then event will be logged on this workstation.

This event generates on domain controllers, member servers, and workstations.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4625</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12546</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-08T22:54:54.962511700Z" />
<EventRecordID>229977</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="3240" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TargetUserSid">S-1-0-0</Data>
<Data Name="TargetUserName">Auditor</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
```

<Data Name="Status">0xc0000234</Data>
<Data Name="FailureReason">%%2307</Data>

```

<Data Name="SubStatus">0x0</Data>
<Data Name="LogonType">2</Data>
<Data Name="LogonProcessName">User32</Data>
<Data Name="AuthenticationPackageName">Negotiate</Data>
<Data Name="WorkstationName">DC01</Data>
<Data Name="TransmittedServices">-</Data>
<Data Name="LmPackageName">-</Data>
<Data Name="KeyLength">0</Data>
<Data Name="ProcessId">0x1bc</Data>
<Data Name="ProcessName">C:\Windows\System32\winlogon.exe</Data>
<Data Name="IpAddress">127.0.0.1</Data>
<Data Name="IpPort">0</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that reported information about logon failure. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that reported information about logon failure.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

Logon Type [Type = UInt32]: the type of logon which was performed. "Table 11. Windows Logon Types" contains the list of possible values for this field.

Logon Type	Logon Title	Description
2	Interactive	A user logged on to this computer.
3	Network	A user or computer logged on to this computer from the network.
4	Batch	Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct

		intervention.
5	Service	A service was started by the Service Control Manager.
7	Unlock	This workstation was unlocked.
8	NetworkCleartext	A user logged on to this computer from the network. The user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext).
9	NewCredentials	A caller cloned its current token and specified new credentials for outbound connections. The new logon session has the same local identity, but uses different credentials for other network connections.
10	RemoteInteractive	A user logged on to this computer remotely using Terminal Services or Remote Desktop.
11	CachedInteractive	A user logged on to this computer with network credentials that were stored locally on the computer. The domain controller was not contacted to verify the credentials.

Table 11. Windows Logon Types

Account For Which Logon Failed:

- **Security ID** [Type = SID]: SID of the account that was specified in the logon attempt. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that was specified in the logon attempt.
- **Account Domain** [Type = UnicodeString]: domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Failure Information:

- **Failure Reason** [Type = UnicodeString]: textual explanation of **Status** field value. For this event it typically has "Account locked out" value.
- **Status** [Type = HexInt32]: the reason why logon failed. For this event it typically has "0xC0000234" value. The most common status codes are listed in "Table 12. Windows logon status codes."

Status\Sub-Status Code	Description
0XC000005E	There are currently no logon servers available to service the logon request.
0xC0000064	User logon with misspelled or bad user account

0xC000006A	User logon with misspelled or bad password
0XC000006D	This is either due to a bad username or authentication information
0XC000006E	Unknown user name or bad password.
0xC000006F	User logon outside authorized hours
0xC0000070	User logon from unauthorized workstation
0xC0000071	User logon with expired password
0xC0000072	User logon to account disabled by administrator
0XC00000DC	Indicates the Sam Server was in the wrong state to perform the desired operation.
0XC0000133	Clocks between DC and other computer too far out of sync
0XC000015B	The user has not been granted the requested logon type (aka logon right) at this machine
0XC000018C	The logon request failed because the trust relationship between the primary domain and the trusted domain failed.
0XC0000192	An attempt was made to logon, but the Netlogon service was not started.
0xC0000193	User logon with expired account
0XC0000224	User is required to change password at next logon
0XC0000225	Evidently a bug in Windows and not a risk
0xC0000234	User logon with account locked
0XC00002EE	Failure Reason: An Error occurred during Logon
0XC0000413	Logon Failure: The machine you are logging onto is protected by an authentication firewall. The specified account is not allowed to authenticate to the machine.
0x0	Status OK.

Table 12. Windows logon status codes.

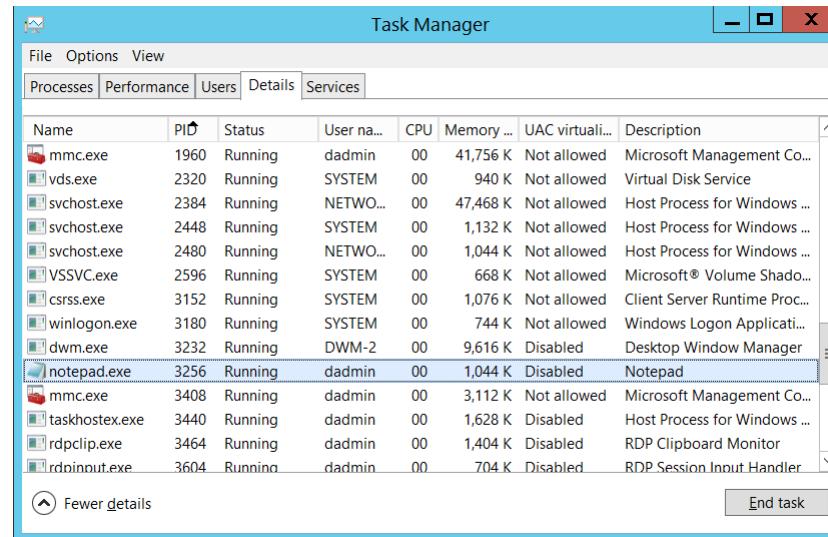
To see the meaning of other status\sub-status codes you may also check for status code in the Window header file **ntstatus.h** in Windows SDK.

More information: <https://dev.windows.com/en-us/downloads>

- **Sub Status** [Type = HexInt32]: additional information about logon failure. The most common sub-status codes listed in the “Table 12. Windows logon status codes.”.

Process Information:

- **Caller Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted the logon. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Caller Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Network Information:

- **Workstation Name** [Type = UnicodeString]: machine name from which logon attempt was performed.
- **Source Network Address** [Type = UnicodeString]: IP address of machine from which logon attempt was performed.
 - IPv6 address or ::ffff:IPv4 address of a client.
 - ::1 or 127.0.0.1 means localhost.
- **Source Port** [Type = UnicodeString]: source port which was used for logon attempt from remote machine.
 - 0 for interactive logons.

Detailed Authentication Information:

- **Logon Process** [Type = UnicodeString]: the name of the trusted logon process that was used for the logon attempt. See event “[4611: A trusted logon process has been registered with the Local Security Authority](#)” description for more information.
- **Authentication Package** [Type = UnicodeString]: The name of the authentication package which was used for the logon authentication process. Default packages loaded on LSA startup are located in “HKLM\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig” registry key. Other packages can be loaded at runtime. When a new package is loaded a “[4610: An authentication package has been loaded by the Local Security Authority](#)” (typically for NTLM) or “[4622: A security package has been loaded by the Local Security Authority](#)” (typically for Kerberos) event is logged to indicate that a new package has been loaded along with the package name. The most common authentication packages are:
 - **NTLM** – NTLM-family Authentication
 - **Kerberos** – Kerberos authentication.
 - **Negotiate** – the Negotiate security package selects between Kerberos and NTLM protocols. Negotiate selects Kerberos unless it cannot be used by one of the systems involved in the authentication or the calling application did not provide sufficient information to use Kerberos.

- **Transited Services** [Type = UnicodeString] [Kerberos-only]: the list of transmitted services. Transmitted services are populated if the logon was a result of a S4U (Service For User) logon process. S4U is a Microsoft extension to the Kerberos Protocol to allow an application service to obtain a Kerberos service ticket on behalf of a user – most commonly done by a front-end website to access an internal resource on behalf of a user. For more information about S4U, see <https://msdn.microsoft.com/en-us/library/cc246072.aspx>
- **Package Name (NTLM only)** [Type = UnicodeString]: The name of the LAN Manager sub-package ([NTLM-family](#) protocol name) that was used during the logon attempt. Possible values are:

- “NTLM V1”
- “NTLM V2”
- “LM”

Only populated if “**Authentication Package**” = “**NTLM**”.

- **Key Length** [Type = UInt32]: the length of [NTLM Session Security](#) key. Typically it has 128 bit or 56 bit length. This parameter is always 0 if “**Authentication Package**” = “**Kerberos**”, because it is not applicable for Kerberos protocol. This field will also have “0” value if Kerberos was negotiated using **Negotiate** authentication package.

Security Monitoring Recommendations:

For 4625(F): An account failed to log on.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If **Subject\Account Name** is a name of service account or user account, it may be useful to investigate whether that account is allowed (or expected) to request logon for **Account For Which Logon Failed\Security ID**.
- To monitor for a mismatch between the logon type and the account that uses it (for example, if **Logon Type** 4-Batch or 5-Service is used by a member of a domain administrative group), monitor **Logon Type** in this event.
- If you have a high-value domain or local account for which you need to monitor every lockout, monitor all [4625](#) events with the “**Subject\Security ID**” that corresponds to the account.
- We recommend monitoring all [4625](#) events for local accounts, because these accounts typically should not be locked out. This is especially relevant for critical servers, administrative workstations, and other high value assets.
- We recommend monitoring all [4625](#) events for service accounts, because these accounts should not be locked out or prevented from functioning. This is especially relevant for critical servers, administrative workstations, and other high value assets.
- If your organization restricts logons in the following ways, you can use this event to monitor accordingly:
 - If the “**Account For Which Logon Failed \Security ID**” should never be used to log on from the specific **Network Information\Workstation Name**.
 - If a specific account, such as a service account, should only be used from your internal IP address list (or some other list of IP addresses). In this case, you can monitor for **Network Information\Source Network Address** and compare the network address with your list of IP addresses.
 - If a particular version of NTLM is always used in your organization. In this case, you can use this event to monitor **Package Name (NTLM only)**, for example, to find events where **Package Name (NTLM only)** does not equal **NTLM V2**.
 - If NTLM is not used in your organization, or should not be used by a specific account (**New Logon\Security ID**). In this case, monitor for all events where **Authentication Package** is NTLM.

- If the **Authentication Package** is NTLM. In this case, monitor for **Key Length** not equal to 128, because all Windows operating systems starting with Windows 2000 support 128-bit Key Length.
- If **Logon Process** is not from a trusted logon processes list.
- Monitor for all events with the fields and values in the following table:

Field	Value to monitor for
Failure Information\Status or Failure Information\Sub Status	0XC000005E – “There are currently no logon servers available to service the logon request.” This is typically not a security issue but it can be an infrastructure or availability issue.
Failure Information\Status or Failure Information\Sub Status	0xC0000064 – “User logon with misspelled or bad user account”. Especially if you get a number of these in a row, it can be a sign of user enumeration attack.
Failure Information\Status or Failure Information\Sub Status	0xC000006A – “User logon with misspelled or bad password” for critical accounts or service accounts. Especially watch for a number of such events in a row.
Failure Information\Status or Failure Information\Sub Status	0XC000006D – “This is either due to a bad username or authentication information” for critical accounts or service accounts. Especially watch for a number of such events in a row.
Failure Information\Status or Failure Information\Sub Status	0xC000006F – “User logon outside authorized hours”.
Failure Information\Status or Failure Information\Sub Status	0xC0000070 – “User logon from unauthorized workstation”.
Failure Information\Status or Failure Information\Sub Status	0xC0000072 – “User logon to account disabled by administrator”.
Failure Information\Status or Failure Information\Sub Status	0XC000015B – “The user has not been granted the requested logon type (aka logon right) at this machine”.
Failure Information\Status or Failure Information\Sub Status	0XC0000192 – “An attempt was made to logon, but the Netlogon service was not started”. This is typically not a security issue but it can be an infrastructure or availability issue.
Failure Information\Status or Failure Information\Sub Status	0xC0000193 – “User logon with expired account”.
Failure Information\Status or Failure Information\Sub Status	0XC0000413 – “Logon Failure: The machine you are logging onto is protected by an authentication firewall. The specified account is not allowed to authenticate to the machine”.

Audit User/Device Claims

Audit User/Device Claims allows you to audit user and device claims information in the account's logon token. Events in this subcategory are generated on the computer on which a logon session is created. For an interactive logon, the security audit event is generated on the computer that the user logged on to.

For a network logon, such as accessing a shared folder on the network, the security audit event is generated on the computer hosting the resource.

Important: [Audit Logon](#) subcategory must also be enabled in order to get events from this subcategory.

Event volume:

- Low on a client computer.
- Medium on a domain controller or network servers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	No	IF	No	IF – if claims are in use in your organization and you need to monitor user/device claims, enable Success auditing for this subcategory. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	IF	No	IF	No	IF – if claims are in use in your organization and you need to monitor user/device claims, enable Success auditing for this subcategory. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	IF	No	IF	No	IF – if claims are in use in your organization and you need to monitor user/device claims, enable Success auditing for this subcategory. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Events List:

- [4626\(S\)](#): User/Device claims information.

4626(S): User/Device claims information.

Event Properties - Event 4626, Microsoft Windows security auditing.

General **Details**

User / Device claims information.

Subject:

- Security ID: NULL SID
- Account Name: -
- Account Domain: -
- Logon ID: 0x0

Logon Type: 3

New Logon:

- Security ID: CONTOSO\dadmin
- Account Name: dadmin
- Account Domain: CONTOSO
- Logon ID: 0x136F7B

Event in sequence: 1 of 1

User Claims:

```
ad://ext/cn:88d2b96fdb2b4c49 <String> : "dadmin"
ad://ext/Department:88d16a8edaa8c66b <String> : "IT"
```

Device Claims: -

The subject fields indicate the account on the local system which requested the logon. This is most commonly a service such as the Server service, or a local process such as Winlogon.exe or Services.exe.

The logon type field indicates the kind of logon that occurred. The most common types are 2 (interactive) and 3 (network).

The New Logon fields indicate the account for whom the new logon was created, i.e. the account that was logged on.

This event is generated when the Audit User/Device claims subcategory is configured and the user's logon token contains user/device claims information. The Logon ID field can be used to correlate this event with the corresponding user logon event as well as to any other security audit events generated during this logon session.

Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4626
Level:	Information
User:	N/A
OpCode:	Info

Logged: 9/9/2015 5:12:02 PM Task Category: User / Device Claims Keywords: Audit Success Computer: DC01.contoso.local

More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates for new account logons and contains user/device claims which were associated with a new logon session.

This event does not generate if the user/device doesn't have claims.

For computer account logons you will also see device claims listed in the “User Claims” field.

You will typically get “[4624](#): An account was successfully logged on” and after it a 4626 event with the same information in **Subject**, **Logon Type** and **New Logon** sections.

This event generates on the computer to which the logon was performed (target computer). For example, for Interactive logons it will be the same computer.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4626</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12553</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-10T00:12:02.243396300Z" />
<EventRecordID>232648</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="1092" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserName"></Data>
<Data Name="SubjectDomainName"></Data>
<Data Name="SubjectLogonId">0x0</Data>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
```

```
<Data Name="SubjectDomainName"></Data>
<Data Name="SubjectLogonId">0x0</Data>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
```

```

<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x136f7b</Data>
<Data Name="LogonType">3</Data>
<Data Name="EventIdx">1</Data>
<Data Name="EventCountTotal">1</Data>
<Data Name="UserClaims">ad://ext/cn:88d2b96fdb2b4c49 <%>1818 : "dadmin" ad://ext/Department:88d16a8edaa8c66b <%>1818 : "IT" </Data>
<Data Name="DeviceClaims">-</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that reported information about claims. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that reported information about claims.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Logon Type [Type = UInt32]: the type of logon which was performed. The table below contains the list of possible values for this field:

Logon Type	Logon Title	Description
2	Interactive	A user logged on to this computer.
3	Network	A user or computer logged on to this computer from the network.
4	Batch	Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct intervention.
5	Service	A service was started by the Service Control Manager.

7	Unlock	This workstation was unlocked.
8	NetworkCleartext	A user logged on to this computer from the network. The user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext).
9	NewCredentials	A caller cloned its current token and specified new credentials for outbound connections. The new logon session has the same local identity, but uses different credentials for other network connections.
10	RemoteInteractive	A user logged on to this computer remotely using Terminal Services or Remote Desktop.
11	CachedInteractive	A user logged on to this computer with network credentials that were stored locally on the computer. The domain controller was not contacted to verify the credentials.

New Logon:

- **Security ID** [Type = SID]: SID of account for which logon was performed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account for which logon was performed.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Event in sequence [Type = UInt32]: If there is not enough space in one event to put all claims, you will see "**1 of N**" in this field and additional events will be generated. Typically this field has "**1 of 1**" value.

User Claims [Type = UnicodeString]: list of user claims for new logon session. This field contains user claims if user account was logged in and device claims if computer account was logged in. Here is an example how to parse the entrance of this field:

- ad://ext/**cn**:88d2b96fdb2b4c49 **<String>** : "dadmin"
 - **cn** – claim display name.
 - 88d2b96fdb2b4c49 – unique claim ID.
 - **<String>** - claim type.
 - "dadmin" – claim value.

Device Claims [Type = UnicodeString]: list of device claims for new logon session. For user accounts this field typically has "-" value. For computer accounts this field has device claims listed.

Security Monitoring Recommendations:

For 4626(S): User/Device claims information.

- Typically this action is reported by the NULL SID account, so we recommend reporting all events with “**Subject\Security ID**” not equal “**NULL SID**”.
- If you need to monitor account logons with specific claims, you can monitor for [4626](#) and check **User Claims\Device Claims** fields.
- If you have specific requirements, such as:
 - Users with specific claims should not access specific computers;
 - Computer account should not have specific claims;
 - User account should not have specific claims;
 - Claim should not be empty
 - And so on...

You can monitor for [4626](#) and check **User Claims\Device Claims** fields.

- If you need to monitor computer/user logon attempts only and you don’t need information about claims, then it is better to monitor “[4624](#): An account was successfully logged on.”

Audit Group Membership

Audit Group Membership enables you to audit group memberships when they are enumerated on the client computer.

This policy allows you to audit the group membership information in the user's logon token. Events in this subcategory are generated on the computer on which a logon session is created. For an interactive logon, the security audit event is generated on the computer that the user logged on to. For a network logon, such as accessing a shared folder on the network, the security audit event is generated on the computer hosting the resource.

You must also enable the [Audit Logon](#) subcategory.

Multiple events are generated if the group membership information cannot fit in a single security audit event

Event volume:

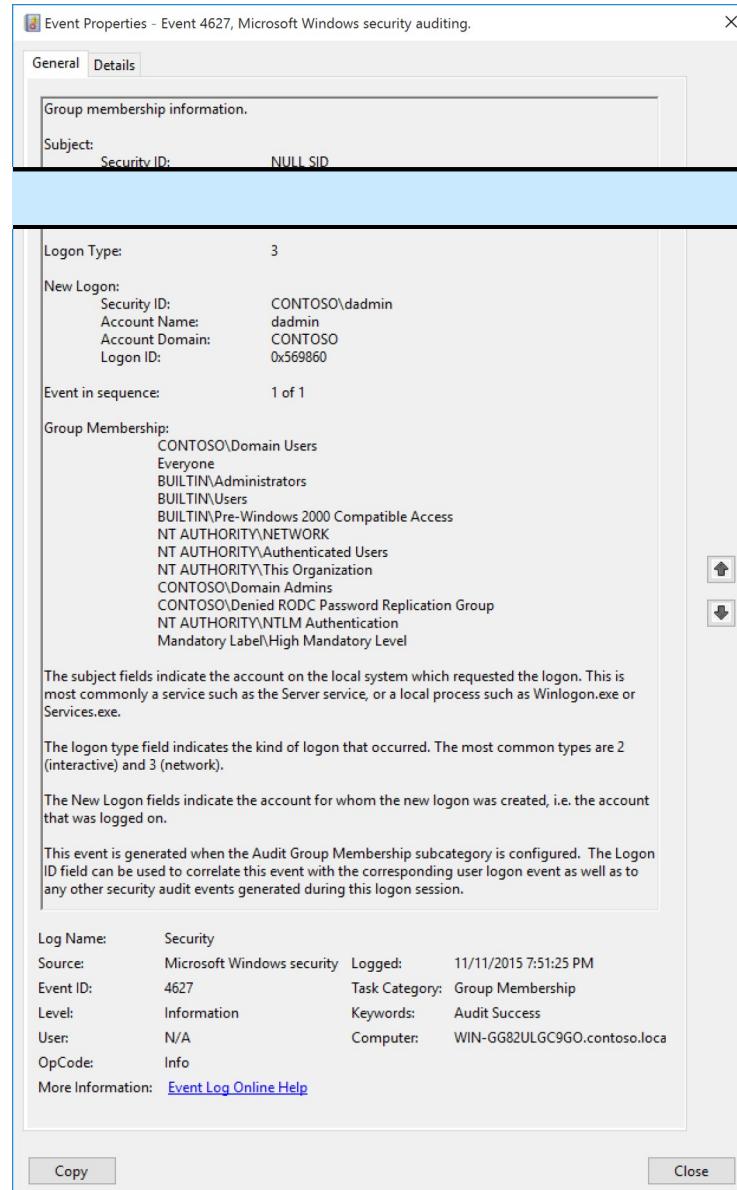
- Low on a client computer.
- Medium on a domain controller or network servers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>Group membership information for logged in user can help to detect that member of specific domain or local group logged in to the machine (for example, member of database administrators, built-in local administrators, domain administrators, service accounts group or other high value groups).</p> <p>For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	Yes	No	Yes	No	<p>Group membership information for logged in user can help to detect that member of specific domain or local group logged in to the machine (for example, member of database administrators, built-in local administrators, domain administrators, service accounts group or other high value groups).</p> <p>For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	Yes	No	Yes	No	<p>Group membership information for logged in user can help to detect that member of specific domain or local group logged in to the machine (for example, member of database administrators, built-in local administrators, domain administrators, service accounts group or other high value groups).</p> <p>For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4627\(S\)](#): Group membership information.

4627(S): Group membership information.

 Event Properties - Event 4627, Microsoft Windows security auditing.

General Details

Group membership information.

Subject: Security ID: NULL SID

Logon Type: 3

New Logon:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x569860

Event in sequence: 1 of 1

Group Membership:

- CONTOSO\Domain Users
- Everyone
- BUILTIN\Administrators
- BUILTIN\Users
- BUILTIN\Pre-Windows 2000 Compatible Access
- NT AUTHORITY\NETWORK
- NT AUTHORITY\Authenticated Users
- NT AUTHORITY\This Organization
- CONTOSO\Domain Admins
- CONTOSO\Denied RODC Password Replication Group
- NT AUTHORITY\NTLM Authentication
- Mandatory Label\High Mandatory Level

The subject fields indicate the account on the local system which requested the logon. This is most commonly a service such as the Server service, or a local process such as Winlogon.exe or Services.exe.

The logon type field indicates the kind of logon that occurred. The most common types are 2 (interactive) and 3 (network).

The New Logon fields indicate the account for whom the new logon was created, i.e. the account that was logged on.

This event is generated when the Audit Group Membership subcategory is configured. The Logon ID field can be used to correlate this event with the corresponding user logon event as well as to any other security audit events generated during this logon session.

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 4627
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online Help](#)

Logged: 11/11/2015 7:51:25 PM
 Task Category: Group Membership
 Keywords: Audit Success
 Computer: WIN-GG82ULGC9GO.contoso.local

Copy **Close**

Event Description:

This event generates with “[4624\(S\): An account was successfully logged on](#)” and shows the list of groups that the logged-on account belongs to.

You must also enable the Success audit for [Audit Logon](#) subcategory to get this event.

Multiple events are generated if the group membership information cannot fit in a single security audit event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4627</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12554</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T03:51:25.843673000Z" />
<EventRecordID>3081</EventRecordID>
<Correlation ActivityID="{913FBF70-1CE6-0000-67BF-3F91E61CD101}" />
<Execution ProcessID="736" ThreadID="808" />
<Channel>Security</Channel>
<Computer>WIN-GG82ULGC9GO.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-0-0</Data>
<Data Name="SubjectUserName">-</Data>
<Data Name="SubjectDomainName">-</Data>
<Data Name="SubjectLogonId">0x0</Data>
<Data Name="TargetUserId">S-1-5-21-1377283216-344919071-3415362939-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x569860</Data>
<Data Name="LogonType">3</Data>
<Data Name="EventIdx">1</Data>
```

```
<Data Name="EventCountTotal">1</Data>
<Data Name="GroupMembership">%{S-1-5-21-1377283216-344919071-3415362939-513} %{S-1-1-0} %{S-1-5-32-544} %{S-1-5-32-545} %{S-1-5-32-554} %{S-1-5-2} %{S-1-5-11} %{S-1-5-15} %{S-1-5-21-1377283216-344919071-3415362939-512} %{S-1-5-21-1377283216-344919071-3415362939-572} %{S-1-5-64-10} %{S-1-16-12288}</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2016, Windows 10.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that reported information about successful logon or invokes it. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that reported information about successful logon or invokes it.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4672](#)(S): Special privileges assigned to new logon."

Logon Type [Type = UInt32]: the type of logon which was performed. The table below contains the list of possible values for this field:

Logon Type	Logon Title	Description
2	Interactive	A user logged on to this computer.
3	Network	A user or computer logged on to this computer from the network.
4	Batch	Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct intervention.
5	Service	A service was started by the Service Control Manager.
7	Unlock	This workstation was unlocked.
8	NetworkCleartext	A user logged on to this computer from the network. The user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext).
9	NewCredentials	A caller cloned its current token and specified new credentials for outbound connections. The new logon session has the

		same local identity, but uses different credentials for other network connections.
10	RemoteInteractive	A user logged on to this computer remotely using Terminal Services or Remote Desktop.
11	CachedInteractive	A user logged on to this computer with network credentials that were stored locally on the computer. The domain controller was not contacted to verify the credentials.

New Logon:

- **Security ID** [Type = SID]: SID of account for which logon was performed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account for which logon was performed.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4672\(S\)](#): Special privileges assigned to new logon."

Event in sequence [Type = UInt32]: If is there is not enough space in one event to put all groups, you will see "**1 of N**" in this field and additional events will be generated. Typically this field has "**1 of 1**" value.

Group Membership [Type = UnicodeString]: the list of group SIDs which logged account belongs to (member of). Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Security Monitoring Recommendations:

For 4627(S): Group membership information.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
- **Typically this action is reported by the NULL SID account, so we recommend reporting all events with "Subject\Security ID" not equal "NULL SID".** If you need to track that a member of a specific group logged on to a computer, check the "**Group Membership**" field.

Audit IPsec Extended Mode

Audit IPsec Extended Mode allows you to audit events generated by Internet Key Exchange protocol (IKE) and Authenticated Internet Protocol (AuthIP) during Extended Mode negotiations.

Audit IPsec Extended Mode subcategory is out of scope of this document, because this subcategory is mainly used for IPsec Extended Mode troubleshooting.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Extended Mode troubleshooting, or for tracing or monitoring IPsec Extended Mode operations.
Member Server	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Extended Mode troubleshooting, or for tracing or monitoring IPsec Extended Mode operations.
Workstation	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Extended Mode troubleshooting, or for tracing or monitoring IPsec Extended Mode operations.

4978: During Extended Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.

4979: IPsec Main Mode and Extended Mode security associations were established.

4980: IPsec Main Mode and Extended Mode security associations were established.

4981: IPsec Main Mode and Extended Mode security associations were established.

4982: IPsec Main Mode and Extended Mode security associations were established.

4983: An IPsec Extended Mode negotiation failed. The corresponding Main Mode security association has been deleted.

4984: An IPsec Extended Mode negotiation failed. The corresponding Main Mode security association has been deleted.

Audit IPsec Main Mode

Audit IPsec Main Mode allows you to audit events generated by Internet Key Exchange protocol (IKE) and Authenticated Internet Protocol (AuthIP) during Main Mode negotiations. Audit IPsec Main Mode subcategory is out of scope of this document, because this subcategory is mainly used for IPsec Main Mode troubleshooting.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Main Mode troubleshooting, or for tracing or monitoring IPsec Main Mode operations.
Member Server	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Main Mode troubleshooting, or for tracing or monitoring IPsec Main Mode operations.
Workstation	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Main Mode troubleshooting, or for tracing or monitoring IPsec Main Mode operations.

4646: Security ID: %1

4650: An IPsec Main Mode security association was established. Extended Mode was not enabled. Certificate authentication was not used.

4651: An IPsec Main Mode security association was established. Extended Mode was not enabled. A certificate was used for authentication.

4652: An IPsec Main Mode negotiation failed.

4653: An IPsec Main Mode negotiation failed.

4655: An IPsec Main Mode security association ended.

4976: During Main Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.

5049: An IPsec Security Association was deleted.

5453: An IPsec negotiation with a remote computer failed because the IKE and AuthIP IPsec Keying Modules (IKEEXT) service is not started.

Audit IPsec Quick Mode

Audit IPsec Quick Mode allows you to audit events generated by Internet Key Exchange protocol (IKE) and Authenticated Internet Protocol (AuthIP) during Quick Mode negotiations. Audit IPsec Quick Mode subcategory is out of scope of this document, because this subcategory is mainly used for IPsec Quick Mode troubleshooting.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Quick Mode troubleshooting, or for tracing or monitoring IPsec Quick Mode operations.
Member Server	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Quick Mode troubleshooting, or for tracing or monitoring IPsec Quick Mode operations.
Workstation	IF	IF	IF	IF	IF - This subcategory is mainly used for IPsec Quick Mode troubleshooting, or for tracing or monitoring IPsec Quick Mode operations.

4977: During Quick Mode negotiation, IPsec received an invalid negotiation packet. If this problem persists, it could indicate a network issue or an attempt to modify or replay this negotiation.

5451: An IPsec Quick Mode security association was established.

5452: An IPsec Quick Mode security association ended.

Audit Logoff

Audit Logoff determines whether the operating system generates audit events when logon sessions are terminated.

These events occur on the computer that was accessed. In the case of an interactive logon, these events are generated on the computer that was logged on to.

There is no failure event in this subcategory because failed logoffs (such as when a system abruptly shuts down) do not generate an audit record.

Logon events are essential to understanding user activity and detecting potential attacks. Logoff events are not 100 percent reliable. For example, the computer can be turned off without a proper logoff and shutdown; in this case, a logoff event is not generated.

Event volume: Low.

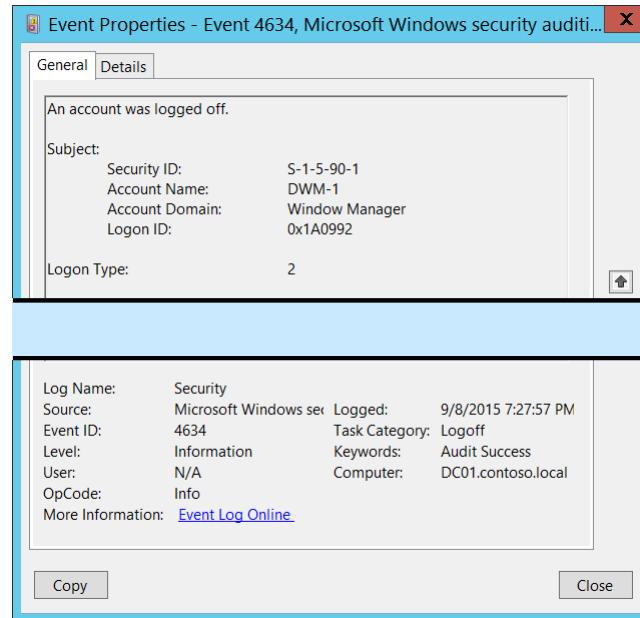
This subcategory allows you to audit events generated by the closing of a logon session. These events occur on the computer that was accessed. For an interactive logoff the security audit event is generated on the computer that the user account logged on to.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	Yes	No	<p>This subcategory typically generates huge amount of “4634(S): An account was logged off.” events which, typically has little security relevance. It is more important to audit Logon events using Audit Logon subcategory, rather than Logoff events.</p> <p>Enable Success audit if you want to track, for example, for how long session was active (in correlation with Audit Logon events) and when user actually logged off.</p> <p>This subcategory doesn’t have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	No	No	Yes	No	<p>This subcategory typically generates huge amount of “4634(S): An account was logged off.” events which, typically has little security relevance. It is more important to audit Logon events using Audit Logon subcategory, rather than Logoff events.</p> <p>Enable Success audit if you want to track, for example, for how long session was active (in correlation with Audit Logon events) and when user actually logged off.</p> <p>This subcategory doesn’t have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	No	No	Yes	No	<p>This subcategory typically generates huge amount of “4634(S): An account was logged off.” events which, typically has little security relevance. It is more important to audit Logon events using Audit Logon subcategory, rather than Logoff events.</p> <p>Enable Success audit if you want to track, for example, for how long session was active (in correlation with Audit Logon events) and when user actually logged off.</p> <p>This subcategory doesn’t have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4634](#)(S): An account was logged off.
- [4647](#)(S): User initiated logoff.

4634(S): An account was logged off.

 Event Properties - Event 4634, Microsoft Windows security audit... X

General **Details**

An account was logged off.

Subject:

Security ID:	S-1-5-90-1
Account Name:	DWM-1
Account Domain:	Window Manager
Logon ID:	0x1A0992

Logon Type: 2

Log Name: Security
Source: Microsoft Windows security
Event ID: 4634
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event shows that logon session was terminated and no longer exists.

The main difference between "[4647](#): User initiated logoff." and 4647 event is that 4647 event is generated when logoff procedure was initiated by specific account using logoff function, and 4634 event shows that session was terminated and no longer exists.

4647 is more typical for **Interactive** and **RemoteInteractive** logon types when user was logged off using standard methods.

You will typically see both 4647 and 4634 events when logoff procedure was initiated by user.

It may be positively correlated with a "[4624](#): An account was successfully logged on." event using the **Logon ID** value. Logon IDs are only unique between reboots on the same computer.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4634</EventID>
<Version>0</Version>
<Level>0</Level>
```

```
<Task>12545</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-09T02:27:57.877205900Z" />
<EventRecordID>230019</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="832" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserSid">S-1-5-90-1</Data>
<Data Name="TargetUserName">DWM-1</Data>
<Data Name="TargetDomainName">Window Manager</Data>
<Data Name="TargetLogonId">0x1a0992</Data>
<Data Name="LogonType">2</Data>
</EventData>
```

</Event>

Required Server Roles: None.**Minimum OS Version:** Windows Server 2008, Windows Vista.**Event Versions:** 0.**Field Descriptions:****Subject:**

- **Security ID** [Type = SID]: SID of account that was logged off. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that was logged off.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Logon Type [Type = UInt32]: the type of logon which was used. The table below contains the list of possible values for this field:

Logon Type	Logon Title	Description
2	Interactive	A user logged on to this computer.
3	Network	A user or computer logged on to this computer from the network.
4	Batch	Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct intervention.
5	Service	A service was started by the Service Control Manager.
7	Unlock	This workstation was unlocked.
8	NetworkCleartext	A user logged on to this computer from the network. The user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext).
9	NewCredentials	A caller cloned its current token and specified new credentials for outbound connections. The new logon session has the same local identity, but uses different credentials for other network connections.
10	RemoteInteractive	A user logged on to this computer remotely using Terminal Services or Remote Desktop.
11	CachedInteractive	A user logged on to this computer with network credentials that were stored locally on the computer. The domain

controller was not contacted to verify the credentials.

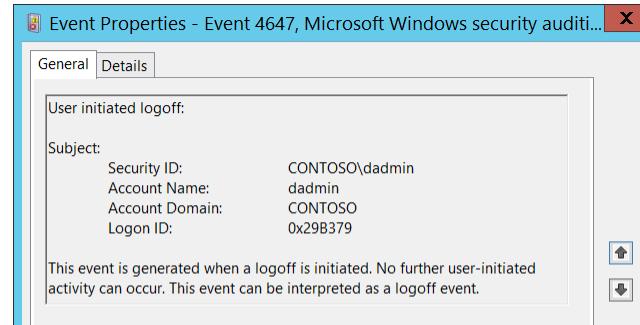
Security Monitoring Recommendations:

For 4634(S): An account was logged off.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If a particular **Logon Type** should not be used by a particular account (for example if **Logon Type** 4-Batch or 5-Service is used by a member of a domain administrative group), monitor this event for such actions.

4647(S): User initiated logoff.

 Event Properties - Event 4647, Microsoft Windows security auditi... X

General		Details	
<p>User initiated logoff:</p> <p>Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x29B379</p> <p>This event is generated when a logoff is initiated. No further user-initiated activity can occur. This event can be interpreted as a logoff event.</p>			
Link New Comments		Up Down	

Event Description:

This event is generated when a logoff is initiated. No further user-initiated activity can occur. This event can be interpreted as a logoff event.

The main difference with “[4634\(S\): An account was logged off.](#)” event is that 4647 event is generated when logoff procedure was initiated by specific account using logoff function, and 4634 event shows that session was terminated and no longer exists.

4647 is more typical for **Interactive** and **RemoteInteractive** logon types when user was logged off using standard methods.

You will typically see both 4647 and 4634 events when logoff procedure was initiated by user.

It may be positively correlated with a “[4624: An account was successfully logged on.](#)” event using the **Logon ID** value. Logon IDs are only unique between reboots on the same computer.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Level: Information	Keywords: Audit Success
User: N/A	Computer: DC01.contoso.local
Opcode: Info	
More Information: Event Log Online	
Copy Close	

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4647</EventID>
```

```

<Version>0</Version>
<Level>0</Level>
<Task>12545</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-09T03:08:39.126890800Z" />
<EventRecordID>230200</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="3864" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x29b379</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “logoff” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “logoff” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Security Monitoring Recommendations:

For 4647(S): User initiated logoff.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

Audit Logon

Audit Logon determines whether the operating system generates audit events when a user attempts to log on to a computer.

These events are related to the creation of logon sessions and occur on the computer that was accessed. For an interactive logon, events are generated on the computer that was logged on to. For a network logon, such as accessing a share, events are generated on the computer that hosts the resource that was accessed.

The following events are recorded:

- Logon success and failure.
- Logon attempts by using explicit credentials. This event is generated when a process attempts to log on an account by explicitly specifying that account's credentials. This most commonly occurs in batch configurations such as scheduled tasks, or when using the **RunAs** command.
- Security identifiers (SIDs) are filtered.

Logon events are essential to tracking user activity and detecting potential attacks.

Event volume:

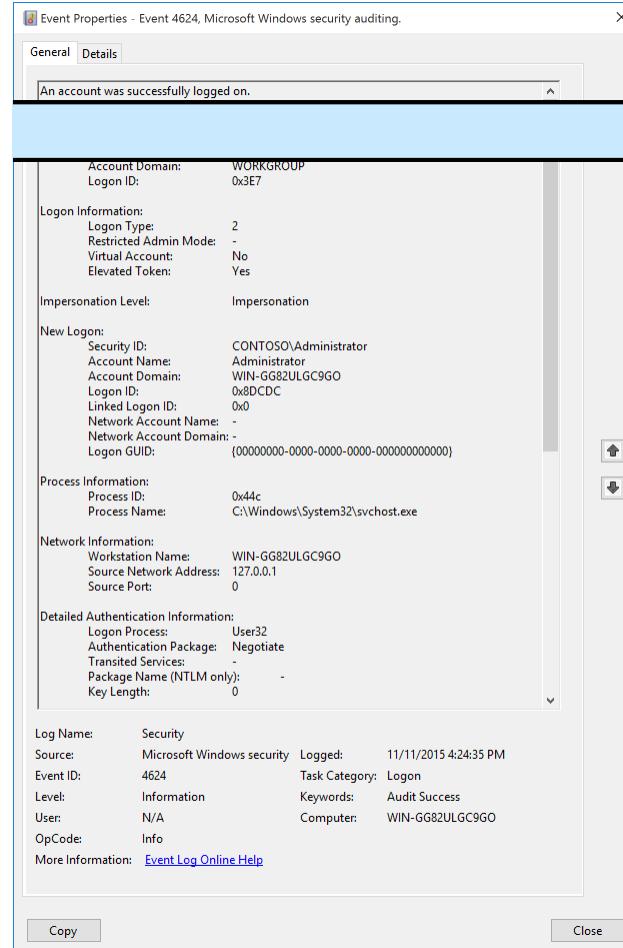
- Low on a client computer.
- Medium on a domain controllers or network servers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	Audit Logon events, for example, will give you information about which account, when, using which Logon Type, from which machine logged on to this machine. Failure events will show you failed logon attempts and the reason why these attempts failed.
Member Server	Yes	Yes	Yes	Yes	Audit Logon events, for example, will give you information about which account, when, using which Logon Type, from which machine logged on to this machine. Failure events will show you failed logon attempts and the reason why these attempts failed.
Workstation	Yes	Yes	Yes	Yes	Audit Logon events, for example, will give you information about which account, when, using which Logon Type, from which machine logged on to this machine. Failure events will show you failed logon attempts and the reason why these attempts failed.

Events List:

- [4624\(S\)](#): An account was successfully logged on.
- [4625\(F\)](#): An account failed to log on.
- [4648\(S\)](#): A logon was attempted using explicit credentials.
- [4675\(S\)](#): SIDs were filtered.

4624(\$): An account was successfully logged on.



Event Description:

This event generates when a logon session is created (on destination machine). It generates on the computer that was accessed, where the session was created.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4624</EventID>
<Version>2</Version>
<Level>0</Level>
<Task>12544</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T00:24:35.079785200Z" />
<EventRecordID>211</EventRecordID>
<Correlation ActivityID="{00D66690-1CDF-0000-AC66-D600DF1CD101}" />
<Execution ProcessID="716" ThreadID="760" />
<Channel>Security</Channel>
<Computer>WIN-GG82ULGC9GO</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">WIN-GG82ULGC9GO$</Data>
<Data Name="SubjectDomainName">WORKGROUP</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TargetUserSid">S-1-5-21-1377283216-344919071-3415362939-500</Data>
<Data Name="TargetUserName">Administrator</Data>
<Data Name="TargetDomainName">WIN-GG82ULGC9GO</Data>
<Data Name="TargetLogonId">0x8dcdc</Data>
<Data Name="LogonType">2</Data>
<Data Name="LogonProcessName">User32</Data>
<Data Name="AuthenticationPackageName">Negotiate</Data>
<Data Name="WorkstationName">WIN-GG82ULGC9GO</Data>
```

```
<Data Name="LogonGuid">{00000000-0000-0000-0000-000000000000}</Data>
<Data Name="TransmittedServices">-</Data>
<Data Name="LmPackageName">-</Data>
<Data Name="KeyLength">0</Data>
<Data Name="ProcessId">0x44c</Data>
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
<Data Name="IpAddress">127.0.0.1</Data>
<Data Name="IpPort">0</Data>
<Data Name="ImpersonationLevel">%%1833</Data>
<Data Name="RestrictedAdminMode">-</Data>
<Data Name="TargetOutboundUserName">-</Data>
<Data Name="TargetOutboundDomainName">-</Data>
<Data Name="VirtualAccount">%%1843</Data>
<Data Name="TargetLinkedLogonId">0x0</Data>
<Data Name="ElevatedToken">%%1842</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.
- 1 - Windows Server 2012, Windows 8.
 - Added “Impersonation Level” field.
- 2 – Windows 10.
 - Added “Logon Information:” section.
 - **Logon Type** moved to “Logon Information:” section.
 - Added “Restricted Admin Mode” field.
 - Added “Virtual Account” field.
 - Added “Elevated Token” field.
 - Added “Linked Logon ID” field.
 - Added “Network Account Name” field.
 - Added “Network Account Domain” field.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that reported information about successful logon or invokes it. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that reported information about successful logon.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4672\(S\)](#): Special privileges assigned to new logon."

Logon Information [Version 2]:

- **Logon Type** [Version 0, 1, 2] [Type = UInt32]: the type of logon which was performed. The table below contains the list of possible values for this field:

Logon Type	Logon Title	Description
2	Interactive	A user logged on to this computer.
3	Network	A user or computer logged on to this computer from the network.
4	Batch	Batch logon type is used by batch servers, where processes may be executing on behalf of a user without their direct intervention.
5	Service	A service was started by the Service Control Manager.
7	Unlock	This workstation was unlocked.
8	NetworkCleartext	A user logged on to this computer from the network. The user's password was passed to the authentication package in its unhashed form. The built-in authentication packages all hash credentials before sending them across the network. The credentials do not traverse the network in plaintext (also called cleartext).
9	NewCredentials	A caller cloned its current token and specified new credentials for outbound connections. The new logon session has the same local identity, but uses different credentials for other network connections.
10	RemoteInteractive	A user logged on to this computer remotely using Terminal Services or Remote Desktop.
11	CachedInteractive	A user logged on to this computer with network credentials that were stored locally on the computer. The domain controller was not contacted to verify the credentials.

- **Restricted Admin Mode** [Version 2] [Type = UnicodeString]: Only populated for **RemoteInteractive** logon type sessions. This is a Yes/No flag indicating if the credentials provided were passed using Restricted Admin mode. Restricted Admin mode was added in Win8.1/2012R2 but this flag was added to the event in Win10.
Reference: <http://blogs.technet.com/b/kfalde/archive/2013/08/14/restricted-admin-mode-for-rdp-in-windows-8-1-2012-r2.aspx>.
If not a **RemoteInteractive** logon, then this will be "-" string.
- **Virtual Account** [Version 2] [Type = UnicodeString]: a "Yes" or "No" flag, which indicates if the account is a virtual account (e.g., "[Managed Service Account](#)"), which was introduced in Windows 7 and Windows Server 2008 R2 to provide the ability to identify the account that a given Service uses, instead of just using "NetworkService".

- **Elevated Token [Version 2] [Type = UnicodeString]**: a “Yes” or “No” flag. If “Yes” then the session this event represents is elevated and has administrator privileges.

Impersonation Level [Version 1, 2] [Type = UnicodeString]: can have one of these four values:

- SecurityAnonymous (displayed as **empty string**): The server process cannot obtain identification information about the client, and it cannot impersonate the client. It is defined with no value given, and thus, by ANSI C rules, defaults to a value of zero.
- SecurityIdentification (displayed as "**Identification**"): The server process can obtain information about the client, such as security identifiers and privileges, but it cannot impersonate the client. This is useful for servers that export their own objects, for example, database products that export tables and views. Using the retrieved client-security information, the server can make access-validation decisions without being able to use other services that are using the client's security context.
- SecurityImpersonation (displayed as "**Impersonation**"): The server process can impersonate the client's security context on its local system. The server cannot impersonate the client on remote systems. This is the most common type.
- SecurityDelegation (displayed as "**Delegation**"): The server process can impersonate the client's security context on remote systems.

New Logon:

- **Security ID [Type = SID]**: SID of account for which logon was performed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]**: the name of the account for which logon was performed.
- **Account Domain [Type = UnicodeString]**: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID [Type = HexInt64]**: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4672](#)(S): Special privileges assigned to new logon.”
- **Linked Logon ID [Version 2] [Type = HexInt64]**: A hexadecimal value of the paired logon session. If there is no other logon session associated with this logon session, then the value is “0x0”.
- **Network Account Name [Version 2] [Type = UnicodeString]**: User name that will be used for outbound (network) connections. Valid only for [NewCredentials](#) logon type. If not [NewCredentials](#) logon, then this will be a “-” string.
- **Network Account Domain [Version 2] [Type = UnicodeString]**: Domain for the user that will be used for outbound (network) connections. Valid only for [NewCredentials](#) logon type. If not [NewCredentials](#) logon, then this will be a “-” string.
- **Logon GUID [Type = GUID]**: a GUID that can help you correlate this event with another event that can contain the same **Logon GUID**, “[4769](#)(S, F): A Kerberos service ticket was requested event on a domain controller.

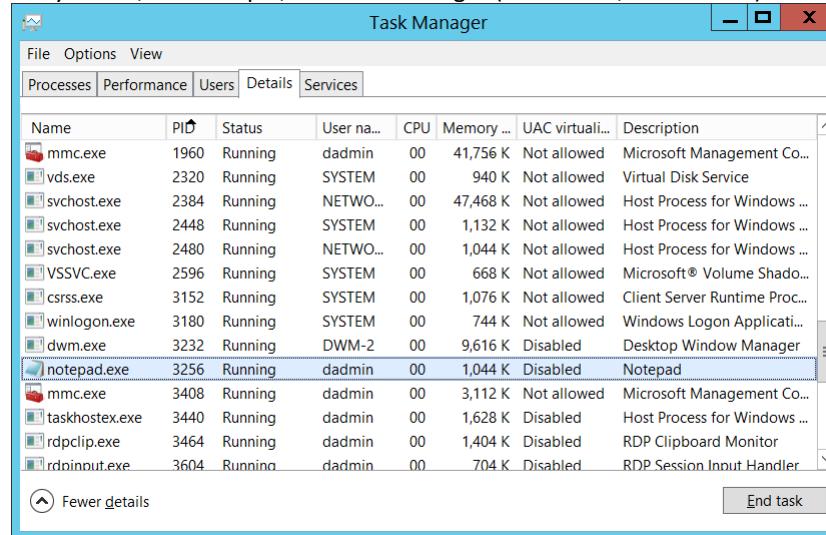
It also can be used for correlation between a 4624 event and several other events (on the same computer) that can contain the same **Logon GUID**, “[4648](#)(S): A logon was attempted using explicit credentials” and “[4964](#)(S): Special groups have been assigned to a new logon.”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Process Information:

- **Caller Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted the logon. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Caller Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Network Information:

- **Workstation Name** [Type = UnicodeString]: machine name from which logon attempt was performed.
- **Source Network Address** [Type = UnicodeString]: IP address of machine from which logon attempt was performed.
 - IPv6 address or ::ffff:IPv4 address of a client.
 - ::1 or 127.0.0.1 means localhost.
- **Source Port** [Type = UnicodeString]: source port which was used for logon attempt from remote machine.
 - 0 for interactive logons.

Detailed Authentication Information:

- **Logon Process** [Type = UnicodeString]: the name of the trusted logon process that was used for the logon. See event “[4611: A trusted logon process has been registered with the Local Security Authority](#)” description for more information.
- **Authentication Package** [Type = UnicodeString]: The name of the authentication package which was used for the logon authentication process. Default packages loaded on LSA startup are located in “HKLM\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig” registry key. Other packages can be loaded at runtime. When a new package is loaded a “[4610](#):

An authentication package has been loaded by the Local Security Authority” (typically for NTLM) or “[4622: A security package has been loaded by the Local Security Authority](#)” (typically for Kerberos) event is logged to indicate that a new package has been loaded along with the package name. The most common authentication packages are:

- **NTLM** – NTLM-family Authentication
 - **Kerberos** – Kerberos authentication.
 - **Negotiate** – the Negotiate security package selects between Kerberos and NTLM protocols. Negotiate selects Kerberos unless it cannot be used by one of the systems involved in the authentication or the calling application did not provide sufficient information to use Kerberos.
 - **Transited Services** [Type = UnicodeString] [Kerberos-only]: the list of transmitted services. Transmitted services are populated if the logon was a result of a S4U (Service For User) logon process. S4U is a Microsoft extension to the Kerberos Protocol to allow an application service to obtain a Kerberos service ticket on behalf of a user – most commonly done by a front-end website to access an internal resource on behalf of a user. For more information about S4U, see <https://msdn.microsoft.com/en-us/library/cc246072.aspx>
 - **Package Name (NTLM only)** [Type = UnicodeString]: The name of the LAN Manager sub-package ([NTLM-family](#) protocol name) that was used during logon. Possible values are:
 - “NTLM V1”
 - “NTLM V2”
 - “LM”
- Only populated if “Authentication Package” = “NTLM”.
- **Key Length** [Type = UInt32]: the length of [NTLM Session Security](#) key. Typically it has 128 bit or 56 bit length. This parameter is always 0 if “Authentication Package” = “Kerberos”, because it is not applicable for Kerberos protocol. This field will also have “0” value if Kerberos was negotiated using **Negotiate** authentication package.

Security Monitoring Recommendations:

For 4624(S): An account was successfully logged on.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ New Logon\Security ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ New Logon\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ New Logon\Security ID ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ New Logon\Security ID ” for accounts that are outside the whitelist.
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user	If this event corresponds to an action you want to monitor for certain account types,

account, vendor or employee account, and so on.	review the “ New Logon\Security ID ” to see whether the account type is as expected.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.	Monitor the target Computer : (or other target device) for actions performed by the “ New Logon\Security ID ” that you are concerned about.
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

- Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- If “**Restricted Admin**” mode must be used for logons by certain accounts, use this event to monitor logons by “**New Logon\Security ID**” in relation to “**Logon Type**=10 and “**Restricted Admin Mode**=“Yes”. If “**Restricted Admin Mode**=“No” for these accounts, trigger an alert.
- If you need to monitor all logon events for accounts with administrator privileges, monitor this event with “**Elevated Token**=“Yes”.
- If you need to monitor all logon events for managed service accounts and group managed service accounts, monitor for events with “**Virtual Account**=“Yes”.
- To monitor for a mismatch between the logon type and the account that uses it (for example, if **Logon Type** 4-Batch or 5-Service is used by a member of a domain administrative group), monitor **Logon Type** in this event.
- If your organization restricts logons in the following ways, you can use this event to monitor accordingly:
 - If the user account “**New Logon\Security ID**” should never be used to log on from the specific **Computer**.
 - If **New Logon\Security ID** credentials should not be used from **Workstation Name** or **Source Network Address**.
 - If a specific account, such as a service account, should only be used from your internal IP address list (or some other list of IP addresses). In this case, you can monitor for **Network Information\Source Network Address** and compare the network address with your list of IP addresses.
 - If a particular version of NTLM is always used in your organization. In this case, you can use this event to monitor **Package Name (NTLM only)**, for example, to find events where **Package Name (NTLM only)** does not equal **NTLM V2**.
 - If NTLM is not used in your organization, or should not be used by a specific account (**New Logon\Security ID**). In this case, monitor for all events where **Authentication Package** is NTLM.
 - If the **Authentication Package** is NTLM. In this case, monitor for **Key Length** not equal to 128, because all Windows operating systems starting with Windows 2000 support 128-bit Key Length.
- If you monitor for potentially malicious software, or software that is not authorized to request logon actions, monitor this event for **Process Name**.
- If you have a trusted logon processes list, monitor for a **Logon Process** that is not from the list.

4625(F): An account failed to log on.

This event also belongs in the **Audit Account Lockout** subcategory, and is described there. See “[4625\(F\): An account failed to log on](#)”.

4648(S): A logon was attempted using explicit credentials.

Event Properties - Event 4648, Microsoft Windows security auditing. X

General Details

A logon was attempted using explicit credentials.

Account Domain:	CONTOSO
Logon ID:	0x31844
Logon GUID:	{00000000-0000-0000-0000-000000000000}

Account Whose Credentials Were Used:

Account Name:	ladmin
Account Domain:	CONTOSO
Logon GUID:	{0887f1e4-39ea-d53c-804f-31d568a06274}

Target Server:

Target Server Name:	localhost
Additional Information:	localhost

Process Information:

Process ID:	0x368
Process Name:	C:\Windows\System32\svchost.exe

Network Information:

Network Address:	::1
Port:	0

This event is generated when a process attempts to log on an account by explicitly specifying that account's credentials. This most commonly occurs in batch-type configurations such as scheduled tasks, or when using the RUNAS command.

Log Name: Security
 Source: Microsoft Windows sec
 Event ID: 4648
 Level: Information
 User: N/A
 OpCode: Info
 Logged: 9/9/2015 7:54:50 PM
 Task Category: Logon
 Keywords: Audit Success
 Computer: DC01.contoso.local
[More Information: Event Log Online](#)

Copy Close

Event Description:

This event is generated when a process attempts an account logon by explicitly specifying that account's credentials. This most commonly occurs in batch-type configurations such as scheduled tasks, or when using the "RUNAS" command. It is also a routine event which periodically occurs during normal operating system activity.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4648</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12544</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-10T02:54:50.771459000Z" />
<EventRecordID>233200</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="1116" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">ladmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x31844</Data>

```

```

<Data Name="LogonGuid">{00000000-0000-0000-0000-000000000000}</Data>
<Data Name="TargetUserName">ladmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonGuid">{0887F1E4-39EA-D53C-804F-31D568A06274}</Data>
<Data Name="TargetServerName">localhost</Data>
<Data Name="TargetInfo">localhost</Data>
<Data Name="ProcessId">0x368</Data>

```

```
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
<Data Name="IpAddress">::1</Data>
<Data Name="IpPort">0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the new logon session with explicit credentials. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the new logon session with explicit credentials.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."
- **Logon GUID** [Type = GUID]: a GUID that can help you correlate this event with another event that can contain the same **Logon GUID**, "[4769](#)(S, F): A Kerberos service ticket was requested event on a domain controller.

It also can be used for correlation between a 4648 event and several other events (on the same computer) that can contain the same **Logon GUID**, "[4624](#)(S): An account was successfully logged on" and "[4964](#)(S): Special groups have been assigned to a new logon."

This parameter might not be captured in the event, and in that case appears as "{00000000-0000-0000-0000-000000000000}".

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Account Whose Credentials Were Used:

- **Account Name** [Type = UnicodeString]: the name of the account whose credentials were used.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local

- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon GUID** [Type = GUID]: a GUID that can help you correlate this event with another event that can contain the same **Logon GUID**, “[4769](#)(S, F): A Kerberos service ticket was requested event on a domain controller.

It also can be used for correlation between a 4648 event and several other events (on the same computer) that can contain the same **Logon GUID**, “[4624](#)(S): An account was successfully logged on” and “[4964](#)(S): Special groups have been assigned to a new logon.”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

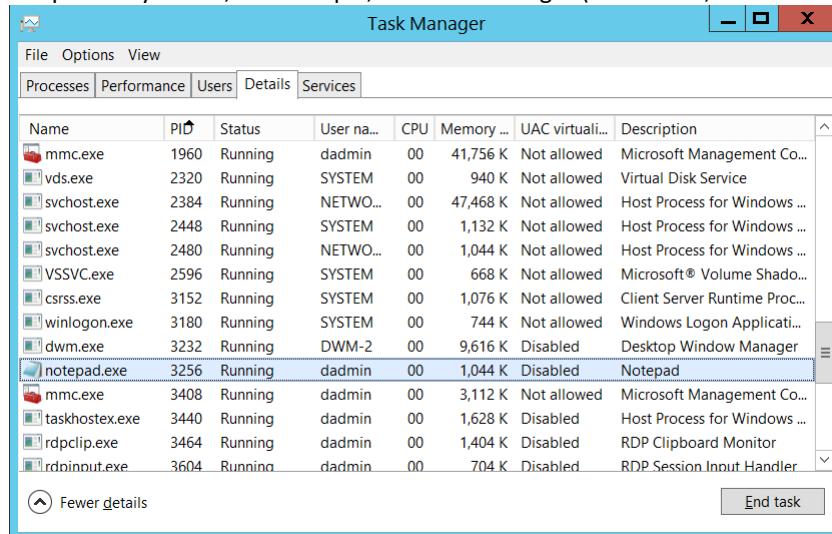
GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Target Server:

- **Target Server Name** [Type = UnicodeString]: the name of the server on which the new process was run. Has “localhost” value if the process was run locally.
- **Additional Information** [Type = UnicodeString]: there is no detailed information about this field in this document.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process which was run using explicit credentials. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Network Information:

- **Network Address** [Type = UnicodeString]: IP address of machine from which logon attempt was performed.

- IPv6 address or ::ffff:IPv4 address of a client.
- ::1 or 127.0.0.1 means localhost.
- **Port** [Type = UnicodeString]: source port which was used for logon attempt from remote machine.
 - 0 for interactive logons.

Security Monitoring Recommendations:

For 4648(S): A logon was attempted using explicit credentials.

The following table is similar to the table in [General recommendations for security auditing and monitoring for Windows 10](#), but also describes ways of monitoring that use “**Account Whose Credentials Were Used\Security ID**.”

Type of monitoring required	Recommendation
High-value accounts: You might have high value domain or local accounts for which you need to monitor each action. Examples of high value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Subject\Security ID ” or “ Account Whose Credentials Were Used\Security ID ” that correspond to the high value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” and “ Account Whose Credentials Were Used\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Security ID ” or “ Account Whose Credentials Were Used\Security ID ” that correspond to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” and “ Account Whose Credentials Were Used\Security ID ” for accounts that are outside the whitelist.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform the action corresponding to this event.	Monitor for the “ Subject\Account Domain ” or “ Account Whose Credentials Were Used\Security ID ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.	Monitor the target Computer : (or other target device) for actions performed by the “ Subject\Security ID ” or “ Account Whose Credentials Were Used\Security ID ” that you are concerned about. For example, you might monitor to ensure that “ Account Whose Credentials Were Used\Security ID ” is not used to log on to a certain computer.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Account Name**” and “**Account Whose Credentials Were Used\Security ID**” for names that don’t comply with naming conventions.

-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If **Subject\Security ID** should not know or use credentials for **Account Whose Credentials Were Used\Account Name**, monitor this event.
- If credentials for **Account Whose Credentials Were Used\Account Name** should not be used from **Network Information\Network Address**, monitor this event.
- Check that **Network Information\Network Address** is from internal IP address list. For example, if you know that a specific account (for example, a service account) should be used only from specific IP addresses, you can monitor for all events where **Network Information\Network Address** is not one of the allowed IP addresses.

4675(S): SIDs were filtered.

This event generates when SIDs were filtered for specific Active Directory trust.

See more information about SID filtering here: [https://technet.microsoft.com/en-us/library/cc772633\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc772633(v=ws.10).aspx).

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

There is no example of this event in this document.

Event Schema:

SIDs were filtered.

Target Account:

Security ID:%1
Account Name:%2
Account Domain:%3

Trust Information:

Trust Direction:%4
Trust Attributes:%5
Trust Type:%6
TDO Domain SID:%7
Filtered SIDs:%8

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- If you need to monitor all SID filtering events/operations for specific or all Active Directory trusts, you can use this event to get all required information.

Audit Network Policy Server

Audit Network Policy Server allows you to audit events generated by RADIUS (IAS) and Network Access Protection (NAP) activity related to user access requests. These requests can be Grant, Deny, Discard, Quarantine, Lock, and Unlock.

If you configure this subcategory, an audit event is generated for each IAS and NAP user access request.

This subcategory generates events only if NAS or IAS role is installed on the server.

NAP events can be used to help understand the overall health of the network.

Event volume: Medium to High on servers that are running [Network Policy Server](#) (NPS).

Role-specific subcategories are outside the scope of this document.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF – if a server has the Network Policy Server (NPS) role installed and you need to monitor access requests and other NPS-related events, enable this subcategory.
Member Server	IF	IF	IF	IF	IF – if a server has the Network Policy Server (NPS) role installed and you need to monitor access requests and other NPS-related events, enable this subcategory.
Workstation	No	No	No	No	Network Policy Server (NPS) role cannot be installed on client OS.

6272: Network Policy Server granted access to a user.

6273: Network Policy Server denied access to a user.

6274: Network Policy Server discarded the request for a user.

6275: Network Policy Server discarded the accounting request for a user.

6276: Network Policy Server quarantined a user.

6277: Network Policy Server granted access to a user but put it on probation because the host did not meet the defined health policy.

6278: Network Policy Server granted full access to a user because the host met the defined health policy.

6279: Network Policy Server locked the user account due to repeated failed authentication attempts.

6280: Network Policy Server unlocked the user account.

Audit Other Logon/Logoff Events

Audit Other Logon/Logoff Events determines whether Windows generates audit events for other logon or logoff events.

These other logon or logoff events include:

- A Remote Desktop session connects or disconnects.
- A workstation is locked or unlocked.
- A screen saver is invoked or dismissed.
- A replay attack is detected. This event indicates that a Kerberos request was received twice with identical information. This condition could also be caused by network misconfiguration.
- A user is granted access to a wireless network. It can be either a user account or the computer account.
- A user is granted access to a wired 802.1x network. It can be either a user account or the computer account.

Logon events are essential to understanding user activity and detecting potential attacks.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	We recommend Success auditing, to track possible Kerberos replay attacks, terminal session connect and disconnect actions, network authentication events, and some other events. Volume of these events is typically very low. Failure events will show you when requested credentials CredSSP delegation was disallowed by policy. The volume of these events is very low—typically you will not get any of these events.
Member Server	Yes	Yes	Yes	Yes	We recommend Success auditing, to track possible terminal session connect and disconnect actions, network authentication events, and some other events. Volume of these events is typically very low. Failure events will show you when requested credentials CredSSP delegation was disallowed by policy. The volume of these events is very low—typically you will not get any of these events.
Workstation	Yes	Yes	Yes	Yes	We recommend Success auditing, to track possible terminal session connect and disconnect actions, network authentication events, and some other events. Volume of these events is typically very low. Failure events will show you when requested credentials CredSSP delegation was disallowed by policy. The volume of these events is very low—typically you will not get any of these events.

Events List:

- [4649\(S\)](#): A replay attack was detected.
- [4778\(S\)](#): A session was reconnected to a Window Station.
- [4779\(S\)](#): A session was disconnected from a Window Station.
- [4800\(S\)](#): The workstation was locked.
- [4801\(S\)](#): The workstation was unlocked.
- [4802\(S\)](#): The screen saver was invoked.
- [4803\(S\)](#): The screen saver was dismissed.

- [5378\(F\)](#): The requested credentials delegation was disallowed by policy.
- [5632\(S\)](#): A request was made to authenticate to a wireless network.
- [5633\(S\)](#): A request was made to authenticate to a wired network.

4649(S): A replay attack was detected.

This event generates on domain controllers when **KRB_AP_ERR_REPEAT** Kerberos response was sent to the client.

Domain controllers cache information from recently received tickets. If the server name, client name, time, and microsecond fields from the Authenticator match recently seen entries in the cache, it will return KRB_AP_ERR_REPEAT. You can read more about this in [RFC-1510](#). One potential cause for this is a misconfigured network device between the client and server that could send the same packet(s) repeatedly.

There is no example of this event in this document.

Event Schema:

A replay attack was detected.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Credentials Which Were Replayed:

*Account Name:%5
Account Domain:%6*

Process Information:

*Process ID:%12
Process Name:%13*

Network Information:

Workstation Name:%10

Detailed Authentication Information:

*Request Type:%7
Logon Process:%8
Authentication Package:%9
Transited Services:%11*

This event indicates that a Kerberos replay attack was detected- a request was received twice with identical information. This condition could be caused by network misconfiguration."

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

For 4649(S): A replay attack was detected.

- This event can be a sign of Kerberos replay attack or, among other things, network device configuration or routing problems. In both cases, we recommend triggering an alert and investigating the reason the event was generated.

4778(S): A session was reconnected to a Window Station.

Event Properties - Event 4778, Microsoft Windows security auditi... X

General	Details
<p>A session was reconnected to a Window Station.</p>	

Event Description:

This event is generated when a user reconnects to an existing Terminal Services session, or when a user switches to an existing desktop using [Fast User Switching](#). This event also generates when user reconnects to virtual host Hyper-V Enhanced Session, for example.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4778</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12551</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-10T23:05:29.743867200Z" />
<EventRecordID>237651</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="2212" />
<Channel>Security</Channel>

```

<Computer>DC01.contoso.local</Computer>

<Security />

</System>

- <EventData>

<Data Name="AccountName">ladmin</Data>

<Data Name="AccountDomain">CONTOSO</Data>

<Data Name="LogonID">0x1e01f6</Data>

```
<Data Name="SessionName">RDP-Tcp#6</Data>
<Data Name="ClientName">WIN81</Data>
<Data Name="ClientAddress">10.0.0.100</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Account Name** [Type = UnicodeString]: the name of the account for which the session was reconnected.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

Session:

- **Session Name** [Type = UnicodeString]: the name of the session to which the user was reconnected. Examples:
 - RDP-Rcp#N, where N is a number of session – typical RDP session name.
 - **Console** – console session, typical for Fast User Switching.
 - **31C5CE94259D4006A9E4#3** – example of "Hyper-V Enhanced Session" session name.

You can see the list of current session's using "**query session**" command in command prompt. Example of output (see **SESSIONNAME** column):

C:\windows\system32>query session	SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
	services		0	Disc		
	console		1	Conn		
	>31c5ce94259d4...	dadmin	3	Active		
	31c5ce94259d4...		65536	Listen		
	rdp-tcp		65537	Listen		

Additional Information:

- **Client Name** [Type = UnicodeString]: computer name from which the user was reconnected. Has "**Unknown**" value for console session.
- **Client Address** [Type = UnicodeString]: IP address of the computer from which the user was reconnected.
 - IPv6 address or ::ffff:IPv4 address of a client.
 - ::1 or 127.0.0.1 means localhost.
 - Has "**LOCAL**" value for console session.

Security Monitoring Recommendations:

For 4778(S): A session was reconnected to a Window Station.

Type of monitoring required	Recommendation
<p>High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.</p> <p>Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.</p>	Monitor this event with the “ Subject\Account Name ” that corresponds to the high-value account or accounts.
<p>Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.</p>	When you monitor for anomalies or malicious actions, use the “ Subject\Account Name ” (with other information) to monitor how or when a particular account is being used.
<p>Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.</p>	Monitor this event with the “ Subject\Account Name ” that corresponds to the accounts that should never be used.
<p>Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.</p>	If this event corresponds to a “whitelist-only” action, review the “ Subject\Account Name ” for accounts that are outside the whitelist.
<p>Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.</p>	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Account Name ” to see whether the account type is as expected.
<p>External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).</p>	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
<p>Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.</p>	Monitor the target Computer: (or other target device) for actions performed by the “ Subject\Account Name ” that you are concerned about.
<p>Account naming conventions: Your organization might have specific naming conventions for account names.</p>	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

- If Fast User Switching is disabled on workstations or specific computers, then monitor for any event with **Session Name** = Console.
- If Remote Desktop Connections are not allowed for specific users (**Subject\Account Name**) or disabled on some computers, then monitor for **Session Name** = RDP-Tcp# (substring).
- If a specific computer or device (**Client Name** or **Client Address**) should never connect to this computer (**Computer**), monitor for any event with that **Client Name** or **Client Address**.
- Check that **Additional Information\Client Address** is from internal IP addresses list.

4779(S): A session was disconnected from a Window Station.

Event Properties - Event 4779, Microsoft Windows security audit... X

General **Details**

A session was disconnected from a Window Station.

Logon ID:	0x1E01F6
Session:	Session Name: RDP-Tcp#3
Additional Information:	
Client Name:	WIN81
Client Address:	10.0.0.100

This event is generated when a user disconnects from an existing Terminal Services session, or when a user switches away from an existing desktop using Fast User Switching.

Log Name: Security	Source: Microsoft Windows se	Logged: 9/10/2015 4:04:41 PM
Event ID: 4779	Task Category: Other Logon/Logoff	
Level: Information	Keywords: Audit Success	
User: N/A	Computer: DC01.contoso.local	
OpCode: Info		
More Information: Event Log Online		

Copy **Close**

Event Description:

This event is generated when a user disconnects from an existing Terminal Services session, or when a user switches away from an existing desktop using [Fast User Switching](#).

This event also generated when user disconnects from virtual host Hyper-V Enhanced Session, for example.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4779</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12551</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-10T23:04:41.044489800Z" />
<EventRecordID>237646</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="AccountName">ladmin</Data>
<Data Name="AccountDomain">CONTOSO</Data>
<Data Name="LogonID">0x1e01f6</Data>
<Data Name="SessionName">RDP-Tcp#3</Data>
<Data Name="ClientName">WIN81</Data>
<Data Name="ClientAddress">10.0.0.100</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Account Name** [Type = UnicodeString]: the name of the account for which the session was disconnected.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Session:

- **Session Name** [Type = UnicodeString]: the name of disconnected session. Examples:
 - RDP-Rcp#N, where N is a number of session – typical RDP session name.
 - Console – console session, typical for Fast User Switching.
 - 31C5CE94259D4006A9E4#3 – example of "Hyper-V Enhanced Session" session name.

You can see the list of current session's using "**query session**" command in command prompt. Example of output (see **SESSIONNAME** column):

C:\windows\system32>query session	SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
	services		0	Disc		
	console		1	Conn		
	>31c5ce94259d4...	dadmin	3	Active		
	31c5ce94259d4...		65536	Listen		
	rdp-tcp		65537	Listen		

Additional Information:

- **Client Name** [Type = UnicodeString]: machine name from which the session was disconnected. Has "Unknown" value for console session.
- **Client Address** [Type = UnicodeString]: IP address of the computer from which the session was disconnected.
 - IPv6 address or ::ffff:IPv4 address of a client.
 - ::1 or 127.0.0.1 means localhost.
 - Has "LOCAL" value for console session.

Security Monitoring Recommendations:

For 4779(S): A session was disconnected from a Window Station.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the " Subject\Account Name " that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting	When you monitor for anomalies or malicious actions, use the " Subject\Account Name "

anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	(with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Account Name ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Account Name ” for accounts that are outside the whitelist.
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Account Name ” to see whether the account type is as expected.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions. For example, you might have computers to which connections should not be made from certain accounts or addresses.	Monitor the target Computer: (or other target device) for actions performed by the “ Subject\Account Name ” that you are concerned about. If you have a target Computer: (or other target device) to which connections should not be made from certain accounts or addresses, monitor this event for the corresponding Client Name or Client Address .
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

- If Fast User Switching is disabled on workstations or specific computers, then monitor for any event with **Session Name** = Console.
- If Remote Desktop Connections are not allowed for specific users (**Subject\Account Name**) or disabled on some computers, then monitor for **Session Name** = RDP-Tcp# (substring).
- To ensure that connections are made only from your internal IP address list, monitor the **Additional Information\Client Address** in this event.

4800(S): The workstation was locked.

Event Properties - Event 4800, Microsoft Windows security audit... X

General	Details
<p>Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x759A9 Session ID: 3</p> <p>Log Name: Security Source: Microsoft Windows security Event ID: 4800 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p>	<p>Logged: 9/10/2015 4:47:02 PM Task Category: Other Logon/Logoff Keywords: Audit Success Computer: DC01.contoso.local</p> <p>Copy Close</p>

Event Description:

This event is generated when a workstation was locked.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4800</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12551</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-10T23:47:02.430644500Z" />
<EventRecordID>237655</EventRecordID>

```

```

<Correlation />
<Execution ProcessID="504" ThreadID="2568" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="TargetUserName">dadmin</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetLogonId">0x759a9</Data>
  <Data Name="SessionId">3</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

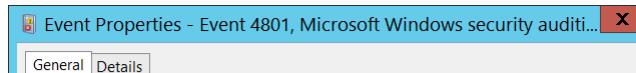
Subject:

- **Security ID** [Type = SID]: SID of account that requested the “lock workstation” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “lock workstation” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”
- **Session ID** [Type = UInt32]: unique ID of locked session. You can see the list of current session IDs using “**query session**” command in command prompt. Example of output (see **ID** column):

```
c:\windows\system32>query session
SESSIONNAME      USERNAME              ID  STATE   TYPE      DEVICE
services          services              0  Disc
console          console               1  Conn
>31c5ce94259d4...  dadmin              3  Active
31c5ce94259d4...          Listen            65536 Listen
rdp-tcp           rdp-tcp             65537 Listen
```



Security Monitoring Recommendations:

For 4800(S): The workstation was locked.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. Typically this is an informational event, and can give you information about when a machine was locked, and which account was used to lock it.

4801(S): The workstation was unlocked.

Event Description:

This event is generated when workstation was unlocked.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4801</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12551</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-10T23:47:05.886096400Z" />
<EventRecordID>237657</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="4540" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x759a9</Data>
<Data Name="SessionId">3</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “unlock workstation” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “unlock workstation” operation.

- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."
- **Session ID** [Type = UInt32]: unique ID of unlocked session. You can see the list of current session IDs using "query session" command in command prompt. Example of output (see **ID** column):

```
C:\>query session
SESSIONNAME      USERNAME              ID  STATE   TYPE      DEVICE
services          services              0   Disc
console           console               1   Conn
>31c5ce94259d4...  dadmin              3   Active
31c5ce94259d4...                               65536 Listen
rdp-tcp          rdp-tcp              65537 Listen
```

Security Monitoring Recommendations:

For 4801(S): The workstation was unlocked.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Typically this is an informational event, and can give you information about when a machine was unlocked, and which account was used to unlock it.

4802(S): The screen saver was invoked.

Event Description:

This event is generated when screen saver was invoked.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event Properties - Event 4802, Microsoft Windows security audit...

General	Details
Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x759A9 Session ID: 3	
Log Name: Security Source: Microsoft Windows security audit Event ID: 4802 Level: Information User: N/A OpCode: Info More Information: Event Log Online	
Logged: 9/10/2015 5:16:32 PM Task Category: Other Logon/Logoff Keywords: Audit Success Computer: DC01.contoso.local	
<input type="button" value="Copy"/> <input type="button" value="Close"/>	

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4802</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12551</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

```
<TimeCreated SystemTime="2015-09-11T00:16:32.377883700Z" />
<EventRecordID>237662</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="1676" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x759a9</Data>
<Data Name="SessionId">3</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “invoke screensaver” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “invoke screensaver” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

- **Session ID** [Type = UInt32]: unique ID of a session for which screen saver was invoked. You can see the list of current session IDs using “**query session**” command in command prompt. Example of output (see **ID** column):

```
C:\>query session
SESSIONNAME      USERNAME              ID  STATE   TYPE      DEVICE
services          services              0   Disc
console           console               1   Conn
>31c5ce94259d4...  dadmin              3   Active
31c5ce94259d4...                               65536 Listen
rdp-tcp          rdp-tcp              65537 Listen
```

Security Monitoring Recommendations:

For 4802(S): The screen saver was invoked.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Typically this is an informational event, and can give you information about when a screen saver was invoked on a machine, and which account invoked it.

4803(S): The screen saver was dismissed.

 Event Properties - Event 4803, Microsoft Windows security audit... X

General Details

Event Description:
This event is generated when screen saver was dismissed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x759A9 Session ID: 3	Event XML: - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>4803</EventID> <Version>0</Version> <Level>0</Level> <Task>12551</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-09-11T00:19:09.576094500Z" /> <EventRecordID>237663</EventRecordID>
Log Name: Security Source: Microsoft Windows sev Event ID: 4803 Level: Information User: N/A OpCode: Info More Information: Event Log Online	
Copy Close	

```
<Correlation />
<Execution ProcessID="504" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
```

```
- <EventData>
<Data Name="TargetUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="TargetUserName">dadmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x759a9</Data>
<Data Name="SessionId">3</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “dismiss screensaver” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “dismiss screensaver” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”
- **Session ID** [Type = UInt32]: unique ID of a session for which screen saver was dismissed. You can see the list of current session IDs using “**query session**” command in command prompt. Example of output (see **ID** column):

C:\windows\system32>query session	SESSIONNAME	USERNAME	ID	STATE	TYPE	DEVICE
	services		0	Disc		
	console		1	Conn		
>31c5ce94259d4...	dadmin		3	Active		
	31c5ce94259d4...		65536	Listen		
	rdp-tcp		65537	Listen		

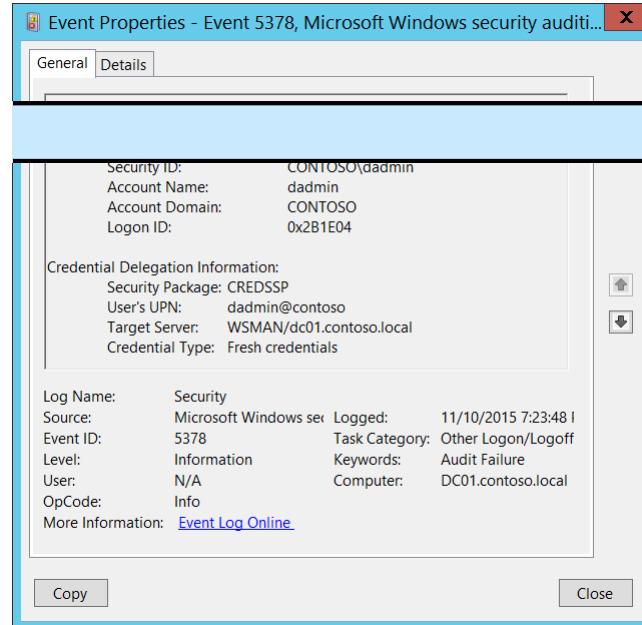
Security Monitoring Recommendations:

For 4803(S): The screen saver was dismissed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Typically this is an informational event, and can give you information about when a screen saver was dismissed on a machine, and which account dismissed it.

5378(F): The requested credentials delegation was disallowed by policy.

 Event Properties - Event 5378, Microsoft Windows security audit... X

<p>General Details</p> <p>Security ID: CONTOSO\admind Account Name: admind Account Domain: CONTOSO Logon ID: 0x2B1E04</p> <p>Credential Delegation Information: Security Package: CREDSSP User's UPN: admind@contoso Target Server: WSMAN/dc01.contoso.local Credential Type: Fresh credentials</p> <p>Log Name: Security Source: Microsoft Windows security audit Event ID: 5378 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p> <p style="text-align: right;">Copy Close</p>	<p>Event Description: This event generates requested CredSSP credentials delegation was disallowed by CredSSP delegation policy. It typically occurs when CredSSP delegation for WinRM double-hop session was not set properly.</p> <p>Note For recommendations, see Security Monitoring Recommendations for this event.</p> <p>Event XML:</p> <pre> - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>5378</EventID> <Version>0</Version> <Level>0</Level> <Task>12551</Task> <Opcode>0</Opcode> <Keywords>0x8010000000000000</Keywords> <TimeCreated SystemTime="2015-11-11T03:23:48.502346900Z" /> <EventRecordID>1198733</EventRecordID> <Correlation /> <Execution ProcessID="500" ThreadID="4308" /></pre>
---	---

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">admind</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x2b1e04</Data>
<Data Name="Package">CREDSSP</Data>
<Data Name="UserUPN">admind@contoso</Data>
<Data Name="TargetServer">WSMAN/dc01.contoso.local</Data>
```

```
<Data Name="CredType">%&8098</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested credentials delegation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested credentials delegation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Credential Delegation Information:

- **Security Package** [Type = UnicodeString]: the name of [Security Package](#) which was used. Always **CREDSSP** for this event.
- **User's UPN** [Type = UnicodeString]: [UPN](#) of the account for which delegation was requested.
- **Target Server** [Type = UnicodeString]: SPN of the target service for which delegation was requested.

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Credential Type** [Type = UnicodeString]: types of credentials which were presented for delegation:

Credentials Type	Description
Default credentials	The credentials obtained when the user first logs on to Windows.
Fresh credentials	The credentials that the user is prompted for when executing an application.
Saved credentials	The credentials that are saved using Credential Manager .

Security Monitoring Recommendations:

For 5378(F): The requested credentials delegation was disallowed by policy.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have defined CredSSP delegation policy, then this event will show you policy violations. We recommend collecting these events and investigating every policy violation.
- This event also can be used for CredSSP delegation troubleshooting.

5632(S, F): A request was made to authenticate to a wireless network.

Event Properties - Event 5632, Microsoft Windows security auditing.

Event Description:
This event generates when [802.1x](#) authentication attempt was made for wireless network. It typically generates when network adapter connects to new wireless network.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

<p>Security ID: host - redmond.corp.microsoft.com</p> <p>Account Name: -</p> <p>Account Domain: -</p> <p>Logon ID: 0x0</p> <p>Network Information:</p> <p>Name (SSID): Nokia</p> <p>Interface GUID: {2bb33827-6bb6-48db-8de6-db9e0b9f9c9b}</p> <p>Local MAC Address: 02:1A:C5:14:59:C9</p> <p>Peer MAC Address: 18:64:72:F3:33:91</p> <p>Additional Information:</p> <p>Reason Code: The operation was successful. (0x0)</p> <p>Error Code: 0x0</p> <p>EAP Reason Code: 0x0</p> <p>EAP Root Cause String: EAP Error Code: 0x0</p> <p>Log Name: Security</p> <p>Source: Microsoft Windows security Logged: 11/10/2015 3:10:34 PM</p> <p>Event ID: 5632 Task Category: Other Logon/Logoff Events</p> <p>Level: Information Keywords: Audit Success</p> <p>User: N/A Computer: redmond.corp.n</p> <p>OpCode: Info</p> <p>More Information: Event Log Online Help</p>	<p>Event XML:</p> <pre> - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>5632</EventID> <Version>1</Version> <Level>0</Level> <Task>12551</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-11-10T23:10:34.052054800Z" /> <EventRecordID>44113845</EventRecordID> <Correlation /> <Execution ProcessID="712" ThreadID="4176" /> <Channel>Security</Channel> <Computer>XXXXXXXX.redmond.corp.microsoft.com</Computer> <Security /></pre>
---	---

```

</System>
- <EventData>
<Data Name="SSID">Nokia</Data>
<Data Name="Identity">host/XXXXXXXX.redmond.corp.microsoft.com</Data>

```

```
<Data Name="SubjectUserName">-</Data>
<Data Name="SubjectDomainName">-</Data>
<Data Name="SubjectLogonId">0x0</Data>
<Data Name="PeerMac">18:64:72:F3:33:91</Data>
<Data Name="LocalMac">02:1A:C5:14:59:C9</Data>
<Data Name="IntfGuid">{2BB33827-6BB6-48DB-8DE6-DB9E0B9F9C9B}</Data>
<Data Name="ReasonCode">0x0</Data>
<Data Name="ReasonText">The operation was successful.</Data>
<Data Name="ErrorCode">0x0</Data>
<Data Name="EAPReasonCode">0x0</Data>
<Data Name="EapRootCauseString" />
<Data Name="EAPErrorCode">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = UnicodeString]: User Principal Name (UPN) or another type of account identifier for which 802.1x authentication request was made.

User principal name (UPN) format is used to specify an Internet-style name, such as `UserName@Example.Microsoft.com`.
- **Account Name** [Type = UnicodeString]: the name of the account for which 802.1x authentication request was made.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: `CONTOSO`
 - Lowercase full domain name: `contoso.local`
 - Uppercase full domain name: `CONTOSO.LOCAL`
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Network Information:

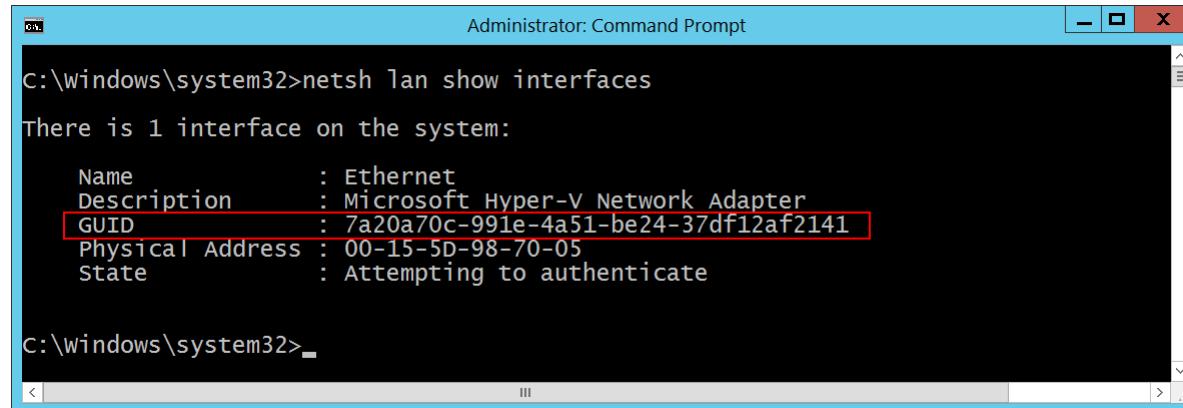
- **Name (SSID)** [Type = UnicodeString]: SSID of the wireless network to which authentication request was sent.

A service set identifier (SSID) is a sequence of characters that uniquely names a wireless local area network (WLAN). An SSID is sometimes referred to as a "network name." This name allows stations to connect to the desired network when multiple independent networks operate in the same physical area.
- **Interface GUID** [Type = GUID]: GUID of the network interface which was used for authentication request.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

You can see interface's GUID using the following commands:

- o "netsh lan show interfaces" – for wired interfaces.
- o "netsh wlan show interfaces" – for wireless interfaces.



```
Administrator: Command Prompt
c:\windows\system32>netsh lan show interfaces
There is 1 interface on the system:
  Name          : Ethernet
  Description   : Microsoft Hyper-V Network Adapter
  GUID          : 7a20a70c-991e-4a51-be24-37df12af2141
  Physical Address: 00-15-5D-98-70-05
  State         : Attempting to authenticate

C:\windows\system32>
```

Event Properties - Event 5633, Microsoft Windows security auditing.

General **Details**

A request was made to authenticate to a wired network.

Subject:

- Security ID: -
- Account Name: -
- Account Domain: -
- Logon ID: 0x0

Interface:

- Name: Microsoft Hyper-V Network Adapter

Additional Information

Reason Code:	The network does not support authentication (0x70003)
Error Code:	0x0

Log Name: Security
Source: Microsoft Windows security
Event ID: 5633
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

- **Local MAC Address** [Type = UnicodeString]: local interface's MAC-address.
 - **Peer MAC Address** [Type = UnicodeString]: peer's (typically – access point) MAC-address.
- Additional Information:**
- **Reason Code** [Type = UnicodeString]: contains Reason Text (explanation of Reason Code) and Reason Code for wireless authentication results. See more information about reason codes for wireless authentication here:
[https://msdn.microsoft.com/en-us/library/windows/desktop/dd877212\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd877212(v=vs.85).aspx),
[https://technet.microsoft.com/en-us/library/cc727747\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc727747(v=ws.10).aspx).
 - **Error Code** [Type = HexInt32]: there is no information about this field in this document.
 - **EAP Reason Code** [Type = HexInt32]: there is no information about this field in this document. See additional information here: [https://technet.microsoft.com/en-us/library/dd197570\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/dd197570(v=ws.10).aspx).
 - **EAP Root Cause String** [Type = UnicodeString]: there is no information about this field in this document.
 - **EAP Error Code** [Type = HexInt32]: there is no information about this field in this document.

Security Monitoring Recommendations:

- For 5632(S, F): A request was made to authenticate to a wireless network.
- There is no recommendation for this event in this document.

5633(S, F): A request was made to authenticate to a wired network.

Event Description:

This event generates when [802.1x](#) authentication attempt was made for wired network.
It typically generates when network adapter connects to new wired network.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5633</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12551</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-11T01:26:59.679232500Z" />
<EventRecordID>1198715</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2920" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="InterfaceName">Microsoft Hyper-V Network Adapter</Data>
<Data Name="Identity">-</Data>
<Data Name="SubjectUserName">-</Data>
<Data Name="SubjectDomainName">-</Data>
<Data Name="SubjectLogonId">0x0</Data>
<Data Name="ReasonCode">0x70003</Data>
<Data Name="ReasonText">The network does not support authentication</Data>
<Data Name="ErrorCode">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

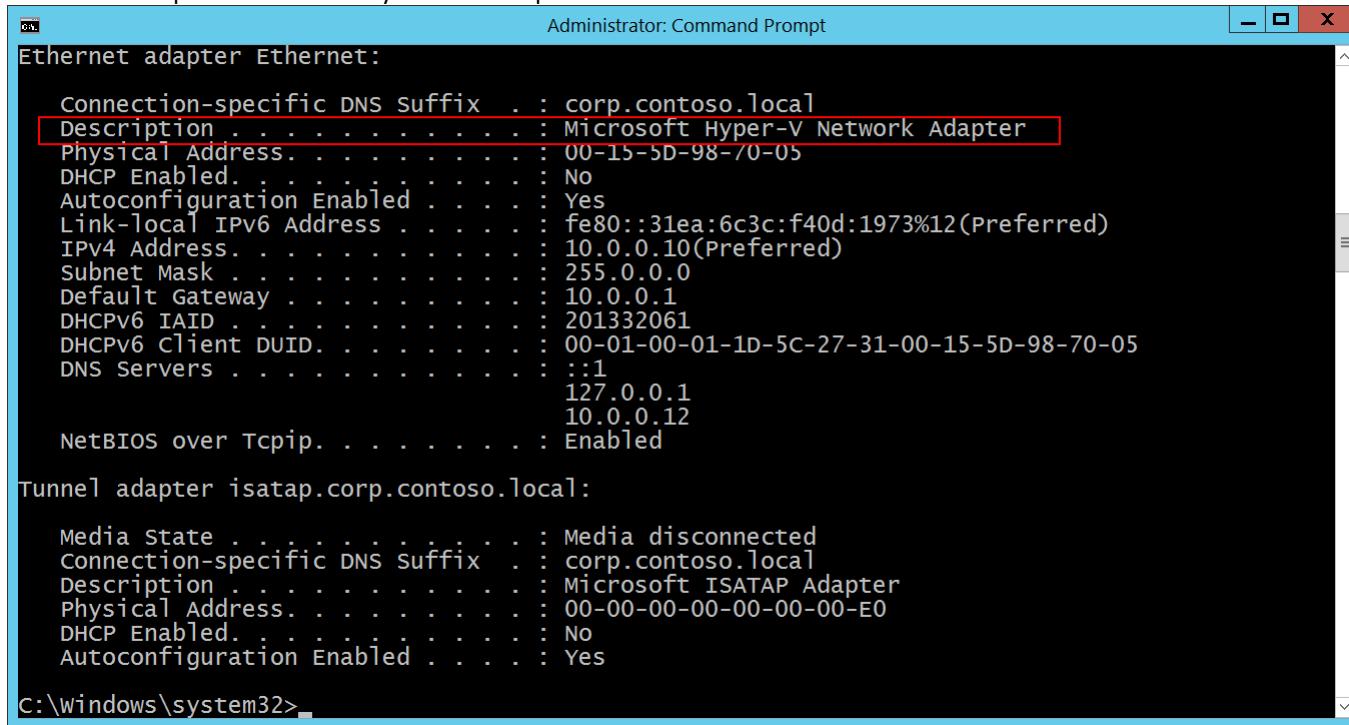
Field Descriptions:

Subject:

- **Security ID** [Type = UnicodeString]: User Principal Name (UPN) of account for which 802.1x authentication request was made.
User principal name (UPN) format is used to specify an Internet-style name, such as `UserName@Example.Microsoft.com`.
- **Account Name** [Type = UnicodeString]: the name of the account for which 802.1x authentication request was made.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Interface:

- **Name** [Type = UnicodeString]: the name (description) of network interface which was used for authentication request. You can get the list of all available network adapters using "ipconfig /all" command. See "Description" row for every network adapter:



```

Administrator: Command Prompt
Ethernet adapter Ethernet:

  Connection-specific DNS Suffix . : corp.contoso.local
  Description . . . . . : Microsoft Hyper-V Network Adapter
  Physical Address . . . . . : 00-15-5D-98-70-05
  DHCP Enabled. . . . . : No
  Autoconfiguration Enabled . . . . . : Yes
  Link-local IPv6 Address . . . . . : fe80::31ea:6c3c:f40d:1973%12(PREFERRED)
  IPv4 Address. . . . . : 10.0.0.10(Preferred)
  Subnet Mask . . . . . : 255.0.0.0
  Default Gateway . . . . . : 10.0.0.1
  DHCPv6 IAID . . . . . : 201332061
  DHCPv6 Client DUID. . . . . : 00-01-00-01-1D-5C-27-31-00-15-5D-98-70-05
  DNS Servers . . . . . :
    127.0.0.1
    10.0.0.12
  NetBIOS over Tcpip. . . . . : Enabled

Tunnel adapter isatap.corp.contoso.local:

  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . : corp.contoso.local
  Description . . . . . : Microsoft ISATAP Adapter
  Physical Address. . . . . : 00-00-00-00-00-00-E0
  DHCP Enabled. . . . . : No
  Autoconfiguration Enabled . . . . . : Yes

C:\windows\system32>

```

Additional Information:

- **Reason Code** [Type = UnicodeString]: contains Reason Text (explanation of Reason Code) and Reason Code for wired authentication results. See more information about reason codes for wired authentication here: [https://msdn.microsoft.com/en-us/library/windows/desktop/dd877212\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd877212(v=vs.85).aspx),
[https://technet.microsoft.com/en-us/library/cc727747\(v=ws.10\).aspx](https://technet.microsoft.com/en-us/library/cc727747(v=ws.10).aspx).
- **Error Code** [Type = HexInt32]: unique [EAP error code](#).

Security Monitoring Recommendations:

For 5633(S, F): A request was made to authenticate to a wired network.

- There is no recommendation for this event in this document.

Audit Special Logon

Audit Special Logon determines whether the operating system generates audit events under special sign on (or log on) circumstances.

This subcategory allows you to audit events generated by special logons such as the following:

- The use of a special logon, which is a logon that has administrator-equivalent privileges and can be used to elevate a process to a higher level.
- A logon by a member of a Special Group. Special Groups enable you to audit events generated when a member of a certain group has logged on to your network. You can configure a list of group security identifiers (SIDs) in the registry. If any of those SIDs are added to a token during logon and the subcategory is enabled, an event is logged.

Event volume:

- Low on a client computer.
- Medium on a domain controllers or network servers.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>This subcategory is very important because of Special Groups related events, you must enable this subcategory for Success audit if you use this feature.</p> <p>At the same time this subcategory allows you to track account logon sessions to which sensitive privileges were assigned.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	Yes	No	Yes	No	<p>This subcategory is very important because of Special Groups related events, you must enable this subcategory for Success audit if you use this feature.</p> <p>At the same time this subcategory allows you to track account logon sessions to which sensitive privileges were assigned.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	Yes	No	Yes	No	<p>This subcategory is very important because of Special Groups related events, you must enable this subcategory for Success audit if you use this feature.</p> <p>At the same time this subcategory allows you to track account logon sessions to which sensitive privileges were assigned.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4964\(S\)](#): Special groups have been assigned to a new logon.
- [4672\(S\)](#): Special privileges assigned to new logon.

4964(S): Special groups have been assigned to a new logon.

Event Properties - Event 4964, Microsoft Windows security auditing.

<input checked="" type="checkbox"/>	<input type="checkbox"/>	General Details
Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0xD972E Logon GUID: {00000000-0000-0000-000000000000}		
New Logon: Security ID: CONTOSO\ladmin Account Name: ladmin Account Domain: CONTOSO Logon ID: 0x139faf Logon GUID: {b03b6192-09ae-e77f-dd10-2dc430766040} Special Groups Assigned: CONTOSO\Domain Admins		
Log Name: Security Source: Microsoft Windows security Logged: 9/10/2015 7:25:16 PM Event ID: 4964 Task Category: Special Logon Level: Information Keywords: Audit Success User: N/A Computer: DC01.contoso.local OpCode: Info More Information : Event Log Online Help		

Event Description:
 This event occurs when an account that is a member of any defined [Special Group](#) logs in.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4964</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12548</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-11T02:25:16.236443300Z" />
  <EventRecordID>238923</EventRecordID>
  <Correlation />
  <Execution ProcessID="504" ThreadID="5008" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
```

- <EventData>

```

<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0xd972e</Data>
<Data Name="LogonGuid">{00000000-0000-0000-000000000000}</Data>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-500</Data>
<Data Name="TargetUserName">ladmin</Data>
<Data Name="TargetDomainName">CONTOSO</Data>
<Data Name="TargetLogonId">0x139faf</Data>
<Data Name="TargetLogonGuid">{B03B6192-09AE-E77F-DD10-2DC430766040}</Data>
<Data Name="SidList">%{S-1-5-21-3457937927-2839227994-823803824-512}</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Special Groups is a new feature in Windows Vista and in Windows Server 2008. The Special Groups feature lets the administrator find out when a member of a certain group logs on to the computer. The Special Groups feature lets an administrator set a list of group security identifiers (SIDs) in the registry.

To add Special Groups perform the following actions:

1. Open Registry Editor.
2. Locate and then click the following registry subkey: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Lsa\Audit
3. On the Edit menu, point to New, and then click String Value.
4. Type SpecialGroups, and then press ENTER.
5. Right-click SpecialGroups, and then click Modify.
6. In the Value data box, type the group SIDs, and then click OK.

A semicolon character (;) can be used to delimit the SID list. For example, you can use the following string that contains a semicolon to delimit two SIDs:
S-1-5-32-544;S-1-5-32-123-54-65

For more information see: <http://blogs.technet.com/b/askds/archive/2008/03/11/special-groups-auditing-via-group-policy-preferences.aspx>

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested logon for **New Logon** account. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested logon for **New Logon** account.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."
- **Logon GUID** [Type = GUID]: a GUID that can help you correlate this event with another event that can contain the same **Logon GUID**, "[4769](#)(S, F): A Kerberos service ticket was requested event on a domain controller.

It also can be used for correlation between a 4964 event and several other events (on the same computer) that can contain the same **Logon GUID**, "[4648](#)(S): A logon was attempted using explicit credentials" and "[4624](#)(S): An account was successfully logged on."

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

New Logon:

- **Security ID** [Type = SID]: SID of account that performed the logon. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Account Name** [Type = UnicodeString]: the name of the account that performed the logon.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”
- **Logon GUID** [Type = GUID]: a GUID that can help you correlate this event with another event that can contain the same **Logon GUID**, “[4769](#)(S, F): A Kerberos service ticket was requested event on a domain controller.

It also can be used for correlation between a 4964 event and several other events (on the same computer) that can contain the same **Logon GUID**, “[4648](#)(S): A logon was attempted using explicit credentials” and “[4624](#)(S): An account was successfully logged on.”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

- **Special Groups Assigned** [Type = UnicodeString]: the list of special group SIDs, which **New Logon\Security ID** is a member of.

Security Monitoring Recommendations:

For 4964(S): Special groups have been assigned to a new logon.

- Generally speaking, every [4964](#) event should be monitored, because the purpose of Special Groups is to define a list of critical or important groups (Domain Admins, Enterprise Admins, service account groups, and so on) and trigger an event every time a member of these groups logs on to a computer. For example, you can monitor for every Domain Administrators logon to a non-administrative workstation.

4672(S): Special privileges assigned to new logon.

Event Description:

This event generates for new account logons if any of the following sensitive privileges are assigned to the new logon session:

- SeTcbPrivilege - Act as part of the operating system
- SeBackupPrivilege - Back up files and directories
- SeCreateTokenPrivilege - Create a token object

- SeDebugPrivilege - Debug programs
- SeEnableDelegationPrivilege - Enable computer and user accounts to be trusted for delegation
- SeAuditPrivilege - Generate security audits
- SeImpersonatePrivilege - Impersonate a client after authentication
- SeLoadDriverPrivilege - Load and unload device drivers
- SeSecurityPrivilege - Manage auditing and security log
- SeSystemEnvironmentPrivilege - Modify firmware environment values
- SeAssignPrimaryTokenPrivilege - Replace a process-level token
- SeRestorePrivilege - Restore files and directories,
- SeTakeOwnershipPrivilege - Take ownership of files or other objects

You typically will see many of these events in the event log, because every logon of SYSTEM (Local System) account triggers this event.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4672</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12548</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-11T01:10:57.091809600Z" />
<EventRecordID>237692</EventRecordID>
<Correlation />
<Execution ProcessID="504" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x671101</Data>
```

<Data Name="PrivilegeList">SeTcbPrivilege SeSecurityPrivilege SeTakeOwnershipPrivilege SeLoadDriverPrivilege SeBackupPrivilege SeRestorePrivilege SeDebugPrivilege
SeSystemEnvironmentPrivilege SeEnableDelegationPrivilege SeImpersonatePrivilege</Data>

</EventData>
</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account to which special privileges were assigned. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account to which special privileges were assigned.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Privileges [Type = UnicodeString]: the list of sensitive privileges, assigned to the new logon. The following table contains the list of possible privileges for this event:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	Required to perform backup operations. With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system. This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list (ACL)</i> specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • READ_CONTROL

		<ul style="list-style-type: none"> • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.</p>
SeImpersonatePrivilege	Impersonate a client after authentication	<p>With this privilege, the user can impersonate other accounts.</p>
SeLoadDriverPrivilege	Load and unload device drivers	<p>Required to load or unload a device driver.</p> <p>With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.</p>
SeRestorePrivilege	Restore files and directories	<p>Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>

SeSecurityPrivilege	Manage auditing and security log	Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log. With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys. A user with this privilege can also view and clear the security log.
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeTakeOwnershipPrivilege	Take ownership of files or other objects	Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object. With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.
SeTcbPrivilege	Act as part of the operating system	This privilege identifies its holder as part of the trusted computer base. This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.

Security Monitoring Recommendations:

For 4672(S): Special privileges assigned to new logon.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Monitor for this event where “**Subject\Security ID**” is not one of these well-known security principals: LOCAL SYSTEM, NETWORK SERVICE, LOCAL SERVICE, and where “**Subject\Security ID**” is not an administrative account that is expected to have the listed **Privileges**.
- If you have a list of specific privileges which should never be granted, or granted only to a few accounts (for example, SeDebugPrivilege), use this event to monitor for those **Privileges**.
- If you are required to monitor any of the sensitive privileges in the [Event Description for this event](#), search for those specific privileges in the event.

Object Access

Audit Application Generated

Audit Application Generated generates events for actions related to Authorization Manager [applications](#).

Audit Application Generated subcategory is out of scope of this document, because [Authorization Manager](#) is very rarely in use and it is deprecated starting from Windows Server 2012.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF – if you use Authorization Manager in your environment and you need to monitor events related to Authorization Manager applications , enable this subcategory.
Member Server	IF	IF	IF	IF	IF – if you use Authorization Manager in your environment and you need to monitor events related to Authorization Manager applications , enable this subcategory.
Workstation	IF	IF	IF	IF	IF – if you use Authorization Manager in your environment and you need to monitor events related to Authorization Manager applications , enable this subcategory.

Events List:

- [4665](#): An attempt was made to create an application client context.
- [4666](#): An application attempted an operation.
- [4667](#): An application client context was deleted.
- [4668](#): An application was initialized.

4665: An attempt was made to create an application client context.

4666: An application attempted an operation.

4667: An application client context was deleted.

4668: An application was initialized.

Audit Certification Services

Audit Certification Services determines whether the operating system generates events when Active Directory Certificate Services (AD CS) operations are performed.

Examples of AD CS operations include:

- AD CS starts, shuts down, is backed up, or is restored.
- Certificate revocation list (CRL)-related tasks are performed.
- Certificates are requested, issued, or revoked.
- Certificate manager settings for AD CS are changed.
- The configuration and properties of the certification authority (CA) are changed.
- AD CS templates are modified.
- Certificates are imported.
- A CA certificate is published to Active Directory Domain Services.
- Security permissions for AD CS role services are modified.
- Keys are archived, imported, or retrieved.
- The OCSP Responder Service is started or stopped.

Monitoring these operational events is important to ensure that AD CS role services are functioning properly.

Event volume: Low to medium on servers that provide AD CS role services.

Role-specific subcategories are outside the scope of this document.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	IF – if a server has the Active Directory Certificate Services (AD CS) role installed and you need to monitor AD CS related events, enable this subcategory.
Member Server	IF	IF	IF	IF	IF – if a server has the Active Directory Certificate Services (AD CS) role installed and you need to monitor AD CS related events, enable this subcategory.
Workstation	No	No	No	No	Active Directory Certificate Services (AD CS) role cannot be installed on client OS.

- 4868: The certificate manager denied a pending certificate request.
- 4869: Certificate Services received a resubmitted certificate request.
- 4870: Certificate Services revoked a certificate.
- 4871: Certificate Services received a request to publish the certificate revocation list (CRL).
- 4872: Certificate Services published the certificate revocation list (CRL).
- 4873: A certificate request extension changed.
- 4874: One or more certificate request attributes changed.
- 4875: Certificate Services received a request to shut down.
- 4876: Certificate Services backup started.
- 4877: Certificate Services backup completed.
- 4878: Certificate Services restore started.
- 4879: Certificate Services restore completed.
- 4880: Certificate Services started.
- 4881: Certificate Services stopped.
- 4882: The security permissions for Certificate Services changed.
- 4883: Certificate Services retrieved an archived key.
- 4884: Certificate Services imported a certificate into its database.
- 4885: The audit filter for Certificate Services changed.
- 4886: Certificate Services received a certificate request.
- 4887: Certificate Services approved a certificate request and issued a certificate.
- 4888: Certificate Services denied a certificate request.

- 4889: Certificate Services set the status of a certificate request to pending.
- 4890: The certificate manager settings for Certificate Services changed.
- 4891: A configuration entry changed in Certificate Services.
- 4892: A property of Certificate Services changed.
- 4893: Certificate Services archived a key.
- 4894: Certificate Services imported and archived a key.
- 4895: Certificate Services published the CA certificate to Active Directory Domain Services.
- 4896: One or more rows have been deleted from the certificate database.
- 4897: Role separation enabled.
- 4898: Certificate Services loaded a template.

Audit Detailed File Share

Audit Detailed File Share allows you to audit attempts to access files and folders on a shared folder.

The Detailed File Share setting logs an event every time a file or folder is accessed, whereas the File Share setting only records one event for any connection established between a client and file share. Detailed File Share audit events include detailed information about the permissions or other criteria used to grant or deny access.

There are no system access control lists (SACLs) for shared folders. If this policy setting is enabled, access to all shared files and folders on the system is audited.

Event volume:

- High on file servers.
- High on domain controllers because of SYSVOL network access required by Group Policy.
- Low on member servers and workstations.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	Yes	No	Yes	<p>Audit Success for this subcategory on domain controllers typically will lead to very high volume of events, especially for SYSVOL share.</p> <p>We recommend monitoring Failure access attempts: the volume should not be very high. You will be able to see who was not able to get access to a file or folder on a network share on a computer.</p>
Member Server	IF	Yes	IF	Yes	<p>IF – If a server has shared network folders which typically get many access requests (File Server, for example), the volume of events might be very high. If you really need to track all successful access events for every file or folder located on a shared folder, enable Success auditing or use the Audit File System subcategory, although that subcategory excludes some information in Audit Detailed File Share, for example, the client's IP address.</p> <p>The volume of Failure events for member servers should not be very high (if they are not File Servers). With Failure auditing, you will be able to see who was not able to get access to a file or folder on a network share on this computer.</p>
Workstation	IF	Yes	IF	Yes	<p>IF – If a workstation has shared network folders which typically get many access requests, the volume of events might be very high. If you really need to track all successful access events for every file or folder located on a shared folder, enable Success auditing or use Audit File System subcategory, although that subcategory excludes some information in Audit Detailed File Share, for example, the client's IP address.</p> <p>The volume of Failure events for workstations should not be very high. With Failure auditing, you will be able to see who was not able to get access to a file or folder on a network share on this computer.</p>

Events List:

- [5145](#)(S, F): A network share object was checked to see whether client can be granted desired access.

5145(S, F): A network share object was checked to see whether client can be granted desired access.

Event Properties - Event 5145, Microsoft Windows security auditing.

General **Details**

A network share object was checked to see whether client can be granted desired access.

Account Domain:	CONTOSO
Logon ID:	0x38D34
Network Information:	
Object Type:	File
Source Address:	fe80::31ea:6c3cf40d:1973
Source Port:	56926
Share Information:	
Share Name:	\?\Documents
Share Path:	\??\C:\Documents
Relative Target Name:	Bginfo.exe
Access Request Information:	
Access Mask:	0x100081
Accesses:	SYNCHRONIZE ReadData (or ListDirectory) ReadAttributes
Access Check Results:	
SYNCHRONIZE:	Granted by D:(A;;FA;;;WD)
ReadData (or ListDirectory):	Granted by D:(A;;FA;;;WD)
ReadAttributes:	Granted by D:(A;;FA;;;WD)
Log Name:	Security
Source:	Microsoft Windows ser
Event ID:	5145
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

Event Description:

This event generates every time network share object (file or folder) was accessed.

Important: Failure events are generated only when access is denied at the file share level. No events are generated if access was denied on the file system (NTFS) level.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-
A5BA-3E3B0328C30D}" />
<EventID>5145</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12811</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-17T23:54:48.941761700Z" />
<EventRecordID>267092</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectLogonId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>

```

```

<Data Name="SubjectLogonId">0x38d34</Data>
<Data Name="ObjectType">File</Data>
<Data Name="IpAddress">fe80::31ea:6c3cf40d:1973</Data>
<Data Name="IpPort">56926</Data>
<Data Name="ShareName">\?\Documents</Data>
<Data Name="ShareLocalPath">\??\C:\Documents</Data>
<Data Name="RelativeTargetName">Bginfo.exe</Data>

```

```
<Data Name="AccessMask">0x100081</Data>
<Data Name="AccessList">%%1541 %%4416 %%4423</Data>
<Data Name="AccessReason">%%1541: %%1801 D:(A;;FA;;;WD) %%4416: %%1801 D:(A;;FA;;;WD) %%4423: %%1801 D:(A;;FA;;;WD)</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested access to network share object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested access to network share object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Network Information:

- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation. Always "File" for this event.

The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Source Address** [Type = UnicodeString]: source IP address from which access was performed.

- IPv6 address or ::ffff:IPv4 address of a client.
- ::1 or 127.0.0.1 means localhost.
- **Source Port** [Type = UnicodeString]: source TCP or UDP port which was used from remote or local machine to request the access.
 - 0 for local access attempts.

Share Information:

- **Share Name** [Type = UnicodeString]: the name of accessed network share. The format is: *\SHARE_NAME.
- **Share Path** [Type = UnicodeString]: the full system (NTFS) path for accessed share. The format is: \\??\PATH. Can be empty, for example for **Share Name**: *\IPC\$.
- **Relative Target Name** [Type = UnicodeString]: relative name of the accessed target file or folder. This file-path is relative to the network share. If access was requested for the share itself, then this field appears as "\".

Access Request Information:

- **Access Mask** [Type = HexInt32]: the sum of hexadecimal values of requested access rights. See "Table 13. File access codes." for different hexadecimal values for access rights.
- **Accesses** [Type = UnicodeString]: the list of access rights which were requested by **Subject\Security ID**. These access rights depend on **Object Type**.

Access	Hex Value	Schema Value	Description
ReadData (or ListDirectory)	0x1	%%4416	ReadData - For a file object, the right to read the corresponding file data. For a directory object, the right to read the corresponding directory data. ListDirectory - For a directory, the right to list the contents of the directory.
WriteData (or AddFile)	0x2	%%4417	WriteData - For a file object, the right to write data to the file. For a directory object, the right to create a file in the directory (FILE_ADD_FILE). AddFile - For a directory, the right to create a file in the directory.
AppendData (or AddSubdirectory or CreatePipeInstance)	0x4	%%4418	AppendData - For a file object, the right to append data to the file. (For local files, write operations will not overwrite existing data if this flag is specified without FILE_WRITE_DATA .) For a directory object, the right to create a subdirectory (FILE_ADD_SUBDIRECTORY). AddSubdirectory - For a directory, the right to create a subdirectory. CreatePipeInstance - For a named pipe, the right to create a pipe.
ReadEA	0x8	%%4419	The right to read extended file attributes.
WriteEA	0x10	%%4420	The right to write extended file attributes.
Execute/Traverse	0x20	%%4421	Execute - For a native code file, the right to execute the file. This access right given to scripts may cause the script to be executable, depending on the script interpreter. Traverse - For a directory, the right to traverse the directory. By default, users are assigned the BYPASS_TRAVERSE_CHECKING privilege, which ignores the FILE_TRAVERSE access right. See the remarks in File Security and Access Rights for more information.
DeleteChild	0x40	%%4422	For a directory, the right to delete a directory and all the files it contains, including read-only files.
ReadAttributes	0x80	%%4423	The right to read file attributes.
WriteAttributes	0x100	%%4424	The right to write file attributes.
DELETE	0x10000	%%1537	The right to delete the object.
READ_CONTROL	0x20000	%%1538	The right to read the information in the object's security descriptor, not including the information in the system access control list (SACL).

WRITE_DAC	0x40000	%%1539	The right to modify the discretionary access control list (DACL) in the object's security descriptor.
WRITE_OWNER	0x80000	%%1540	The right to change the owner in the object's security descriptor
SYNCHRONIZE	0x1000000	%%1541	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.
ACCESS_SYS_SEC	0x10000000	%%1542	The ACCESS_SYS_SEC access right controls the ability to get or set the SACL in an object's security descriptor.

Table 13. File access codes.

Access Check Results [Type = UnicodeString]: the list of access check results. The format of the result is:

REQUESTED_ACCESS: RESULT ACE_WICH_ALLOWED_OR_DENIED_ACCESS.

- REQUESTED_ACCESS – the name of requested access (see “Table 13. File access codes.”).
- RESULT:
 - Granted by – if access was granted.
 - Denied by – if access was denied.
- ACE_WICH_ALLOWED_OR_DENIED_ACCESS: the Security Descriptor Definition Language (SDDL) value for Access Control Entry (ACE), which granted or denied access.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

Q:BA|G:SYD:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)

- **Q:** = Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code

"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- G: = Primary Group.
- D: = DACL Entries.
- S: = SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: `D:(A;;FA;;;WD)`

- entry_type:
 - "D" - DACL
 - "S" - SACL
- inheritance_flags:
 - "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.
 - "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" Is not also set.
 - "AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.
- ace_type:
 - "A" - ACCESS ALLOWED
 - "D" - ACCESS DENIED
 - "OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).
 - "OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).
 - "AU" - SYSTEM AUDIT
 - "A" - SYSTEM ALARM
 - "OU" - OBJECT SYSTEM AUDIT
 - "OL" - OBJECT SYSTEM ALARM
- ace_flags:
 - "CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.
 - "OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.
 - "NP" - NO PROPAGATE: only immediate children inherit this ace.
 - "IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.
 - "ID" - ACE IS INHERITED
 - "SA" - SUCCESSFUL ACCESS AUDIT
 - "FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete

"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A
- inherit_object_guid: N/A
- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,
[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 5145(S, F): A network share object was checked to see whether client can be granted desired access.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Monitor this event if the **Network Information\Source Address** is not from your internal IP range.
- Monitor this event if the **Network Information\Source Address** should not be able to connect with the specific computer (**Computer:**).
- If you have critical files or folders on specific network shares, for which you need to monitor access attempts (Success and Failure), monitor for specific **Share Information\Share Name** and **Share Information\Relative Target Name**.
- If you have domain or local accounts that should only be able to access a specific list of shared files or folders, you can monitor for access attempts outside the allowed list.
- We recommend that you monitor for these **Access Request Information\Accesses** rights (especially for Failure):
 - WriteData (or AddFile)
 - AppendData (or AddSubdirectory or CreatePipeInstance)
 - WriteEA
 - DeleteChild
 - WriteAttributes
 - DELETE
 - WRITE_DAC

- WRITE_OWNER

Audit File Share

Audit File Share allows you to audit events related to file shares: creation, deletion, modification, and access attempts. Also, it shows failed SMB SPN checks.

There are no system access control lists (SACLs) for shares; therefore, after this setting is enabled, access to all shares on the system will be audited.

Combined with File System auditing, File Share auditing enables you to track what content was accessed, the source (IP address and port) of the request, and the user account that was used for the access.

Event volume:

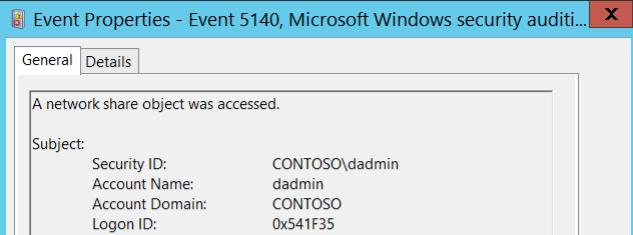
- High on file servers.
- High on domain controllers because of SYSVOL network access required by Group Policy.
- Low on member servers and workstations.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	Yes	Yes	Yes	Yes	We recommend Success auditing for domain controllers, because it's important to track deletion, creation, and modification events for network shares. We recommend Failure auditing to track failed SMB SPN checks and failed access attempts to network shares.
Member Server	Yes	Yes	Yes	Yes	We recommend Success auditing to track deletion, creation, modification, and access attempts to network share objects. We recommend Failure auditing to track failed SMB SPN checks and failed access attempts to network shares.
Workstation	Yes	Yes	Yes	Yes	We recommend Success auditing to track deletion, creation, modification and access attempts to network share objects. We recommend Failure auditing to track failed SMB SPN checks and failed access attempts to network shares.

Events List:

- [5140\(S, F\)](#): A network share object was accessed.
- [5142\(S\)](#): A network share object was added.
- [5143\(S\)](#): A network share object was modified.
- [5144\(S\)](#): A network share object was deleted.
- [5168\(F\)](#): SPN check for SMB/SMB2 failed.

 Event Properties - Event 5140, Microsoft Windows security audit... X

General Details

A network share object was accessed.

Subject:

Security ID:	CONTOSO\admind
Account Name:	admind
Account Domain:	CONTOSO
Logon ID:	0x541f35

Source Address: 10.0.0.100
Source Port: 49212

Share Information:

Share Name:	\\\"Documents
Share Path:	\??\C\Documents

Access Request Information:

Access Mask:	0x1
Accesses:	ReadData (or ListDirectory)

Log Name: Security
Source: Microsoft Windows security
Event ID: 5140
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy Close

5140(S, F): A network share object was accessed.

Event Description:

This event generates every time network share object was accessed.

This event generates once per session, when first access attempt was made.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5140</EventID>
<Version>1</Version>
<Level>0</Level>
<Task>12808</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T02:45:13.581231400Z" />
<EventRecordID>268495</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="772" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x541f35</Data>
  <Data Name="ObjectType">File</Data>
  <Data Name="IpAddress">10.0.0.100</Data>
  <Data Name="IpPort">49212</Data>
  <Data Name="ShareName">\\*\Documents</Data>
  <Data Name="ShareLocalPath">\\?\C:\Documents</Data>
  <Data Name="AccessMask">0x1</Data>
  <Data Name="AccessList">%%4416</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested access to network share object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested access to network share object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624: An account was successfully logged on.](#)”

Network Information:

- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation. Always “File” for this event.

The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Source Address** [Type = UnicodeString]: source IP address from which access was performed.
 - IPv6 address or ::ffff:IPv4 address of a client.
 - ::1 or 127.0.0.1 means localhost.
- **Source Port** [Type = UnicodeString]: source TCP or UDP port which was used from remote or local machine to request the access.
 - 0 for local access attempts.

Share Information:

- **Share Name** [Type = UnicodeString]: the name of accessed network share. The format is: *\SHARE_NAME.
- **Share Path** [Type = UnicodeString]: the full system (NTFS) path for accessed share. The format is: \\?\PATH. Can be empty, for example for **Share Name**: *\IPC\$.

Access Request Information:

- **Access Mask** [Type = HexInt32]: the sum of hexadecimal values of requested access rights. See “Table 13. File access codes.” for different hexadecimal values for access rights. Has always “0x1” value for this event.
- **Accesses** [Type = UnicodeString]: the list of access rights which were requested by **Subject\Security ID**. These access rights depend on **Object Type**. Has always “**ReadData (or ListDirectory)**” value for this event.

Security Monitoring Recommendations:

For 5140(S, F): A network share object was accessed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have high-value computers for which you need to monitor all access to all shares or specific shares (“**Share Name**”), monitor this event. For example, you could monitor share C\$ on domain controllers.
- Monitor this event if the **Network Information\Source Address** is not from your internal IP range.
- Monitor this event if the **Network Information\Source Address** should not be able to connect with the specific computer (**Computer**):.
- If you need to monitor access attempts to local shares from a specific IP address (“**Network Information\Source Address**”), use this event.
- If you need to monitor for specific Access Types (for example, ReadData or WriteData), for all or specific shares (“**Share Name**”), monitor this event for the “**Access Type**.”

5142(S): A network share object was added.

Event Properties - Event 5142, Microsoft Windows security auditi... X

General Details

Subject:	Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x38D12
Share Information:	Share Name: *\Documents Share Path: C:\Documents
Log Name:	Security
Source:	Microsoft Windows se
Event ID:	5142
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time network share object was added.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5142</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12808</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T02:27:01.206646900Z" />
<EventRecordID>268462</EventRecordID>
<Correlation />
```

```

<Execution ProcessID="4" ThreadID="4304" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38d12</Data>
<Data Name="ShareName">\\*\Documents</Data>
<Data Name="ShareLocalPath">C:\Documents</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

Subject:

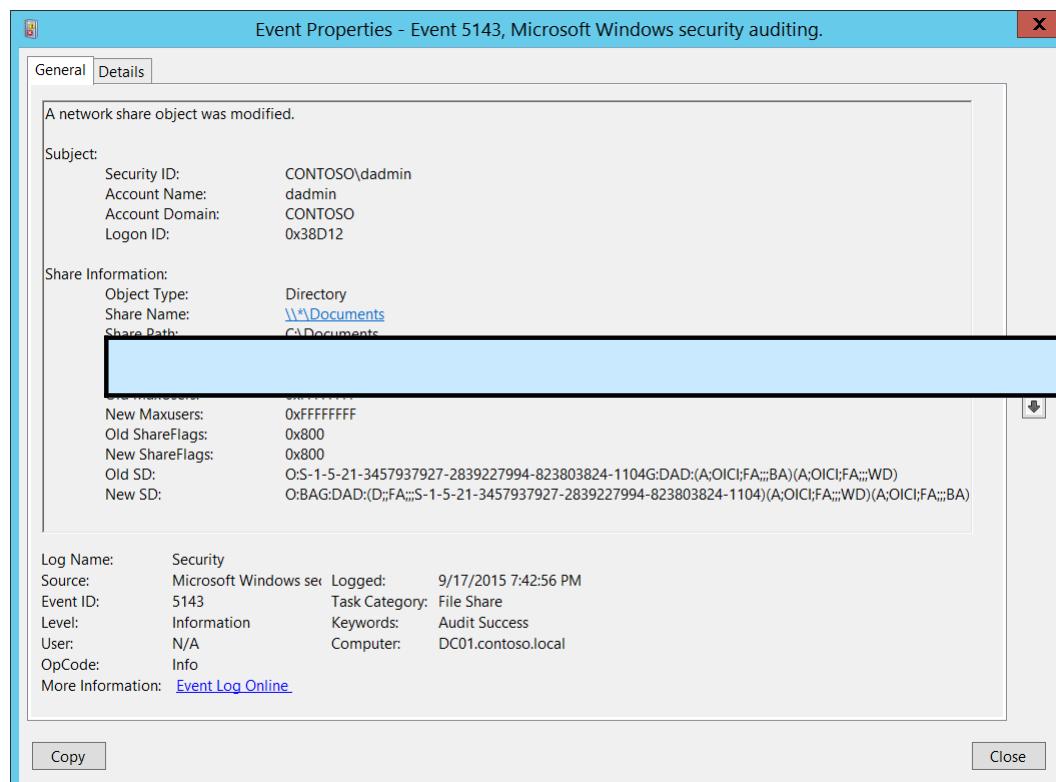
- **Security ID** [Type = SID]: SID of account that requested the “add network share object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “add network share object” operation.

- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:

- Domain NETBIOS name example: CONTOSO
- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.



- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, [4624](#): An account was successfully logged on.”

Share Information:

- **Share Name** [Type = UnicodeString]: the name of the added share object. The format is: *\SHARE_NAME.
- **Share Path** [Type = UnicodeString]: the full system (NTFS) path for the added share object. The format is: \\??\PATH.

Security Monitoring Recommendations:

For 5142(S): A network share object was added.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have high-value computers for which you need to monitor creation of new file shares, monitor this event. For example, you could monitor domain controllers.
- We recommend checking “**Share Path**”, because it should not point to system directories, such as **C:\Windows** or **C:**, or to critical local folders which contain private or high value information.

5143(S): A network share object was modified.

Event Description:

This event generates every time network share object was modified.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5143</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12808</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T02:42:56.743298600Z" />
<EventRecordID>268483</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38d12</Data>
<Data Name="ObjectType">Directory</Data>
<Data Name="ShareName">\*\|Documents</Data>
<Data Name="ShareLocalPath">C:\Documents</Data>
<Data Name="OldRemark">N/A</Data>
<Data Name="NewRemark">N/A</Data>
<Data Name="OldMaxUsers">0xffffffff</Data>
<Data Name="NewMaxUsers">0xffffffff</Data>
<Data Name="OldShareFlags">0x800</Data>
<Data Name="NewShareFlags">0x800</Data>
<Data Name="OldSD">O:S-1-5-21-3457937927-2839227994-823803824-1104G:DAD:(A;OICI;FA;;;BA)(A;OICI;FA;;;WD)</Data>
<Data Name="NewSD">O:BAG:DAD:(D;;FA;;;S-1-5-21-3457937927-2839227994-823803824-1104)(A;OICI;FA;;;WD)(A;OICI;FA;;;BA)</Data>
</EventData>
```

</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “modify network share object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “modify network share object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

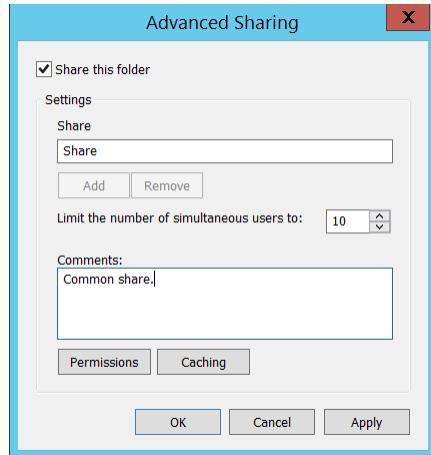
Share Information:

- **Object Type** [Type = UnicodeString]: The type of an object that was modified. Always “**Directory**” for this event.

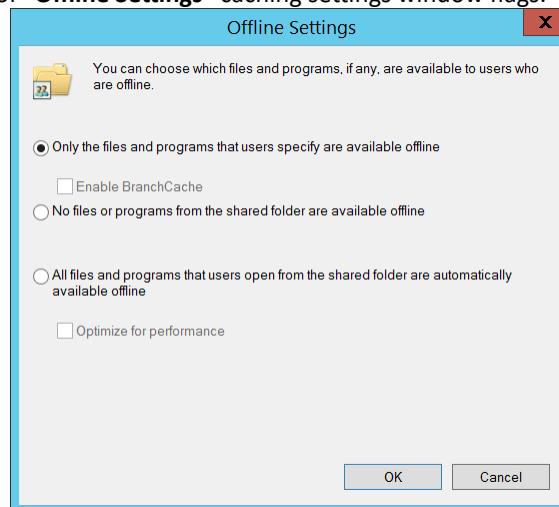
The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Share Name** [Type = UnicodeString]: the name of the modified share object. The format is: *\SHARE_NAME
- **Share Path** [Type = UnicodeString]: the full system (NTFS) path for the added share object. The format is: \\??\PATH. Can be empty, for example for **Share Name**: *\IPC\$.



- **Old Remark** [Type = UnicodeString]: the old value of network share “**Comments:**” field. Has “**N/A**” value if it is not set.
- **New Remark** [Type = UnicodeString]: the new value of network share “**Comments:**” field. Has “**N/A**” value if it is not set.
- **Old MaxUsers** [Type = HexInt32]: old hexadecimal value of “**Limit the number of simultaneous user to:**” field. Has “**0xFFFFFFFF**” value if the number of connections is unlimited.
- **New Maxusers** [Type = HexInt32]: new hexadecimal value of “**Limit the number of simultaneous user to:**” field. Has “**0xFFFFFFFF**” value if the number of connections is unlimited.
- **Old ShareFlags** [Type = HexInt32]: old hexadecimal value of “**Offline Settings**” caching settings window flags.



- **New ShareFlags** [Type = HexInt32]: new hexadecimal value of “**Offline Settings**” caching settings window flags.
- **Old SD** [Type = UnicodeString]: the old Security Descriptor Definition Language (SDDL) value for network share security descriptor.

- **New SD** [Type = UnicodeString]: the new Security Descriptor Definition Language (SDDL) value for network share security descriptor.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

Q:BAG:SYD:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)

- **Q**: = Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- **G**: = Primary Group.
- **D**: = DACL Entries.
- **S**: = SACL Entries.

DACL/SACL entry format: **entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)**

Example: **D:(A;;FA;;;WD)**

- **entry_type**:
- "D" - DACL
- "S" - SACL
- **inheritance_flags**:

"P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.

"AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" is not also set.

"AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.

- ace_type:

"A" - ACCESS ALLOWED

"D" - ACCESS DENIED

"OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).

"OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).

"AU" - SYSTEM AUDIT

"A" - SYSTEM ALARM

"OU" - OBJECT SYSTEM AUDIT

"OL" - OBJECT SYSTEM ALARM

- ace_flags:

"CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.

"OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.

"NP" - NO PROPAGATE: only immediate children inherit this ace.

"IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.

"ID" - ACE IS INHERITED

"SA" - SUCCESSFUL ACCESS AUDIT

"FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights

"KX" KEY EXECUTE

- object_guid: N/A
- inherit_object_guid: N/A
- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>, [https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

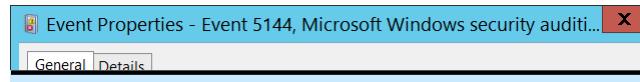
Security Monitoring Recommendations:

For 5143(S): A network share object was modified.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have high-value computers for which you need to monitor all modifications to all shares or specific shares ("Share Name"), monitor this event. For example, you could monitor all changes to the SYSVOL share on domain controllers.

5144(S): A network share object was deleted.

 Event Properties - Event 5144, Microsoft Windows security audit... X

General Details

Event Description:
This event generates every time a network share object is deleted.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

<p>Subject: Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x38D12</p> <p>Share Information: Share Name: <a \\documents"="" href="\\\">\\\"\\Documents Share Path: C:\Documents</p> <p>Log Name: Security Source: Microsoft Windows security Event ID: 5144 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p>	<p>Event XML:</p> <pre><Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>5144</EventID> <Version>0</Version> <Level>0</Level> <Task>12808</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-09-18T02:17:14.820551800Z" /> <EventRecordID>268368</EventRecordID> <Correlation /> <Execution ProcessID="4" ThreadID="4656" /></pre>
--	--

Copy Close

<Channel>Security</Channel>

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId" S-1-5-21-3457937927-2839227994-823803824-1104></Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38d12</Data>
<Data Name="ShareName">\*\Documents</Data>
<Data Name="ShareLocalPath">C:\Documents</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete network share object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete network share object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Share Information:

- **Share Name** [Type = UnicodeString]: the name of the deleted share object. The format is: *\SHARE_NAME
- **Share Path** [Type = UnicodeString]: the full system (NTFS) path for the deleted share object. The format is: \?\PATH.

Security Monitoring Recommendations:

For 5144(S): A network share object was deleted.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have critical network shares for which you need to monitor all changes (especially, the deletion of that share), monitor for specific “**Share Information\Share Name**”.
- If you have high-value computers for which you need to monitor all changes (especially, deletion of file shares), monitor for all [5144](#) events on these computers. For example, you could monitor file shares on domain controllers.

5168(F): SPN check for SMB/SMB2 failed.

Event Properties - Event 5168, Microsoft Windows security auditing.

General Details

Spn check for SMB/SMB2 fails.
Subject: Security ID: CONTOSO\dadmin

Logon ID: 0x000CD4

SPN:
SPN Name: N/A
Error Code: 0xC0000022

Server Information:
Server Names: CONTOSO;contoso.local;DC01.contoso.local;DC01;localhost;
Configured Names: N/A
IP Addresses: 127.0.0.1;10.0.0.10;fe80::31ea:6c3cf40d:1973;fe80::5efe:10.0.0.10;

Log Name: Security
Source: Microsoft Windows security
Event ID: 5168
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates when SMB SPN check fails.

It often happens because of NTLMv1 or LM protocols usage from client side when “[Microsoft Network Server: Server SPN target name validation level](#)” group policy set to “Require from client” on server side. SPN only sent to server when NTLMv2 or Kerberos protocols are used, and after that SPN can be validated.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5168</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12808</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T17:53:40.294859800Z" />
<EventRecordID>268946</EventRecordID>

```

```

<Correlation />
<Execution ProcessID="4" ThreadID="80" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserID">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>

```

```
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0xd0cd4</Data>
<Data Name="SpnName">N/A</Data>
<Data Name="ErrorCode">0xc0000022</Data>
<Data Name="ServerNames">CONTOSO;contoso.local;DC01.contoso.local;DC01;LocalHost;</Data>
<Data Name="ConfiguredNames">N/A</Data>
<Data Name="IpAddresses">127.0.0.1::1;10.0.0.10;fe80::31ea:6c3c:f40d:1973;fe80::5efe:10.0.0.10;</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account for which SPN check operation was failed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account for which SPN check operation was failed.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

SPN:

- **SPN Name** [Type = UnicodeString]: SPN which was used to access the server. If SPN was not provided, then the value will be "N/A".

Service Principal Name (SPN) is the name by which a client uniquely identifies an instance of a service. If you install multiple instances of a service on computers throughout a forest, each instance must have its own SPN. A given service instance can have multiple SPNs if there are multiple names that clients might use for authentication. For example, an SPN always includes the name of the host computer on which the service instance is running, so a service instance might register an SPN for each name or alias of its host.

- **Error Code** [Type = HexInt32]: hexadecimal error code, for example “0xC0000022” = STATUS_ACCESS_DENIED. You can find description for all SMB error codes here: <https://msdn.microsoft.com/en-us/library/ee441884.aspx>.

Server Information:

- **Server Names** [Type = UnicodeString]: information about possible server names to use to access the target server (NETBIOS, DNS, localhost, etc.).
- **Configured Names** [Type = UnicodeString]: information about the names which were provided for validation. If no information was provided the value will be “N/A”.
- **IP Addresses** [Type = UnicodeString]: information about possible IP addresses to use to access the target server (IPv4, IPv6).

Security Monitoring Recommendations:

For 5168(F): SPN check for SMB/SMB2 failed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. We recommend monitoring for any [5168](#) event, because it can be a sign of a configuration issue or a malicious authentication attempt.

Audit File System

Audit File System determines whether the operating system generates audit events when users attempt to access file system objects.

Audit events are generated only for objects that have configured system access control lists ([SACLs](#)), and only if the type of access requested (such as Write, Read, or Modify) and the account making the request match the settings in the [SACL](#).

If success auditing is enabled, an audit entry is generated each time any account successfully accesses a file system object that has a matching SACL. If failure auditing is enabled, an audit entry is generated each time any user unsuccessfully attempts to access a file system object that has a matching SACL.

These events are essential for tracking activity for file objects that are sensitive or valuable and require extra monitoring.

Event volume: Varies, depending on how file system [SACLs](#) are configured.

No audit events are generated for the default file system [SACLs](#).

This subcategory allows you to audit user attempts to access file system objects, file system object deletion and permissions change operations and hard link creation actions.

Only one event, “[4658](#): The handle to an object was closed,” depends on the [Audit Handle Manipulation](#) subcategory (Success auditing must be enabled). All other events generate without any additional configuration.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	We strongly recommend that you develop a File System Security Monitoring policy and define appropriate SACLs for file system objects for different operating system templates and roles. Do not enable this subcategory if you have not planned how to use and analyze the collected information. It is also important to delete non-effective, excess SACLs . Otherwise the auditing log will be overloaded with useless information.
Member Server	IF	IF	IF	IF	Failure events can show you unsuccessful attempts to access specific file system objects. Consider enabling this subcategory for critical computers first, after you develop a File System Security Monitoring policy for them.
Workstation	IF	IF	IF	IF	

Events List:

- [4656](#)(S, F): A handle to an object was requested.
- [4658](#)(S): The handle to an object was closed.
- [4660](#)(S): An object was deleted.
- [4663](#)(S): An attempt was made to access an object.
- [4664](#)(S): An attempt was made to create a hard link.
- [4985](#)(S): The state of a transaction has changed.
- [5051](#)(-): A file was virtualized.
- [4670](#)(S): Permissions on an object were changed.

4656(S, F): A handle to an object was requested.

Event Properties - Event 4656, Microsoft Windows security auditing.

General **Details**

A handle to an object was requested.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x4367B

Object:

Object Server:	Security
Object Type:	File
Object Name:	C:\Documents\HBI Data.txt
Handle ID:	0x0

Access Request Information:

Process ID:	0x1071
Process Name:	C:\Windows\System32\notepad.exe
Transaction ID:	(00000000-0000-0000-000000000000)
Accesses:	READ_CONTROL SYNCHRONIZE ReadData (or ListDirectory) WriteData (or AddFile) AppendData (or AddSubdirectory or CreatePipeInstance) ReadEA WriteEA ReadAttributes WriteAttributes
Access Reasons:	READ_CONTROL: Granted by Ownership SYNCHRONIZE: Unknown or unchecked ReadData (or ListDirectory): Unknown or unchecked WriteData (or AddFile): Unknown or unchecked AppendData (or AddSubdirectory or CreatePipeInstance): Denied by D:(D;;LC;;S-1-5-21-3457937927-2839227994-823803824-1104) ReadEA: Unknown or unchecked WriteEA: Unknown or unchecked ReadAttributes: Granted by ACE on parent folder D:(A;OICI;FA;;S-1-5-21-3457937927-2839227994-823803824-1104) WriteAttributes: Unknown or unchecked
Access Mask:	0x12019F
Privileges Used for Access Check:	-
Restricted SID Count:	0

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4656
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

<Data Name="SubjectLogonId">0x4367b</Data>

Event Description:

This event indicates that specific access was requested for an object. The object could be a file system, kernel, or registry object, or a file system object on removable storage or a device.

If access was declined, a Failure event is generated.

This event generates only if the object's [SACL](#) has the required ACE to handle the use of specific access rights.

This event shows that access was requested, and the results of the request, but it doesn't show that the operation was performed. To see that the operation was performed, check ["4663\(S\): An attempt was made to access an object."](#)

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-
A5BA-3E3B0328C30D}" />
  <EventID>4656</EventID>
  <Version>1</Version>
  <Level>0</Level>
  <Task>12800</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8010000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-18T22:15:19.346776600Z" />
  <EventRecordID>274057</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="524" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>

```

```
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Documents\HBI Data.txt</Data>
<Data Name="HandleId">0x0</Data>
<Data Name="TransactionId">{00000000-0000-0000-000000000000}</Data>
<Data Name="AccessList">%%1538 %%1541 %%4416 %%4417 %%4418 %%4419 %%4420 %%4423 %%4424</Data>
<Data Name="AccessReason">%%1538: %%1804 %%1541: %%1809 %%4416: %%1809 %%4417: %%1809 %%4418: %%1802 D:(D;;LC;;S-1-5-21-3457937927-2839227994-823803824-1104) %%4419: %%1809 %%4420: %%1809 %%4423: %%1811 D:(A;OICI;FA;;;S-1-5-21-3457937927-2839227994-823803824-1104) %%4424: %%1809</Data>
<Data Name="AccessMask">0x12019f</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="RestrictedSidCount">0</Data>
<Data Name="ProcessId">0x1074</Data>
<Data Name="ProcessName">C:\Windows\System32\notepad.exe</Data>
<Data Name="ResourceAttributes">S:AI(RA;ID;;;;WD;("Impact_MS",TI,0x10020,3000))</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.
- 1 - Windows Server 2012, Windows 8.
 - Added “Resource Attributes” field.
 - Added “Access Reasons” field.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested a handle to an object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested a handle to an object.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.

- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString]: has "**Security**" value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation.

The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: name and other identifying information for the object for which access was requested. For example, for a file, the path would be included.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, "[4663\(S\)](#): An attempt was made to access an object." This parameter might not be captured in the event, and in that case appears as "0x0".
- **Resource Attributes** [Type = UnicodeString] [Version 1]: attributes associated with the object. For some objects, the field does not apply and "-" is displayed.

For example, for a file, the following might be displayed: S:AI(RA;ID;;;;WD;("Impact_MS",Tl,0x10020,3000))

- Impact_MS: Resource Property ID.
- 3000: Recourse Property Value.

Impact

General

A resource property describes a characteristic of a resource, such as a file or a folder. It is used to define target resources and permissions when authoring central access rules. It is also used to classify resources.

Display name: **Impact**

Value type: **Ordered List**

Description:
The Impact property specifies the degree of organizational impact from inappropriate access or loss of the resource.

ID: **Impact_MS**

Is used for authorization
 Protect from accidental deletion

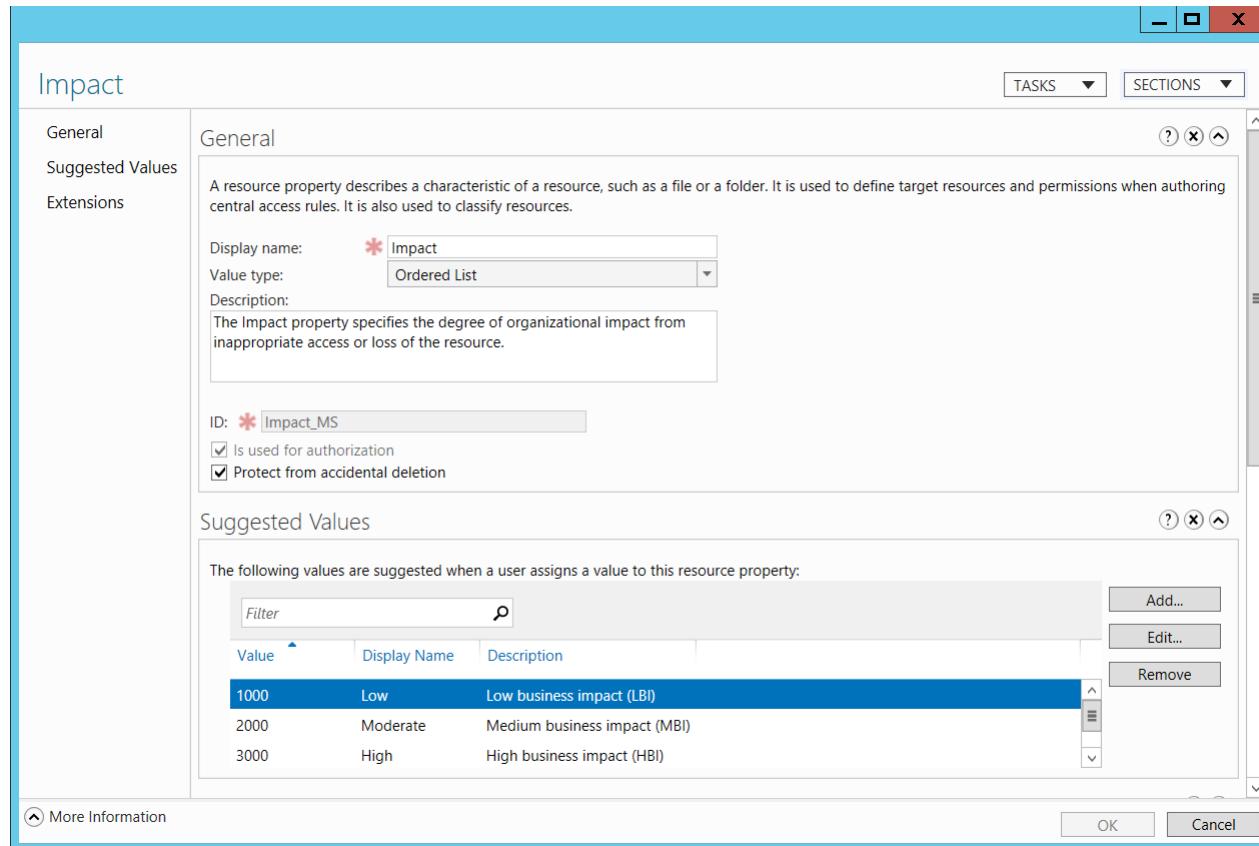
Suggested Values

The following values are suggested when a user assigns a value to this resource property:

Value	Display Name	Description
1000	Low	Low business impact (LBI)
2000	Moderate	Medium business impact (MBI)
3000	High	High business impact (HBI)

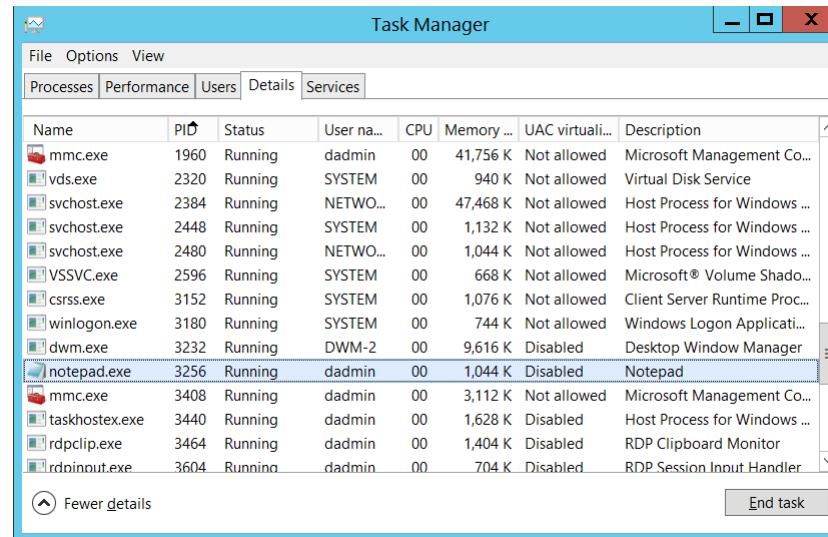
[More Information](#)

OK **Cancel**



Process Information:

- **Process ID [Type = Pointer]:** hexadecimal Process ID of the process through which the access was requested. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Access Request Information:

- **Transaction ID** [Type = GUID]: unique GUID of the transaction. This field can help you correlate this event with other events that might contain the same **Transaction ID**, such as “[4660\(S\): An object was deleted](#).”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Accesses** [Type = UnicodeString]: the list of access rights which were requested by **Subject\Security ID**. These access rights depend on **Object Type**. The following table contains information about the most common access rights for file system objects. Access rights for registry objects are often similar to file system objects, but the table contains a few notes about how they vary.

Access	Hex Value	Schema Value	Description
ReadData (or ListDirectory) (For registry objects, this is “Query key value.”)	0x1	%%4416	ReadData - For a file object, the right to read the corresponding file data. For a directory object, the right to read the corresponding directory data. ListDirectory - For a directory, the right to list the contents of the directory.
WriteData (or AddFile) (For registry objects,	0x2	%%4417	WriteData - For a file object, the right to write data to the file. For a directory object, the right to create a file in the directory (FILE_ADD_FILE). AddFile - For a directory, the right to create a file in the directory.

this is "Set key value.")			
AppendData (or AddSubdirectory or CreatePipeInstance)	0x4	%%4418	AppendData - For a file object, the right to append data to the file. (For local files, write operations will not overwrite existing data if this flag is specified without FILE_WRITE_DATA .) For a directory object, the right to create a subdirectory (FILE_ADD_SUBDIRECTORY). AddSubdirectory - For a directory, the right to create a subdirectory. CreatePipeInstance - For a named pipe, the right to create a pipe.
ReadEA (For registry objects, this is "Enumerate sub-keys.")	0x8	%%4419	The right to read extended file attributes.
WriteEA	0x10	%%4420	The right to write extended file attributes.
Execute/Traverse	0x20	%%4421	Execute - For a native code file, the right to execute the file. This access right given to scripts may cause the script to be executable, depending on the script interpreter. Traverse - For a directory, the right to traverse the directory. By default, users are assigned the BYPASS_TRAVERSE_CHECKING privilege, which ignores the FILE_TRAVERSE access right. See the remarks in File Security and Access Rights for more information.
DeleteChild	0x40	%%4422	For a directory, the right to delete a directory and all the files it contains, including read-only files.
ReadAttributes	0x80	%%4423	The right to read file attributes.
WriteAttributes	0x100	%%4424	The right to write file attributes.
DELETE	0x10000	%%1537	The right to delete the object.
READ_CONTROL	0x20000	%%1538	The right to read the information in the object's security descriptor, not including the information in the system access control list (SACL).
WRITE_DAC	0x40000	%%1539	The right to modify the discretionary access control list (DACL) in the object's security descriptor.
WRITE_OWNER	0x80000	%%1540	The right to change the owner in the object's security descriptor
SYNCHRONIZE	0x100000	%%1541	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.
ACCESS_SYS_SEC	0x1000000	%%1542	The ACCESS_SYS_SEC access right controls the ability to get or set the SACL in an object's security descriptor.

Table 14. File System objects access rights.

- **Access Reasons** [Type = UnicodeString] [Version 1]: the list of access check results. The format of this varies, depending on the object. For kernel objects, this field does not apply.
- **Access Mask** [Type = HexInt32]: hexadecimal mask for the requested or performed operation. For more information, see the preceding table.
- **Privileges Used for Access Check** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in the table below:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process.

		With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	<p>Required to perform backup operations.</p> <p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer</p>

	delegation	<p>object.</p> <p>The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.</p>
SeImpersonatePrivilege	Impersonate a client after authentication	<p>With this privilege, the user can impersonate other accounts.</p>
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	<p>Required to increase the base priority of a process.</p> <p>With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.</p>
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	<p>Required to increase the quota assigned to a process.</p> <p>With this privilege, the user can change the maximum memory that can be consumed by a process.</p>
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	<p>Required to load or unload a device driver.</p> <p>With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.</p>
SeLockMemoryPrivilege	Lock pages in memory	<p>Required to lock physical pages in memory.</p> <p>With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).</p>
SeMachineAccountPrivilege	Add workstations to domain	<p>With this privilege, the user can create a computer account.</p> <p>This privilege is valid only on domain controllers.</p>
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	<p>Required to gather profiling information for a single process.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.</p>
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	<p>Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER

		<ul style="list-style-type: none"> • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	<p>This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers.</p> <p>With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.</p>
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	<p>Required to gather profiling information for the entire system.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.</p>
SeSystemtimePrivilege	Change the system time	<p>Required to modify the system time.</p> <p>With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.</p>
SeTakeOwnershipPrivilege	Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>
SeTcbPrivilege	Act as part of the operating system	<p>This privilege identifies its holder as part of the trusted computer base.</p> <p>This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.</p>
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.

SeTrustedCredManAccessPrivilege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

- **Restricted SID Count** [Type = UInt32]: Number of [restricted SIDs](#) in the token. Applicable to only specific **Object Types**.

Security Monitoring Recommendations:

For 4656(S, F): A handle to an object was requested.

For kernel objects, this event and other auditing events have little to no security relevance and are hard to parse or analyze. There is no recommendation for auditing them, unless you know exactly what you need to monitor at the Kernel objects level.

For other types of objects, the following recommendations apply.

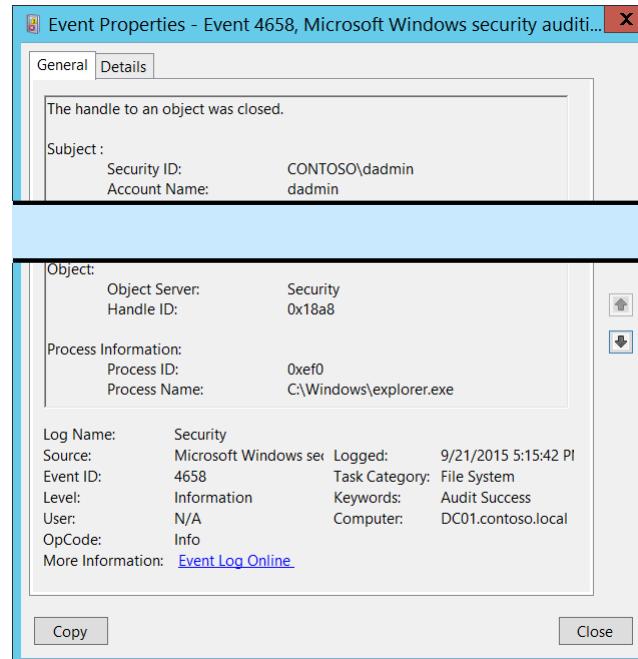
[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If **Object Name** is a sensitive or critical object for which you need to monitor any access attempt, monitor all [4656](#) events.
- If **Object Name** is a sensitive or critical object for which you need to monitor specific access attempts (for example, only write actions), monitor for all [4656](#) events with the corresponding **Access Request Information\Accesses** values.
- If you need to monitor files and folders with specific Resource Attribute values, monitor for all [4656](#) events with specific **Resource Attributes** field values.

For file system objects, we recommend that you monitor these **Access Request Information\Accesses** rights (especially for Failure events):

- WriteData (or AddFile)
- AppendData (or AddSubdirectory or CreatePipeInstance)
- WriteEA
- DeleteChild
- WriteAttributes
- DELETE
- WRITE_DAC
- WRITE_OWNER

4658(S): The handle to an object was closed.

 Event Properties - Event 4658, Microsoft Windows security audit...

The handle to an object was closed.

Subject:
Security ID: CONTOSO\dadmin
Account Name: dadmin

Object:
Object Server: Security
Handle ID: 0x18a8

Process Information:
Process ID: 0xef0
Process Name: C:\Windows\explorer.exe

Log Name: Security
Source: Microsoft Windows security
Event ID: 4658
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates when the handle to an object is closed. The object could be a file system, kernel, or registry object, or a file system object on removable storage or a device.

This event generates only if Success auditing is enabled for [Audit Handle Manipulation](#) subcategory.

Typically this event is needed if you need to know how long the handle to the object was open. Otherwise, it might not have any security relevance.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4658</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12800</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T00:15:42.910428100Z" />
<EventRecordID>276724</EventRecordID>
```

```
<Correlation />
<Execution ProcessID="4" ThreadID="5056" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x4367b</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="HandleId">0x18a8</Data>
<Data Name="ProcessId">0xef0</Data>
<Data Name="ProcessName">C:\Windows\explorer.exe</Data>
</EventData>
```

</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “close object’s handle” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

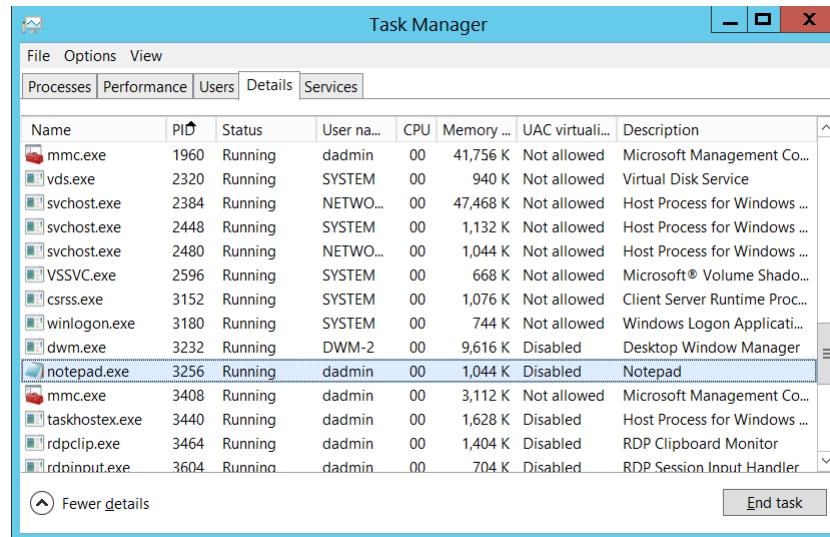
- **Account Name** [Type = UnicodeString]: the name of the account that requested the “close object’s handle” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “**Security**” value for this event.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that requested that the handle be closed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

Event Properties - Event 4660, Microsoft Windows security audit... X

General Details

An object was deleted.

Subject:

Security ID: CONTOSO\dadmin

Object:

Object Server: Security
Handle ID: 0x1678

Process Information:

Process ID: 0xef0
Process Name: C:\Windows\explorer.exe
Transaction ID: {00000000-0000-0000-0000-000000000000}

Log Name: Security
Source: Microsoft Windows sec... Logged: 9/18/2015 2:05:28 PM
Event ID: 4660 Task Category: File System
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy Close

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Security Monitoring Recommendations:

For 4658(S): The handle to an object was closed.

Appendix A: Security monitoring recommendations for many audit events

- **Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.** Typically this event has little to no security relevance and is hard to parse or analyze. There is no recommendation for this event, unless you know exactly what you need to monitor with it.
- This event can be used to track all actions or operations related to a specific object handle.
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.

4660(S): An object was deleted.

Event Description:

This event generates when an object was deleted. The object could be a file system, kernel, or registry object.

This event generates only if “Delete” auditing is set in object’s [SACL](#).

This event doesn't contain the name of the deleted object (only the **Handle ID**). It is better to use “[4663\(S\): An attempt was made to access an object](#)” with DELETE access to track object deletion.

The advantage of this event is that it's generated only during real delete operations. In contrast, “4663(S): An attempt was made to access an object” also generates during other actions, such as object renaming.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4660</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12800</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-18T21:05:28.677152100Z" />
  <EventRecordID>270188</EventRecordID>
  <Correlation />
  <Execution ProcessID="4" ThreadID="3060" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x4367b</Data>
  <Data Name="ObjectServer">Security</Data>
  <Data Name="HandleId">0x1678</Data>
  <Data Name="ProcessId">0xef0</Data>
  <Data Name="ProcessName">C:\Windows\explorer.exe</Data>
  <Data Name="TransactionId">{00000000-0000-0000-0000-000000000000}</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

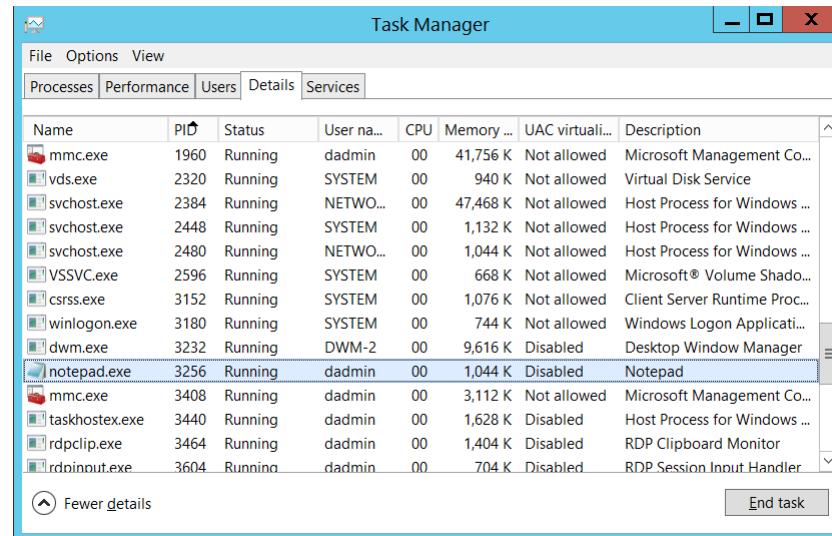
- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “**Security**” value for this event.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that deleted the object. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.
- **Transaction ID** [Type = GUID]: unique GUID of the transaction. This field can help you correlate this event with other events that might contain the same **Transaction ID**, such as “[4656\(S, F\): A handle to an object was requested](#).”

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Security Monitoring Recommendations:

For 4660(S): An object was deleted.

- This event doesn't contain the name of deleted object (only **Handle ID**). It is better to use “[4663\(S\): An attempt was made to access an object](#).” events with DELETE access to track object deletion actions.
- For kernel objects, this event and other auditing events have little to no security relevance and are hard to parse or analyze. There is no recommendation for auditing them, unless you know exactly what you need to monitor at the Kernel objects level.

4663(**S**): An attempt was made to access an object.

Event Properties - Event 4663, Microsoft Windows security auditing.

General **Details**

An attempt was made to access an object.

Subject:
 Security ID: CONTOSO\dadmin
 Account Name: dadmin

Object:	Object Server: Security Object Type: File Object Name: C:\Documents\HBI Data.txt Handle ID: 0x1bc Resource Attributes: SAI(RA;ID;WD;("Impact_MS",TI,0x10020,3000))
Process Information:	Process ID: 0x458 Process Name: C:\Windows\System32\notepad.exe
Access Request Information:	Accesses: WriteData (or AddFile) AppendData (or AddSubdirectory or CreatePipeInstance)
Access Mask:	0x6
Log Name:	Security
Source:	Microsoft Windows se
Event ID:	4663
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy **Close**

Event Description:

This event indicates that a specific operation was performed on an object. The object could be a file system, kernel, or registry object, or a file system object on removable storage or a device.

This event generates only if object's [SACL](#) has required ACE to handle specific access right use.

The main difference with "[4656](#): A handle to an object was requested." event is that 4663 shows that access right was used instead of just requested and 4663 doesn't have Failure events.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4663</EventID>
<Version>1</Version>
<Level>0</Level>
<Task>12800</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T22:13:54.770429700Z" />
<EventRecordID>273866</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
  
```

```

</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x4367b</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Documents\HBI Data.txt</Data>
<Data Name="HandleId">0x1bc</Data>
<Data Name="AccessList">%4417 %4418</Data>
<Data Name="AccessMask">0x6</Data>
  
```

```
<Data Name="ProcessId">0x458</Data>
<Data Name="ProcessName">C:\Windows\System32\notepad.exe</Data>
<Data Name="ResourceAttributes">S:AI(RA;ID;;;;WD;("Impact_MS",TI,0x10020,3000))</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.
- 1 - Windows Server 2012, Windows 8.
 - Added "Resource Attributes" field.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that made an attempt to access an object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]:** the name of the account that made an attempt to access an object.
- **Account Domain [Type = UnicodeString]:** subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID [Type = HexInt64]:** hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

Object:

- **Object Server [Type = UnicodeString]:** has "**Security**" value for this event.
- **Object Type [Type = UnicodeString]:** The type of object that was accessed during the operation.

The following table contains the list of the most common **Object Types**:

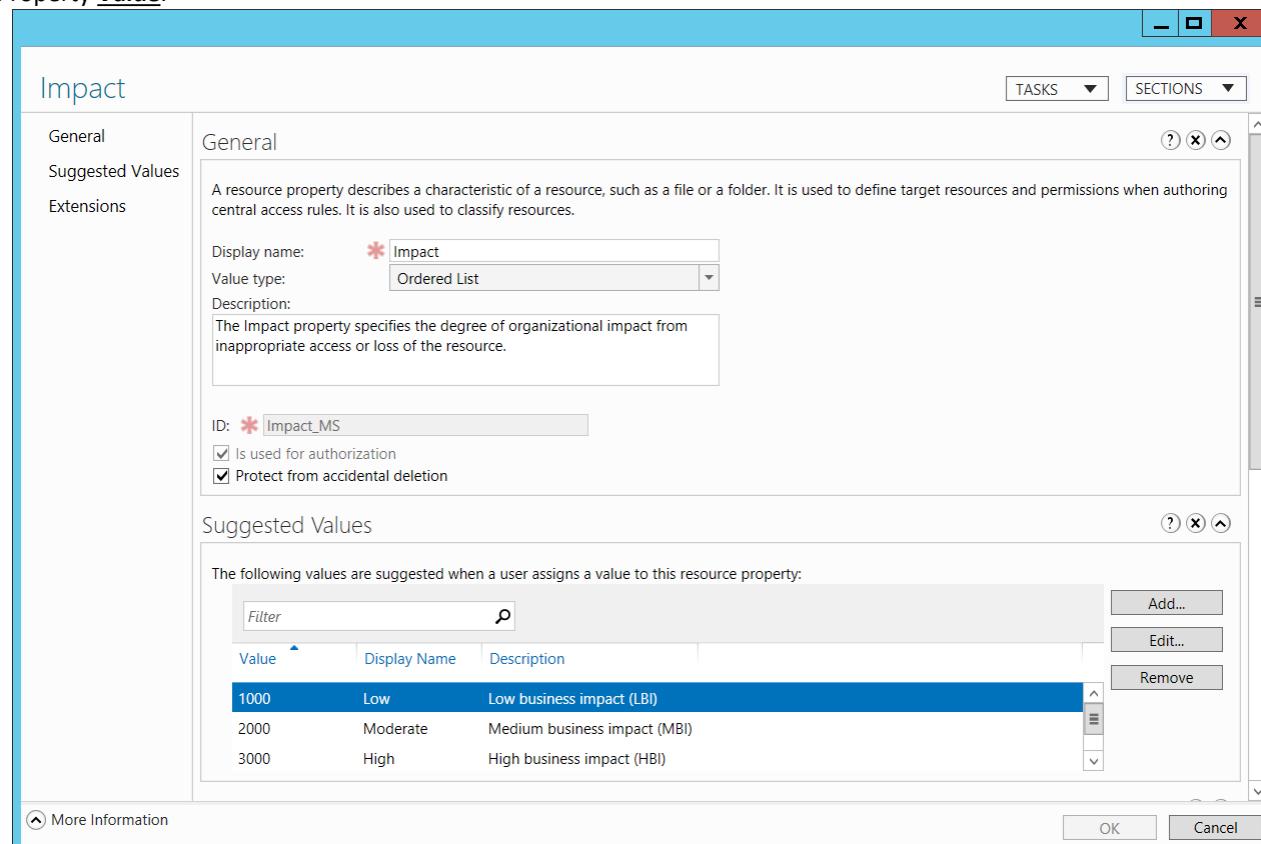
Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process

Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: name and other identifying information for the object for which access was requested. For example, for a file, the path would be included.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can be used for correlation with other events, for example with **Handle ID** field in “[4656\(S, F\)](#): A handle to an object was requested.” This parameter might not be captured in the event, and in that case appears as “0x0”.
- **Resource Attributes** [Type = UnicodeString] [Version 1]: attributes associated with the object. For some objects, the field does not apply and “-” is displayed.

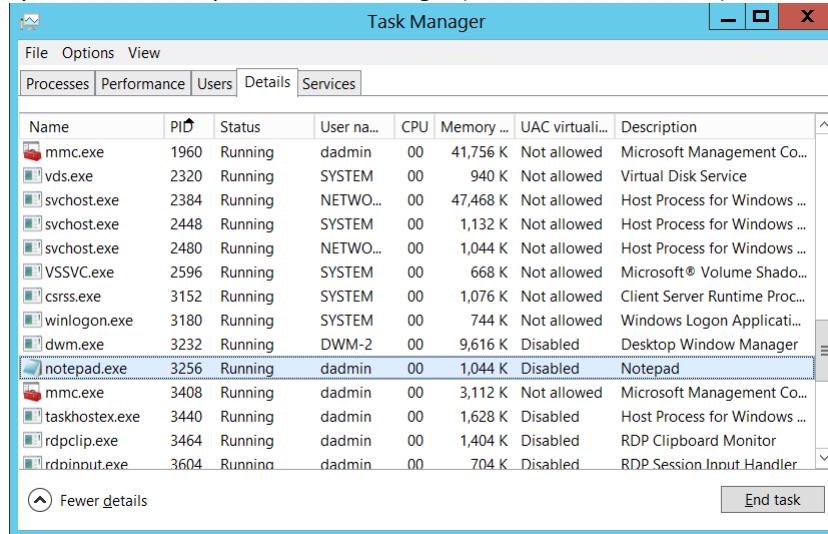
For example, for a file, the following might be displayed: S:AI(RA;ID;;;;WD;("Impact_MS",TI,0x10020,3000))

- Impact_MS: Resource Property ID.
- 3000: Recourse Property Value.



Process Information:

- **Process ID [Type = Pointer]:** hexadecimal Process ID of the process that accessed the object. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name [Type = UnicodeString]:** full path and the name of the executable for the process.

Access Request Information:

- **Accesses [Type = UnicodeString]:** the list of access rights which were used by **Subject\Security ID**. These access rights depend on **Object Type**. The following table contains information about the most common access rights for file system objects. Access rights for registry objects are often similar to file system objects, but the table contains a few notes about how they vary.

Access	Hex Value	Schema Value	Description
ReadData (or ListDirectory) (For registry objects, this is “Query key value.”)	0x1	%%4416	ReadData - For a file object, the right to read the corresponding file data. For a directory object, the right to read the corresponding directory data. ListDirectory - For a directory, the right to list the contents of the directory.
WriteData (or AddFile) (For registry objects, this is “Set key value.”)	0x2	%%4417	WriteData - For a file object, the right to write data to the file. For a directory object, the right to create a file in the directory (FILE_ADD_FILE). AddFile - For a directory, the right to create a file in the directory.

AppendData (or AddSubdirectory or CreatePipeInstance)	0x4	%%4418	AppendData - For a file object, the right to append data to the file. (For local files, write operations will not overwrite existing data if this flag is specified without FILE_WRITE_DATA .) For a directory object, the right to create a subdirectory (FILE_ADD_SUBDIRECTORY). AddSubdirectory - For a directory, the right to create a subdirectory. CreatePipeInstance - For a named pipe, the right to create a pipe.
ReadEA (For registry objects, this is "Enumerate sub-keys.")	0x8	%%4419	The right to read extended file attributes.
WriteEA	0x10	%%4420	The right to write extended file attributes.
	0x20	%%4421	Execute - For a native code file, the right to execute the file. This access right given to scripts may cause the script to be executable, depending on the script interpreter. Traverse - For a directory, the right to traverse the directory. By default, users are assigned the BYPASS_TRAVERSE_CHECKING privilege, which ignores the FILE_TRAVERSE access right. See the remarks in File Security and Access Rights for more information.
Execute/Traverse			
DeleteChild	0x40	%%4422	For a directory, the right to delete a directory and all the files it contains, including read-only files.
ReadAttributes	0x80	%%4423	The right to read file attributes.
WriteAttributes	0x100	%%4424	The right to write file attributes.
DELETE	0x10000	%%1537	The right to delete the object.
READ_CONTROL	0x20000	%%1538	The right to read the information in the object's security descriptor, not including the information in the system access control list (SACL).
WRITE_DAC	0x40000	%%1539	The right to modify the discretionary access control list (DACL) in the object's security descriptor.
WRITE_OWNER	0x80000	%%1540	The right to change the owner in the object's security descriptor
SYNCHRONIZE	0x100000	%%1541	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.
ACCESS_SYS_SEC	0x1000000	%%1542	The ACCESS_SYS_SEC access right controls the ability to get or set the SACL in an object's security descriptor.

Table 15. File System objects access rights.

- **Access Mask** [Type = HexInt32]: hexadecimal mask for the requested or performed operation. For more information, see the preceding table.

Security Monitoring Recommendations:

For 4663(S): An attempt was made to access an object.

For kernel objects, this event and other auditing events have little to no security relevance and are hard to parse or analyze. There is no recommendation for auditing them, unless you know exactly what you need to monitor at the Kernel objects level.

For other types of objects, the following recommendations apply.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have critical file system objects for which you need to monitor all access attempts, monitor this event for **Object Name**.
- If you have critical file system objects for which you need to monitor certain access attempts (for example, write actions), monitor this event for **Object Name** in relation to **Access Request Information\Accesses**.
- If you have file system objects with specific attributes, for which you need to monitor access attempts, monitor this event for **Resource Attributes**.
- If **Object Name** is a sensitive or critical registry key for which you need to monitor specific access attempts (for example, only write actions), monitor for all [4663](#) events with the corresponding **Access Request Information\Accesses**.
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- For file system objects, we recommend that you monitor for these **Access Request Information\Accesses** rights:
 - WriteData (or AddFile)
 - AppendData (or AddSubdirectory or CreatePipeInstance)
 - WriteEA
 - DeleteChild
 - WriteAttributes
 - DELETE
 - WRITE_DAC
 - WRITE_OWNER

 Event Properties - Event 4664, Microsoft Windows security audit...

General **Details**

An attempt was made to create a hard link.

```

Account Name: dadmin
Account Domain: CONTOSO
Logon ID: 0x43659

Link Information:
File Name: C:\notepad.exe
Link Name: C:\Docs\My.exe
Transaction ID: (00000000-0000-0000-0000-000000000000)

Log Name: Security
Source: Microsoft Windows security
Event ID: 4664
Level: Information
User: N/A
OpCode: Info
More Information: Event Log Online

```

Logged: 9/21/2015 4:50:26 PM

Task Category: File System

Keywords: Audit Success

Computer: DC01.contoso.local

Copy **Close**

4664(S): An attempt was made to create a hard link.

Event Description:

This event generates when an NTFS hard link was successfully created.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4664</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12800</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-21T23:50:26.871375900Z" />
<EventRecordID>276680</EventRecordID>

```

```
<Correlation />
<Execution ProcessID="4" ThreadID="2624" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x43659</Data>
<Data Name="FileName">C:\notepad.exe</Data>
<Data Name="LinkName">C:\Docs\My.exe</Data>
<Data Name="TransactionId">{00000000-0000-0000-0000-000000000000}</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made an attempt to create the hard link. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to create the hard link.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Link Information:

- **File Name** [Type = UnicodeString]: the name of a file or folder that new hard link refers to.
- **Link Name** [Type = UnicodeString]: full path name with new hard link file name.
- **Transaction ID** [Type = GUID]: unique GUID of the transaction. This field can help you correlate this event with other events that might contain the same **Transaction ID**, such as [“4660\(S\): An object was deleted.”](#)

This parameter might not be captured in the event, and in that case appears as “{00000000-0000-0000-0000-000000000000}”.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Security Monitoring Recommendations:

For 4664(S): An attempt was made to create a hard link.

- We recommend monitoring for any [4664](#) event, because this action is not typical for normal operating system behavior and can be a sign of malicious activity.

4985(S): The state of a transaction has changed.

Event Properties - Event 4985, Microsoft Windows security auditing. X

[General](#) [Details](#)

Event Description:
This is an informational event from file system [Transaction Manager](#).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Subject: Security ID: SYSTEM Account Name: DC01\$ Account Domain: CONTOSO Logon ID: 0x3E7	Event XML: <pre>- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>4985</EventID> <Version>0</Version> <Level>0</Level> <Task>12800</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-09-19T00:00:40.099093300Z" /> <EventRecordID>274277</EventRecordID> <Correlation /> <Execution ProcessID="4" ThreadID="5048" /> <Channel>Security</Channel> <Computer>DC01.contoso.local</Computer></pre>
Transaction Information: RM Transaction ID: {17ef5e21-5e2c-11e5-810f-00155d987005} New State: 52 Resource Manager: {5f5ed427-fcca-11e3-bd73-b54ab417b853}	
Process Information: Process ID: 0x370 Process Name: C:\Windows\System32\svchost.exe	
Log Name: Security Source: Microsoft Windows security Event ID: 4985 Level: Information User: N/A OpCode: Info More Information: Event Log Online	Copy Close

```
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
```

```
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TransactionId">{17EF5E21-5E2C-11E5-810F-00155D987005}</Data>
<Data Name="NewState">52</Data>
<Data Name="ResourceManager">{5F5ED427-FCCA-11E3-BD73-B54AB417B853}</Data>
<Data Name="ProcessId">0x370</Data>
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account through which the state of the transaction was changed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that changed the state of the transaction.
- **Account Domain** [Type = UnicodeString]: domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Transaction Information:

- **RM Transaction ID** [Type = GUID]: unique GUID of the [transaction](#). This field can help you correlate this event with other events that might contain the same **Transaction ID**, such as "[4656\(S, F\)](#): A handle to an object was requested."

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **New State** [Type = UInt32]: identifier of the new state of the [transaction](#).
- **Resource Manager** [Type = GUID]: unique GUID-Identifier of the [Resource Manager](#) which associated with this [transaction](#).

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the state of the transaction was changed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):

Task Manager							
Name	PID	Status	User na...	CPU	Memory ...	UAC virtuali...	Description
mmc.exe	1960	Running	dadmin	00	41,756 K	Not allowed	Microsoft Management Co...
vds.exe	2320	Running	SYSTEM	00	940 K	Not allowed	Virtual Disk Service
svchost.exe	2384	Running	NETWO...	00	47,468 K	Not allowed	Host Process for Windows ...
svchost.exe	2448	Running	SYSTEM	00	1,132 K	Not allowed	Host Process for Windows ...
svchost.exe	2480	Running	NETWO...	00	1,044 K	Not allowed	Host Process for Windows ...
VSSVC.exe	2596	Running	SYSTEM	00	668 K	Not allowed	Microsoft® Volume Shado...
csrss.exe	3152	Running	SYSTEM	00	1,076 K	Not allowed	Client Server Runtime Proc...
winlogon.exe	3180	Running	SYSTEM	00	744 K	Not allowed	Windows Logon Application
dwm.exe	3232	Running	DWM-2	00	9,616 K	Disabled	Desktop Window Manager
notepad.exe	3256	Running	dadmin	00	1,044 K	Disabled	Notepad
mmc.exe	3408	Running	dadmin	00	3,112 K	Not allowed	Microsoft Management Co...
taskhostex.exe	3440	Running	dadmin	00	1,628 K	Disabled	Host Process for Windows ...
rdpclip.exe	3464	Running	dadmin	00	1,404 K	Disabled	RDP Clipboard Monitor
rdpoinput.exe	3604	Running	dadmin	00	704 K	Disabled	RDP Session Input Handler

If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Security Monitoring Recommendations:

For 4985(S): The state of a transaction has changed.

- This event typically has no security relevance and used for [Transaction Manager](#) troubleshooting.

5051(-): A file was virtualized.

This event should be generated when file was virtualized using [LUAFV](#).

This event occurs very rarely during standard LUAFV file virtualization.

There is no example of this event in this document.

Event Schema:

A file was virtualized.

Subject:

Security ID:%1%

Account Name:%2

Account Domain:%3

Logon ID:%4

Object:

File Name:%5

Virtual File Name:%6

Process Information:

Process ID:%7

Process Name%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.



Event Properties - Event 4670, Microsoft Windows security audit... X

General **Details**

Permissions on an object were changed.

Subject:

Security ID:	CONTOSO\admind
Account Name:	admind
Account Domain:	CONTOSO
Logon ID:	0x43659

Object:

Object Server:	Security
Object Type:	File
Object Name:	C:\Documents\netcat-1.11
Handle ID:	0x3f0

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4670(S): Permissions on an object were changed.

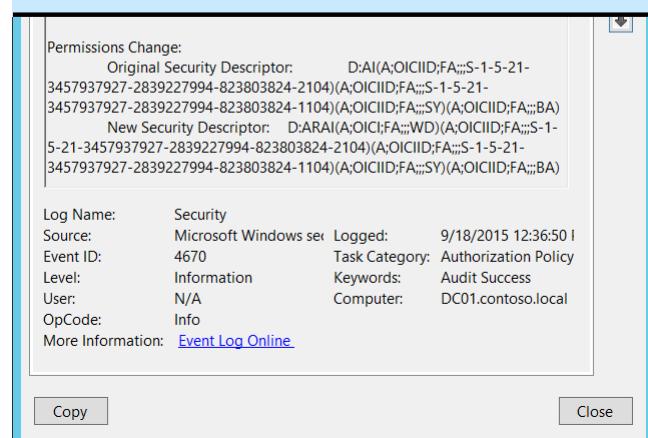
Event Description:

This event generates when the permissions for an object are changed. The object could be a file system, registry, or security token object.

This event does not generate if the [SACL](#) (Auditing ACL) was changed.

Before this event can generate, certain ACEs might need to be set in the object's [SACL](#). For example, for a file system object, it generates only if "Change Permissions" and/or "Take Ownership" are set in the object's SACL. For a registry key, it generates only if "Write DAC" and/or "Write Owner" are set in the object's SACL.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Permissions Change:
Original Security Descriptor: D:AI(A;OICIID;FA;;;S-1-5-21-3457937927-2839227994-823803824-2104)(A;OICIID;FA;;;S-1-5-21-3457937927-2839227994-823803824-1104)(A;OICIID;FA;;;SY)(A;OICIID;FA;;;BA)
New Security Descriptor: D:ARAI(A;OIC;FA;;;WD)(A;OICIID;FA;;;S-1-5-21-3457937927-2839227994-823803824-2104)(A;OICIID;FA;;;S-1-5-21-3457937927-2839227994-823803824-1104)(A;OICIID;FA;;;SY)(A;OICIID;FA;;;BA)

Log Name: Security
Source: Microsoft Windows sec Log ID: 9/18/2015 12:36:50 I
Event ID: 4670 Task Category: Authorization Policy
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy Close

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4670</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13570</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-18T19:36:50.187044600Z" />
<EventRecordID>269529</EventRecordID>
```

```
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x43659</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Documents\ncat-1.11</Data>
<Data Name="HandleId">0x3f0</Data>
<Data Name="OldSd">D:AI(A;OICIID;FA;;S-1-5-21-3457937927-2839227994-823803824-2104)(A;OICIID;FA;;S-1-5-21-3457937927-2839227994-823803824-1104)(A;OICIID;FA;;SY)(A;OICIID;FA;;BA)</Data>
<Data Name="NewSd">D:ARAI(A;OICI;FA;;;WD)(A;OICIID;FA;;S-1-5-21-3457937927-2839227994-823803824-2104)(A;OICIID;FA;;S-1-5-21-3457937927-2839227994-823803824-1104)(A;OICIID;FA;;SY)(A;OICIID;FA;;BA)</Data>
<Data Name="ProcessId">0xdb0</Data>
<Data Name="ProcessName">C:\Windows\System32\dllhost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change object’s permissions” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change object’s permissions” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO

- Lowercase full domain name: contoso.local
- Uppercase full domain name: CONTOSO.LOCAL
- For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “**Security**” value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation.

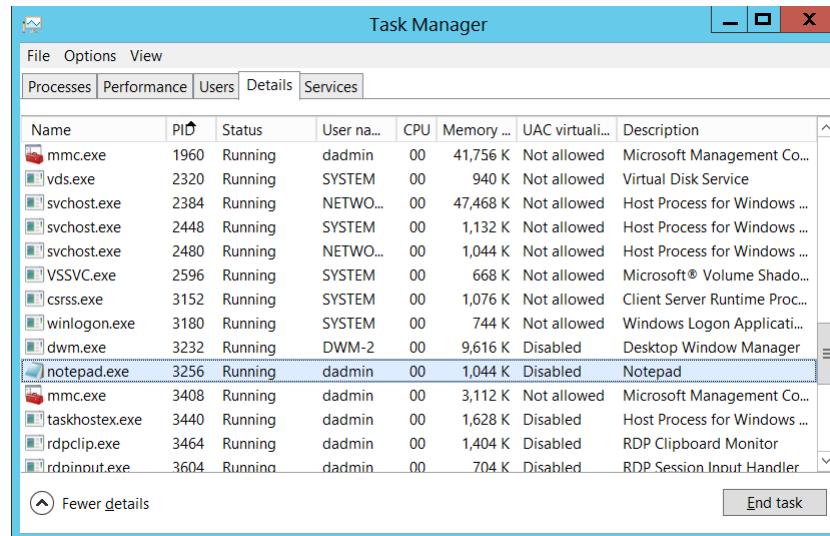
The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: name and other identifying information for the object for which permissions were changed. For example, for a file, the path would be included. For Token objects, this field typically equals “-”.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the permissions were changed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Permissions Change:

- **Original Security Descriptor** [Type = UnicodeString]: the old Security Descriptor Definition Language (SDDL) value for the object.
- **New Security Descriptor** [Type = UnicodeString]: the new Security Descriptor Definition Language (SDDL) value for the object.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

`O:BAG:SY:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)`

- O: Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators

"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- G: = Primary Group.
- D: = DACL Entries.
- S: = SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: **D:(A;FA;;WD)**

- **entry_type:**
 - "D" - DACL
 - "S" - SACL
- **inheritance_flags:**
 - "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.
 - "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" Is not also set.
 - "AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.
- **ace_type:**
 - "A" - ACCESS ALLOWED
 - "D" - ACCESS DENIED
 - "OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).
 - "OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).
 - "AU" - SYSTEM AUDIT
 - "A" - SYSTEM ALARM
 - "OU" - OBJECT SYSTEM AUDIT
 - "OL" - OBJECT SYSTEM ALARM
- **ace_flags:**
 - "CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.
 - "OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.
 - "NP" - NO PROPAGATE: only immediate children inherit this ace.
 - "IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.

"ID" - ACE IS INHERITED

"SA" - SUCCESSFUL ACCESS AUDIT

"FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A

- inherit_object_guid: N/A

- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,

[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 4670(S): Permissions on an object were changed.

For token objects, this is typically an informational event, and at the same time it is difficult to identify which token's permission were changed. For token objects, there are no monitoring recommendations for this event in this document.

For file system and registry objects, the following recommendations apply.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If you have critical registry objects for which you need to monitor all modifications (especially permissions changes and owner changes), monitor for the specific **Object\ObjectName**.
- If you have high-value computers for which you need to monitor all changes for all or specific objects (for example, file system or registry objects), monitor for all [4670](#) events on these computers. For example, you could monitor the **ntds.dit** file on domain controllers.

Audit Filtering Platform Connection

Audit Filtering Platform Connection determines whether the operating system generates audit events when connections are allowed or blocked by the [Windows Filtering Platform](#). Windows Filtering Platform (WFP) enables independent software vendors (ISVs) to filter and modify TCP/IP packets, monitor or authorize connections, filter Internet Protocol security (IPSec)-protected traffic, and filter remote procedure calls (RPCs).

This subcategory contains Windows Filtering Platform events about blocked and allowed connections, blocked and allowed port bindings, blocked and allowed port listening actions, and blocked to accept incoming connections applications.

Event volume: High.

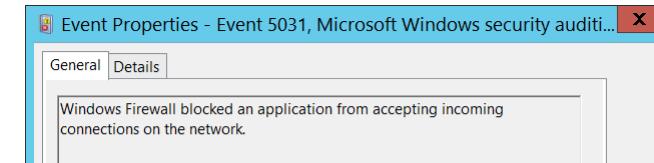
Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	Yes	IF	Yes	Success auditing for this subcategory typically generates a very high volume of events, for example, one event for every connection that was made to the system. It is much more important to audit Failure events (blocked connections, for example). For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. IF - Enable Success audit in case you need to monitor successful outbound or inbound connections to and from untrusted IP addresses on high value computers or devices.
Member Server	No	Yes	IF	Yes	Success auditing for this subcategory typically generates a very high volume of events, for example, one event for every connection that was made to the system. It is much more important to audit Failure events (blocked connections, for example). For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. IF - Enable Success audit in case you need to monitor successful outbound or inbound connections to and from untrusted IP addresses on high value computers or devices.
Workstation	No	Yes	IF	Yes	Success auditing for this subcategory typically generates a very high volume of events, for example, one event for every connection that was made to the system. It is much more important to audit Failure events (blocked connections, for example). For recommendations for using and analyzing the collected information, see the Security Monitoring Recommendations sections. IF - Enable Success audit in case you need to monitor successful outbound or inbound connections to and from untrusted IP addresses on high value computers or devices.

Events List:

- [5031\(F\)](#): The Windows Firewall Service blocked an application from accepting incoming connections on the network.
- [5150\(-\)](#): The Windows Filtering Platform blocked a packet.
- [5151\(-\)](#): A more restrictive Windows Filtering Platform filter has blocked a packet.
- [5154\(S\)](#): The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.
- [5155\(F\)](#): The Windows Filtering Platform has blocked an application or service from listening on a port for incoming connections.
- [5156\(S\)](#): The Windows Filtering Platform has permitted a connection.
- [5157\(F\)](#): The Windows Filtering Platform has blocked a connection.

- [5158\(S\)](#): The Windows Filtering Platform has permitted a bind to a local port.
- [5159\(F\)](#): The Windows Filtering Platform has blocked a bind to a local port.

5031(F): The Windows Firewall Service blocked an application from accepting incoming connections on the network.



Event Description:

This event generates when an application was blocked from accepting incoming connections on the network by [Windows Filtering Platform](#).

If you don't have any firewall rules (Allow or Deny) in Windows Firewall for specific applications, you will get this event from [Windows Filtering Platform](#) layer, because by default this layer is denying any incoming connections.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Source:	Microsoft Windows security audit	Logged:	9/21/2015 8:46:36 PM
Event ID:	5031	Task Category:	Filtering Platform Cc
Level:	Information	Keywords:	Audit Failure
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info		
More Information: Event Log Online			

Copy **Close**

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5031</EventID>
<Version>0</Version>
```

```
<Level>0</Level>
<Task>12810</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T03:46:36.634473000Z" />
<EventRecordID>304373</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="2976" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
```

```
- <EventData>
<Data Name="Profiles">Domain</Data>
<Data Name="Application">C:\documents\listener.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

- **Profiles** [Type = UnicodeString]: network profile using which application was blocked. Possible values:
 - Domain
 - Public
 - Private
- **Application** [Type = UnicodeString]: full path and file name of executable file for blocked application.

Security Monitoring Recommendations:

For 5031(F): The Windows Firewall Service blocked an application from accepting incoming connections on the network.

- You can use this event to detect applications for which no Windows Firewall rules were created.
- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”

5150(-): The Windows Filtering Platform blocked a packet.

This event is logged if the Windows Filtering Platform [MAC filter](#) blocked a packet.

There is no example of this event in this document.

Event Schema:

The Windows Filtering Platform has blocked a packet.

Network Information:

*Direction:%1
Source Address:%2
Destination Address:%3
EtherType:%4
MediaType:%5
InterfaceType:%6
VlanTag:%7*

Filter Information:

*Filter Run-Time ID:%8
Layer Name:%9
Layer Run-Time ID:%10*

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

5151(-): A more restrictive Windows Filtering Platform filter has blocked a packet.

This event is logged if a more restrictive Windows Filtering Platform [MAC filter](#) has blocked a packet.

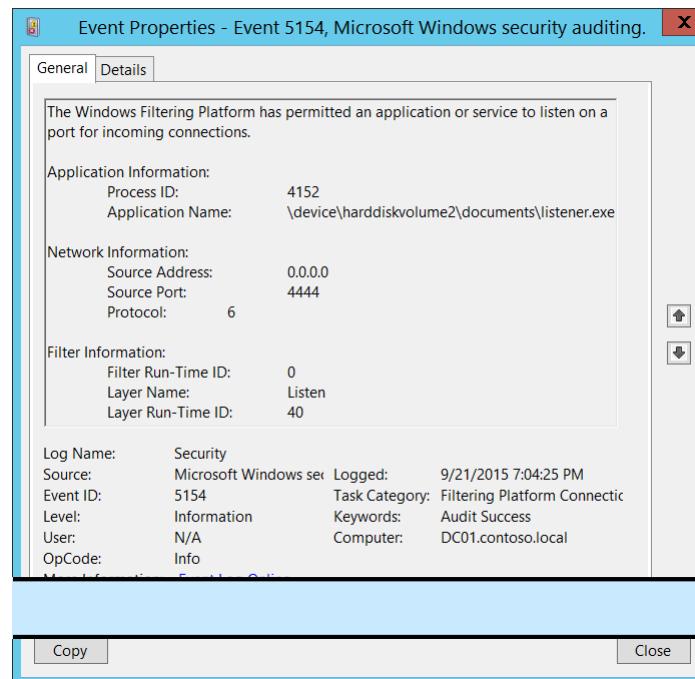
There is no example of this event in this document.

Event Schema:

A more restrictive Windows Filtering Platform filter has blocked a packet.

Network Information:

*Direction:%1
Source Address:%2
Destination Address:%3
EtherType:%4
MediaType:%5
InterfaceType:%6*

 Event Properties - Event 5154, Microsoft Windows security auditing.

The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.

Application Information:	
Process ID:	4152
Application Name:	\device\harddiskvolume2\documents\listener.exe
Network Information:	
Source Address:	0.0.0.0
Source Port:	4444
Protocol:	6
Filter Information:	
Filter Run-Time ID:	0
Layer Name:	Listen
Layer Run-Time ID:	40
Log Name: Security	
Source: Microsoft Windows sec	Logged: 9/21/2015 7:04:25 PM
Event ID: 5154	Task Category: Filtering Platform Connectic
Level: Information	Keywords: Audit Success
User: N/A	Computer: DC01.contoso.local
OpCode: Info	

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Copy **Close**

VlanTag:%7

Filter Information:

*Filter Run-Time ID:%8
Layer Name:%9
Layer Run-Time ID:%10*

Required Server Roles:

None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions:

0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

5154(S): The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.

Event Description:

This event generates every time [Windows Filtering Platform](#) permits an application or service to listen on a port.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5154</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12810</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T02:04:25.757462900Z" />
<EventRecordID>287929</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="3968" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ProcessId">4152</Data>
<Data Name="Application">\device\harddiskvolume2\documents\listener.exe</Data>
<Data Name="SourceAddress">0.0.0.0</Data>
<Data Name="SourcePort">4444</Data>
<Data Name="Protocol">6</Data>
<Data Name="FilterRTID">0</Data>
<Data Name="LayerName">%14609</Data>
<Data Name="LayerRTID">40</Data>
</EventData>
</Event>
```

Required Server Roles: None.

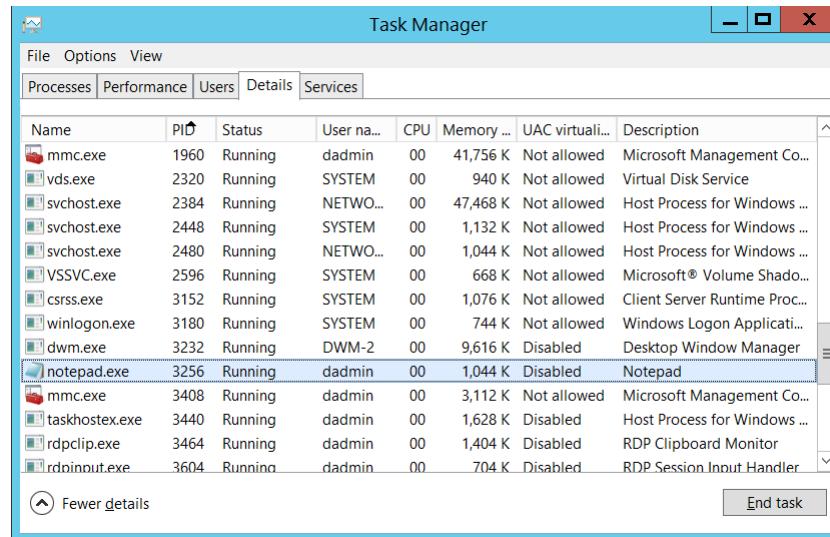
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Application Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process which was permitted to listen on the port. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Application Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Logical disk is displayed in format \device\harddiskvolume#. You can get all local volume numbers by using **diskpart** utility. The command to get volume numbers using diskpart is “list volume”:

```
c:\windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (c) 1999-2013 Microsoft Corporation.
On computer: DC01

DISKPART> list volume

  volume ###  Ltr  Label        Fs     Type        size    status     Info
  -----  ---  -----  -----  -----  -----  -----  -----
  Volume 0      D          DVD-ROM   0 B  No Media
  Volume 1      System  Rese NTFS  Partition  350 MB  Healthy  System
  Volume 2      C          NTFS  Partition  126 GB  Healthy  Boot
```

Network Information:

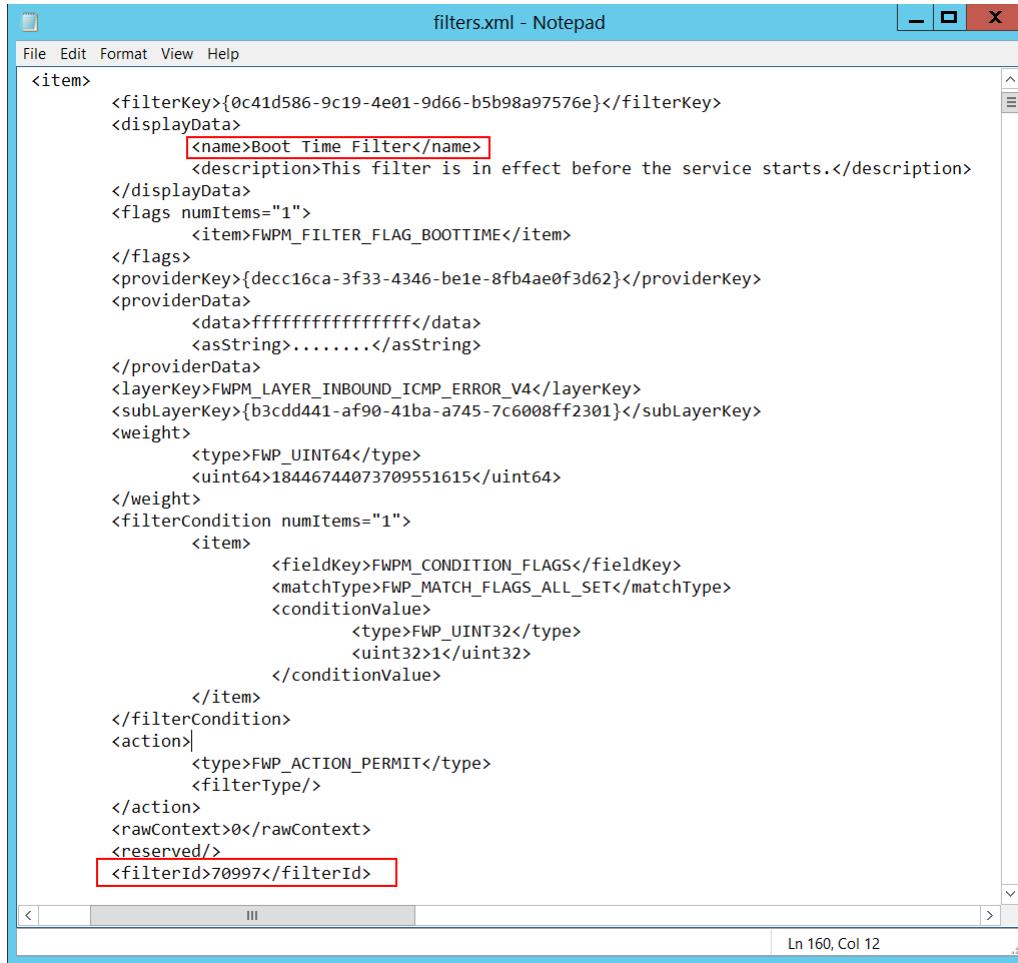
- **Source Address** [Type = UnicodeString]: local IP address on which application requested to listen on the port.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format

- 127.0.0.1 , ::1 - localhost
- **Source Port** [Type = UnicodeString]: source TCP\UDP port number which was requested for listening by application.
- **Protocol** [Type = UInt32]: protocol number. For example:
 - 6 – TCP.
 - 17 – UDP.

More information about possible values for this field: <https://technet.microsoft.com/en-us/library/cc959827.aspx>.

Filter Information:

- **Filter Run-Time ID** [Type = UInt64]: unique filter ID which allows application to listen on the specific port. By default Windows firewall won't prevent a port from being listened by an application and if this application doesn't match any filters you will get value **0** in this field.
To find specific Windows Filtering Platform filter by ID you need to execute the following command: **netsh wfp show filters**. As result of this command **filters.xml** file will be generated. You need to open this file and find specific substring with required filter ID (**<filterId>**), for example:



The screenshot shows a Notepad window titled "filters.xml - Notepad". The XML code defines a filter named "Boot Time Filter" with provider key {decc16ca-3f33-4346-be1e-8fb4ae0f3d62} and sub-layer key {b3cdd441-af90-41ba-a745-7c6008ff2301}. The filter has a weight of 18446744073709551615 and a condition that matches all flags. It permits traffic and has a reserved field. The filter ID is 70997.

```

<item>
    <filterKey>{0c41d586-9c19-4e01-9d66-b5b98a97576e}</filterKey>
    <displayData>
        <name>Boot Time Filter</name>
        <description>This filter is in effect before the service starts.</description>
    </displayData>
    <flags numItems="1">
        <item>FWPM_FILTER_FLAG_BOOTTIME</item>
    </flags>
    <providerKey>{decc16ca-3f33-4346-be1e-8fb4ae0f3d62}</providerKey>
    <providerData>
        <data>ffffffffffff</data>
        <asString>.....</asString>
    </providerData>
    <layerKey>FWPM_LAYER_INBOUND_ICMP_ERROR_V4</layerKey>
    <subLayerKey>{b3cdd441-af90-41ba-a745-7c6008ff2301}</subLayerKey>
    <weight>
        <type>FWP_UINT64</type>
        <uint64>18446744073709551615</uint64>
    </weight>
    <filterCondition numItems="1">
        <item>
            <fieldKey>FWPM_CONDITION_FLAGS</fieldKey>
            <matchType>FWP_MATCH_FLAGS_ALL_SET</matchType>
            <conditionValue>
                <type>FWP_UINT32</type>
                <uint32>1</uint32>
            </conditionValue>
        </item>
    </filterCondition>
    <action>
        <type>FWP_ACTION_PERMIT</type>
        <filterType/>
    </action>
    <rawContext>0</rawContext>
    <reserved/>
    <filterId>70997</filterId>

```

- **Layer Name** [Type = UnicodeString]: [Application Layer Enforcement](#) layer name.
- **Layer Run-Time ID** [Type = UInt64]: Windows Filtering Platform layer identifier. To find specific Windows Filtering Platform layer ID you need to execute the following command: **netsh wfp show state**. As result of this command **wfpstate.xml** file will be generated. You need to open this file and find specific substring with required layer ID (**<layerId>**), for example:

wfpstate.xml - Notepad

```

</item>
<item>
    <fieldKey>FWPM_CONDITION_INTERFACE_QUARANTINE_EPOCH</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_UINT64</dataType>
</item>
<item>
    <fieldKey>FWPM_CONDITION_ALE_PACKAGE_ID</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_SID</dataType>
</item>
</field>
<defaultSubLayerKey>FWPM_SUBLAYER_UNIVERSAL</defaultSubLayerKey>
<layerId>44</layerId>
</layer>
<callouts numItems="6">
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V4</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v4 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4</applicableLayer>
        <calloutId>13</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V6</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v6 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V6</applicableLayer>
        <calloutId>14</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V5</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v5 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V5</applicableLayer>
        <calloutId>15</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V2</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v2 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V2</applicableLayer>
        <calloutId>16</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V3</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v3 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V3</applicableLayer>
        <calloutId>17</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V1</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v1 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V1</applicableLayer>
        <calloutId>18</calloutId>
    </item>
</callouts>

```

Ln 4584, Col 37

Security Monitoring Recommendations:

For 5154(S): The Windows Filtering Platform has permitted an application or service to listen on a port for incoming connections.

- If you have a “whitelist” of applications that are associated with certain operating systems or server roles, and that are expected to listen on specific ports, monitor this event for “**Application Name**” and other relevant information.
- If a certain application is allowed to listen only on specific port numbers, monitor this event for “**Application Name**” and “**Network Information\Source Port**.”
- If a certain application is allowed to listen only on a specific IP address, monitor this event for “**Application Name**” and “**Network Information\Source Address**.”
- If a certain application is allowed to use only TCP or UDP protocols, monitor this event for “**Application Name**” and the protocol number in “**Network Information\Protocol**.”
- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”
- Typically this event has an informational purpose.

5155(F): The Windows Filtering Platform has blocked an application or service from listening on a port for incoming connections.

By default Windows firewall won't prevent a port from being listened by an application. In the other word, Windows system will not generate Event 5155 by itself.

You can add your own filters using the WFP APIs to block listen to reproduce this event: [https://msdn.microsoft.com/en-us/library/aa364046\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/aa364046(v=vs.85).aspx).

There is no event example in this document.

Event Schema:

The Windows Filtering Platform has blocked an application or service from listening on a port for incoming connections.

Application Information:

Process ID:%1

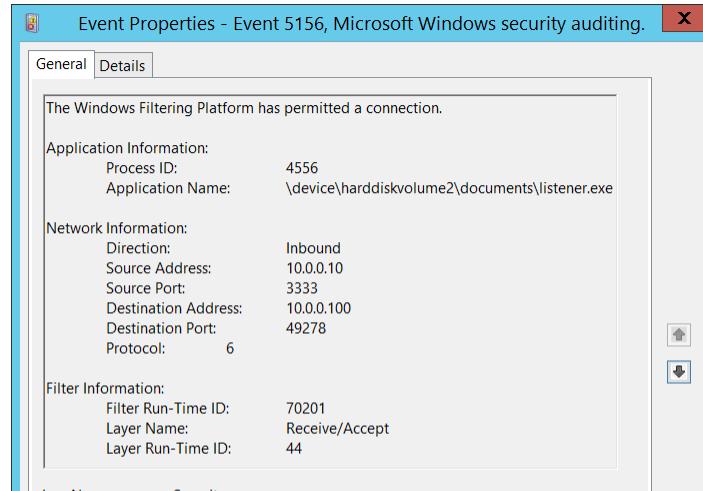
Application Name:%2

Network Information:

Source Address:%3

Source Port:%4

Protocol:%5

 Event Properties - Event 5156, Microsoft Windows security auditing.

The Windows Filtering Platform has permitted a connection.

Application Information:	Process ID: 4556
	Application Name: \device\harddiskvolume2\documents\listener.exe
Network Information:	Direction: Inbound
	Source Address: 10.0.0.10
	Source Port: 3333
	Destination Address: 10.0.0.100
	Destination Port: 49278
	Protocol: 6
Filter Information:	Filter Run-Time ID: 70201
	Layer Name: Receive/Accept
	Layer Run-Time ID: 44
Log Name:	Security

Filter Information:

Filter Run-Time ID:%6

Layer Name:%7

Layer Run-Time ID:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- If you use Windows Filtering Platform APIs to block application or services from listening on a port, then you can use this event for troubleshooting and monitoring.

5156(S): The Windows Filtering Platform has permitted a connection.

Event Description:

This event generates when [Windows Filtering Platform](#) has allowed a connection.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

User: N/A	Computer: DC01.contoso.local
OpCode: Info	
More Information: Event Log Online	

Copy **Close**

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
```

```
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5156</EventID>
<Version>1</Version>
<Level>0</Level>
<Task>12810</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T05:24:22.622090200Z" />
<EventRecordID>308129</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="3712" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="ProcessID">4556</Data>
  <Data Name="Application">\device\harddiskvolume2\documents\listener.exe</Data>
  <Data Name="Direction">%%14592</Data>
  <Data Name="SourceAddress">10.0.0.10</Data>
  <Data Name="SourcePort">3333</Data>
  <Data Name="DestAddress">10.0.0.100</Data>
  <Data Name="DestPort">49278</Data>
  <Data Name="Protocol">6</Data>
  <Data Name="FilterRTID">70201</Data>
  <Data Name="LayerName">%%14610</Data>
  <Data Name="LayerRTID">44</Data>
  <Data Name="RemoteUserID">S-1-0-0</Data>
  <Data Name="RemoteMachineID">S-1-0-0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

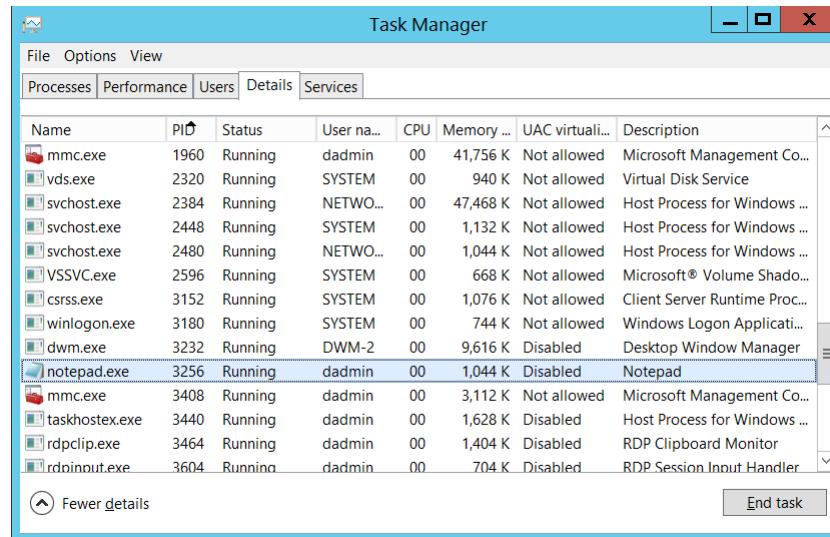
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Application Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process which received the connection. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Application Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Logical disk is displayed in format \device\harddiskvolume#. You can get all local volume numbers by using **diskpart** utility. The command to get volume numbers using diskpart is “list volume”:

```
C:\Windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (C) 1999-2013 Microsoft Corporation.
On computer: DC01

DISKPART> list volume

  volume ###  Ltr  Label        Fs  Type        size    status     Info
  -----  ---  -----  ----  -----  -----  -----  -----
  Volume 0      D                DVD-ROM       0 B  No Media
  Volume 1      System Rese    NTFS  Partition   350 MB  Healthy
  Volume 2      C                NTFS  Partition  126 GB  Healthy
                                         System
                                         Boot
```

Network Information:

- **Direction** [Type = UnicodeString]: direction of allowed connection.
 - Inbound – for inbound connections.
 - Outbound – for outbound connections.
- **Source Address** [Type = UnicodeString]: local IP address on which application received the connection.
 - IPv4 Address

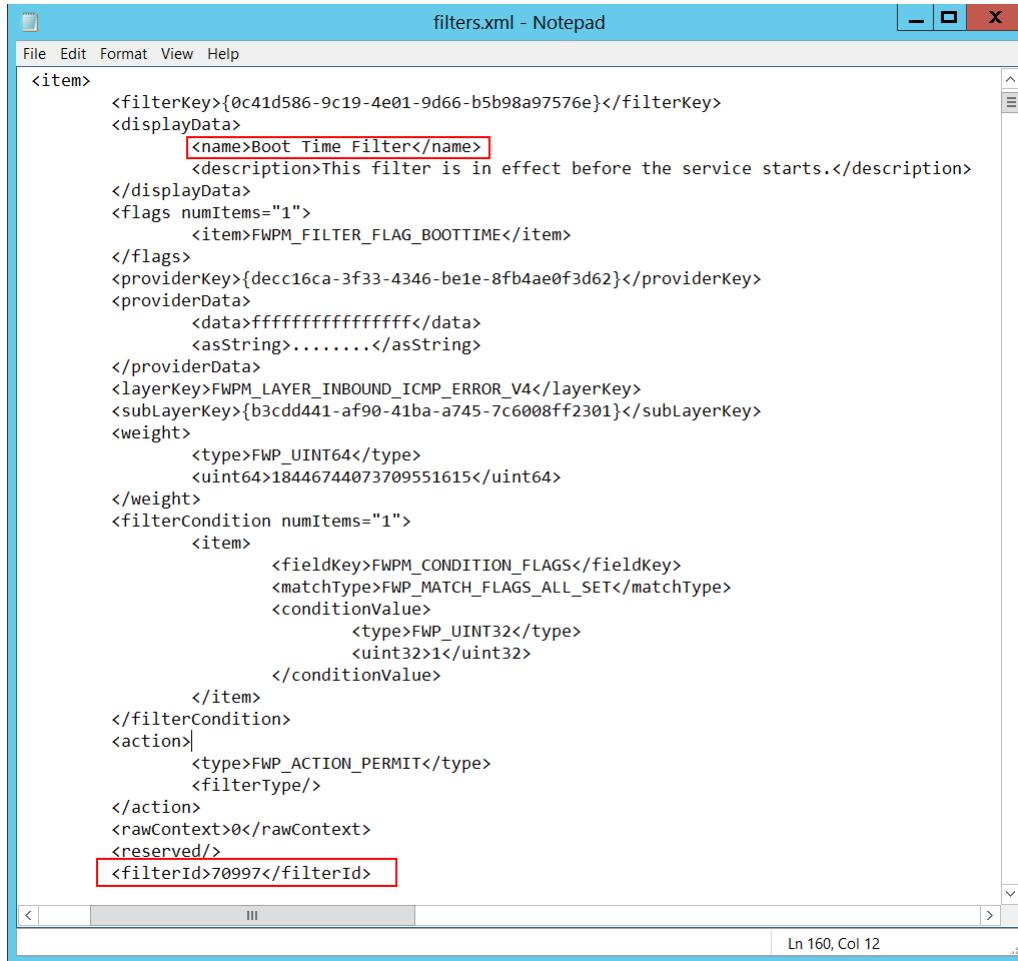
- IPv6 Address
- :: - all IP addresses in IPv6 format
- 0.0.0.0 - all IP addresses in IPv4 format
- 127.0.0.1 , ::1 - localhost
- **Source Port** [Type = UnicodeString]: port number on which application received the connection.
- **Destination Address** [Type = UnicodeString]: IP address from which connection was received or initiated.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format
 - 127.0.0.1 , ::1 - localhost
- **Destination Port** [Type = UnicodeString]: port number which was used from remote machine to initiate connection.
- **Protocol** [Type = UInt32]: number of protocol which was used.

Service	Protocol Number
Internet Control Message Protocol (ICMP)	1
Transmission Control Protocol (TCP)	6
User Datagram Protocol (UDP)	17
General Routing Encapsulation (PPTP data over GRE)	47
Authentication Header (AH) IPSec	51
Encapsulation Security Payload (ESP) IPSec	50
Exterior Gateway Protocol (EGP)	8
Gateway-Gateway Protocol (GGP)	3
Host Monitoring Protocol (HMP)	20
Internet Group Management Protocol (IGMP)	88
MIT Remote Virtual Disk (RVD)	66
OSPF Open Shortest Path First	89
PARC Universal Packet Protocol (PUP)	12
Reliable Datagram Protocol (RDP)	27
Reservation Protocol (RSVP) QoS	46

Filter Information:

- **Filter Run-Time ID** [Type = UInt64]: unique filter ID which allowed the connection.

To find specific Windows Filtering Platform filter by ID you need to execute the following command: **netsh wfp show filters**. As result of this command **filters.xml** file will be generated. You need to open this file and find specific substring with required filter ID (**<filterId>**), for example:



The screenshot shows a Notepad window titled "filters.xml - Notepad". The XML code defines a filter named "Boot Time Filter" with provider key {decc16ca-3f33-4346-be1e-8fb4ae0f3d62} and sub-layer key {b3cdd441-af90-41ba-a745-7c6008ff2301}. The filter has a weight of 18446744073709551615 and a condition that matches all flags. It permits traffic and has a layer ID of 70997.

```

<item>
    <filterKey>{0c41d586-9c19-4e01-9d66-b5b98a97576e}</filterKey>
    <displayData>
        <name>Boot Time Filter</name>
        <description>This filter is in effect before the service starts.</description>
    </displayData>
    <flags numItems="1">
        <item>FWPM_FILTER_FLAG_BOOTTIME</item>
    </flags>
    <providerKey>{decc16ca-3f33-4346-be1e-8fb4ae0f3d62}</providerKey>
    <providerData>
        <data>ffffffffffff</data>
        <asString>.....</asString>
    </providerData>
    <layerKey>FWPM_LAYER_INBOUND_ICMP_ERROR_V4</layerKey>
    <subLayerKey>{b3cdd441-af90-41ba-a745-7c6008ff2301}</subLayerKey>
    <weight>
        <type>FWP_UINT64</type>
        <uint64>18446744073709551615</uint64>
    </weight>
    <filterCondition numItems="1">
        <item>
            <fieldKey>FWPM_CONDITION_FLAGS</fieldKey>
            <matchType>FWP_MATCH_FLAGS_ALL_SET</matchType>
            <conditionValue>
                <type>FWP_UINT32</type>
                <uint32>1</uint32>
            </conditionValue>
        </item>
    </filterCondition>
    <action>
        <type>FWP_ACTION_PERMIT</type>
        <filterType/>
    </action>
    <rawContext>0</rawContext>
    <reserved/>
    <filterId>70997</filterId>

```

- **Layer Name** [Type = UnicodeString]: [Application Layer Enforcement](#) layer name.
- **Layer Run-Time ID** [Type = UInt64]: Windows Filtering Platform layer identifier. To find specific Windows Filtering Platform layer ID you need to execute the following command: **netsh wfp show state**. As result of this command **wfpstate.xml** file will be generated. You need to open this file and find specific substring with required layer ID (**<layerId>**), for example:

wfpstate.xml - Notepad

```

</item>
<item>
    <fieldKey>FWPM_CONDITION_INTERFACE_QUARANTINE_EPOCH</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_UINT64</dataType>
</item>
<item>
    <fieldKey>FWPM_CONDITION_ALE_PACKAGE_ID</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_SID</dataType>
</item>
</field>
<defaultSubLayerKey>FWPM_SUBLAYER_UNIVERSAL</defaultSubLayerKey>
<layerId>44</layerId>
</layer>
<callouts numItems="6">
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V4</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v4 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4</applicableLayer>
        <calloutId>13</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V6</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v6 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V6</applicableLayer>
        <calloutId>14</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V5</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v5 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V5</applicableLayer>
        <calloutId>15</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V2</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v2 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V2</applicableLayer>
        <calloutId>16</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V3</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v3 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V3</applicableLayer>
        <calloutId>17</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V1</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v1 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V1</applicableLayer>
        <calloutId>18</calloutId>
    </item>
</callouts>

```

Ln 4584, Col 37

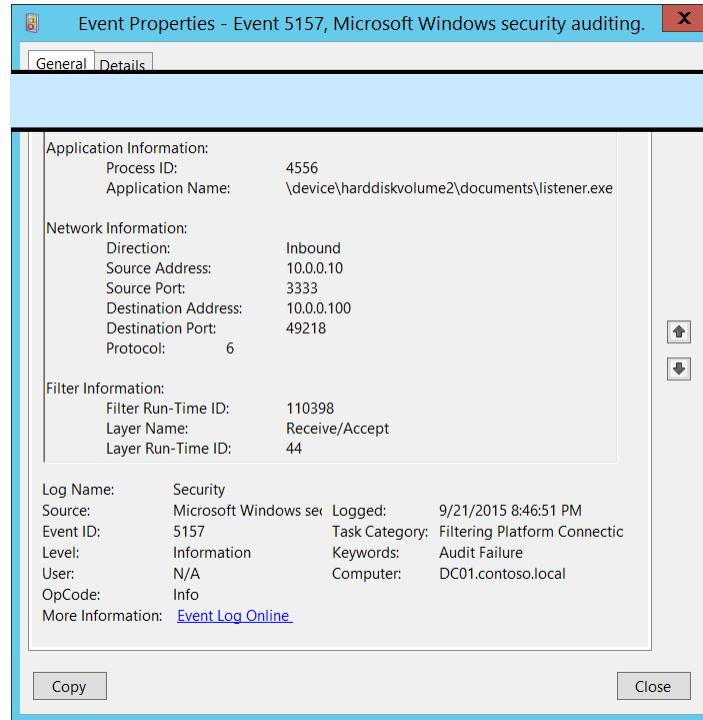
Security Monitoring Recommendations:

For 5156(S): The Windows Filtering Platform has permitted a connection.

- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”
- Check that “**Source Address**” is one of the addresses assigned to the computer.
- If the computer or device should not have access to the Internet, or contains only applications that don’t connect to the Internet, monitor for [5156](#) events where “**Destination Address**” is an IP address from the Internet (not from private IP ranges).
- If you know that the computer should never contact or be contacted by certain network IP addresses, monitor for these addresses in “**Destination Address**.”
- If you have a “whitelist” of IP addresses that the computer or device is expected to contact or be contacted by, monitor for IP addresses in “**Destination Address**” that are not in the whitelist.
- If you need to monitor all inbound connections to a specific local port, monitor for [5156](#) events with that “**Source Port**.”

- Monitor for all connections with a “**Protocol Number**” that is not typical for this device or computer, for example, anything other than 1, 6, or 17.
- If the computer’s communication with “**Destination Address**” should always use a specific “**Destination Port**,” monitor for any other “**Destination Port**.”

5157(F): The Windows Filtering Platform has blocked a connection.

 Event Properties - Event 5157, Microsoft Windows security auditing.

Event Description:
This event generates when [Windows Filtering Platform](#) has blocked a connection.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Application Information: Process ID: 4556 Application Name: \device\harddiskvolume2\documents\listener.exe	Network Information: Direction: Inbound Source Address: 10.0.0.10 Source Port: 3333 Destination Address: 10.0.0.100 Destination Port: 49218 Protocol: 6	Filter Information: Filter Run-Time ID: 110398 Layer Name: Receive/Accept Layer Run-Time ID: 44
Log Name: Security Source: Microsoft Windows security Event ID: 5157 Level: Information User: N/A OpCode: Info More Information: Event Log Online	Logged: 9/21/2015 8:46:51 PM Task Category: Filtering Platform Connectic Keywords: Audit Failure Computer: DC01.contoso.local	

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5157</EventID>
<Version>1</Version>
<Level>0</Level>
<Task>12810</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T03:46:51.662750400Z" />
<EventRecordID>304390</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="4520" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
```

```

- <EventData>
<Data Name="ProcessID">4556</Data>
<Data Name="Application">\device\harddiskvolume2\documents\listener.exe</Data>
<Data Name="Direction">%14592</Data>
<Data Name="SourceAddress">10.0.0.10</Data>
<Data Name="SourcePort">3333</Data>
<Data Name="DestAddress">10.0.0.100</Data>
<Data Name="DestPort">49218</Data>
<Data Name="Protocol">6</Data>
<Data Name="FilterRTID">110398</Data>
<Data Name="LayerName">%14610</Data>
```

```
<Data Name="LayerRTID">44</Data>
<Data Name="RemoteUserID">S-1-0-0</Data>
<Data Name="RemoteMachineID">S-1-0-0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

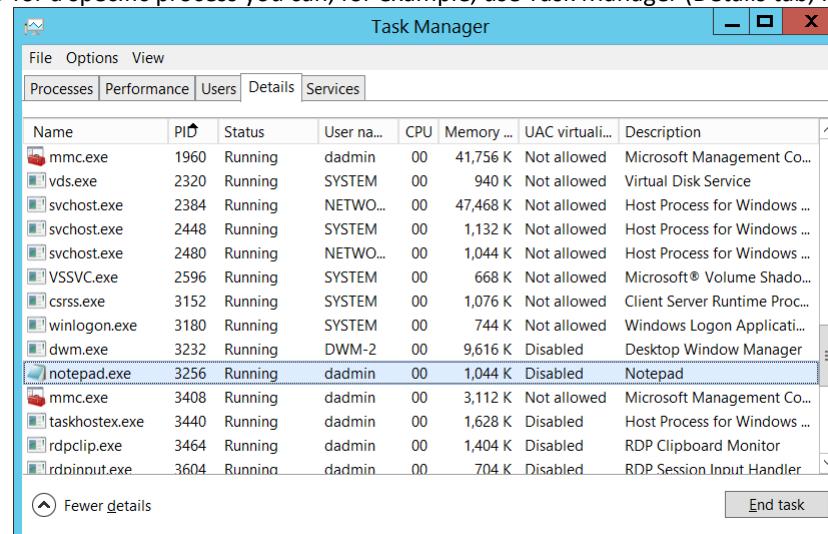
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Application Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted to create the connection. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Application Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Logical disk is displayed in format \device\harddiskvolume#. You can get all local volume numbers by using **diskpart** utility. The command to get volume numbers using diskpart is “**list volume**”:

```
C:\Windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (C) 1999-2013 Microsoft corporation.
On computer: DC01
DISKPART> list volume
Volume ### Ltr Label Fs Type Size Status Info
----- -- -- -- -- -- --
volume 0 D System NTFS DVD-ROM 0 B No Media
Volume 1 System Rese NTFS Partition 350 MB Healthy
Volume 2 C NTFS Partition 126 GB Healthy System Boot
```

Network Information:

- **Direction** [Type = UnicodeString]: direction of blocked connection.
 - Inbound – for inbound connections.
 - Outbound – for unbound connections.
- **Source Address** [Type = UnicodeString]: local IP address on which application received the connection.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format
 - 127.0.0.1 , ::1 - localhost
- **Source Port** [Type = UnicodeString]: port number on which application received the connection.
- **Destination Address** [Type = UnicodeString]: IP address from which connection was received or initiated.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format
 - 127.0.0.1 , ::1 - localhost
- **Destination Port** [Type = UnicodeString]: port number which was used from remote machine to initiate connection.
- **Protocol** [Type = UInt32]: number of protocol which was used.

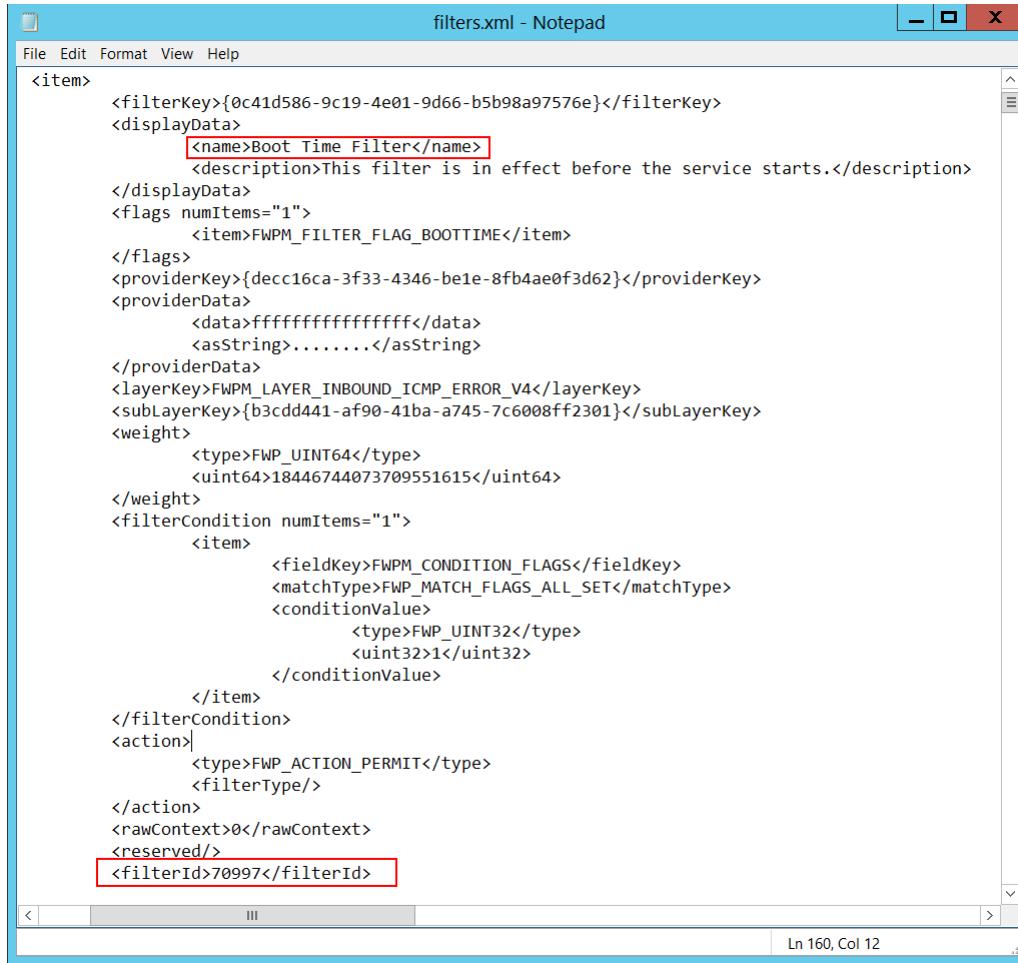
Service	Protocol Number
Internet Control Message Protocol (ICMP)	1
Transmission Control Protocol (TCP)	6
User Datagram Protocol (UDP)	17
General Routing Encapsulation (PPTP data over GRE)	47
Authentication Header (AH) IPSec	51
Encapsulation Security Payload (ESP) IPSec	50

Exterior Gateway Protocol (EGP)	8
Gateway-Gateway Protocol (GGP)	3
Host Monitoring Protocol (HMP)	20
Internet Group Management Protocol (IGMP)	88
MIT Remote Virtual Disk (RVD)	66
OSPF Open Shortest Path First	89
PARC Universal Packet Protocol (PUP)	12
Reliable Datagram Protocol (RDP)	27
Reservation Protocol (RSVP) QoS	46

Filter Information:

- **Filter Run-Time ID** [Type = UInt64]: unique filter ID which blocked the connection.

To find specific Windows Filtering Platform filter by ID you need to execute the following command: **netsh wfp show filters**. As result of this command **filters.xml** file will be generated. You need to open this file and find specific substring with required filter ID (**<filterId>**), for example:



The screenshot shows a Notepad window titled "filters.xml - Notepad". The XML code defines a filter named "Boot Time Filter" with provider key {decc16ca-3f33-4346-be1e-8fb4ae0f3d62} and sub-layer key {b3cdd441-af90-41ba-a745-7c6008ff2301}. The filter has a weight of 18446744073709551615 and uses FWP_CONDITION_FLAGS with a value of 1. It permits actions and has a reserved field with filter ID 70997.

```

<item>
    <filterKey>{0c41d586-9c19-4e01-9d66-b5b98a97576e}</filterKey>
    <displayData>
        <name>Boot Time Filter</name>
        <description>This filter is in effect before the service starts.</description>
    </displayData>
    <flags numItems="1">
        <item>FWPM_FILTER_FLAG_BOOTTIME</item>
    </flags>
    <providerKey>{decc16ca-3f33-4346-be1e-8fb4ae0f3d62}</providerKey>
    <providerData>
        <data>ffffffffffff</data>
        <asString>.....</asString>
    </providerData>
    <layerKey>FWPM_LAYER_INBOUND_ICMP_ERROR_V4</layerKey>
    <subLayerKey>{b3cdd441-af90-41ba-a745-7c6008ff2301}</subLayerKey>
    <weight>
        <type>FWP_UINT64</type>
        <uint64>18446744073709551615</uint64>
    </weight>
    <filterCondition numItems="1">
        <item>
            <fieldKey>FWPM_CONDITION_FLAGS</fieldKey>
            <matchType>FWP_MATCH_FLAGS_ALL_SET</matchType>
            <conditionValue>
                <type>FWP_UINT32</type>
                <uint32>1</uint32>
            </conditionValue>
        </item>
    </filterCondition>
    <action>
        <type>FWP_ACTION_PERMIT</type>
        <filterType/>
    </action>
    <rawContext>0</rawContext>
    <reserved/>
    <filterId>70997</filterId>

```

- **Layer Name** [Type = UnicodeString]: [Application Layer Enforcement](#) layer name.
- **Layer Run-Time ID** [Type = UInt64]: Windows Filtering Platform layer identifier. To find specific Windows Filtering Platform layer ID you need to execute the following command: **netsh wfp show state**. As result of this command **wfpstate.xml** file will be generated. You need to open this file and find specific substring with required layer ID (**<layerId>**), for example:

wfpstate.xml - Notepad

```

</item>
<item>
    <fieldKey>FWPM_CONDITION_INTERFACE_QUARANTINE_EPOCH</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_UINT64</dataType>
</item>
<item>
    <fieldKey>FWPM_CONDITION_ALE_PACKAGE_ID</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_SID</dataType>
</item>
</field>
<defaultSubLayerKey>FWPM_SUBLAYER_UNIVERSAL</defaultSubLayerKey>
<layerId>44</layerId>
</layer>
<callouts numItems="6">
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V4</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v4 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4</applicableLayer>
        <calloutId>13</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V6</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v6 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V6</applicableLayer>
        <calloutId>14</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V5</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v5 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V5</applicableLayer>
        <calloutId>15</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V7</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v7 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V7</applicableLayer>
        <calloutId>16</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V8</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v8 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V8</applicableLayer>
        <calloutId>17</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V9</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v9 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V9</applicableLayer>
        <calloutId>18</calloutId>
    </item>
</callouts>

```

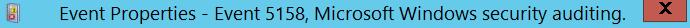
Security Monitoring Recommendations:

For 5157(F): The Windows Filtering Platform has blocked a connection.

- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”
- Check that “**Source Address**” is one of the addresses assigned to the computer.
- If the computer or device should not have access to the Internet, or contains only applications that don’t connect to the Internet, monitor for [5157](#) events where “**Destination Address**” is an IP address from the Internet (not from private IP ranges).
- If you know that the computer should never contact or be contacted by certain network IP addresses, monitor for these addresses in “**Destination Address**.”
- If you have a “whitelist” of IP addresses that the computer or device is expected to contact or be contacted by, monitor for IP addresses in “**Destination Address**” that are not in the whitelist.
- If you need to monitor all inbound connections to a specific local port, monitor for [5157](#) events with that “**Source Port**.”

- Monitor for all connections with a “**Protocol Number**” that is not typical for this device or computer, for example, anything other than 1, 6, or 17.
- If the computer’s communication with “**Destination Address**” should always use a specific “**Destination Port**,” monitor for any other “**Destination Port**.”

5158(S): The Windows Filtering Platform has permitted a bind to a local port.

 Event Properties - Event 5158, Microsoft Windows security auditing. X

General Details

Event Description:
This event generates every time [Windows Filtering Platform](#) permits an application or service to bind to a local port.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Application Information: Process ID: 4556 Application Name: \device\harddiskvolume2\documents\listener.exe	Event XML: <pre>- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>5158</EventID> <Version>0</Version> <Level>0</Level> <Task>12810</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-09-22T05:24:03.376171200Z" /> <EventRecordID>308122</EventRecordID> <Correlation /> <Execution ProcessID="4" ThreadID="3712" /> <Channel>Security</Channel> <Computer>DC01.contoso.local</Computer></pre>
Network Information: Source Address: 0.0.0.0 Source Port: 3333 Protocol: 6	
Filter Information: Filter Run-Time ID: 0 Layer Name: Resource Assignment Layer Run-Time ID: 36	
Log Name: Security Source: Microsoft Windows sec Event ID: 5158 Level: Information User: N/A OpCode: Info More Information: Event Log Online .	Copy Close

```
<Security />
</System>
- <EventData>
<Data Name="ProcessId">4556</Data>
<Data Name="Application">\device\harddiskvolume2\documents\listener.exe</Data>
<Data Name="SourceAddress">0.0.0.0</Data>
<Data Name="SourcePort">3333</Data>
<Data Name="Protocol">6</Data>
<Data Name="FilterRTID">0</Data>
<Data Name="LayerName">%14608</Data>
<Data Name="LayerRTID">36</Data>
</EventData>
</Event>
```

Required Server Roles: None.

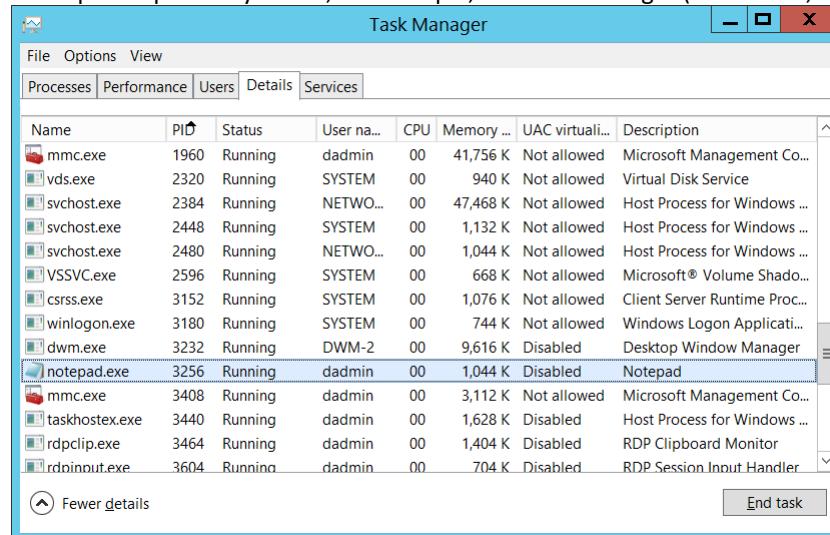
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Application Information:

- **Process ID [Type = Pointer]:** hexadecimal Process ID of the process which was permitted to bind to the local port. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



Name	PID	Status	User na...	CPU	Memory ...	UAC virtuall...	Description
mmc.exe	1960	Running	dadmin	00	41,756 K	Not allowed	Microsoft Management Co...
vds.exe	2320	Running	SYSTEM	00	940 K	Not allowed	Virtual Disk Service
svchost.exe	2384	Running	NETWO...	00	47,468 K	Not allowed	Host Process for Windows ...
svchost.exe	2448	Running	SYSTEM	00	1,132 K	Not allowed	Host Process for Windows ...
svchost.exe	2480	Running	NETWO...	00	1,044 K	Not allowed	Host Process for Windows ...
VSSVC.exe	2596	Running	SYSTEM	00	668 K	Not allowed	Microsoft® Volume Shado...
csrss.exe	3152	Running	SYSTEM	00	1,076 K	Not allowed	Client Server Runtime Proc...
winlogon.exe	3180	Running	SYSTEM	00	744 K	Not allowed	Windows Logon Applicati...
dwm.exe	3232	Running	DWM-2	00	9,616 K	Disabled	Desktop Window Manager
notepad.exe	3256	Running	dadmin	00	1,044 K	Disabled	Notepad
mmc.exe	3408	Running	dadmin	00	3,112 K	Not allowed	Microsoft Management Co...
taskhostex.exe	3440	Running	dadmin	00	1,628 K	Disabled	Host Process for Windows ...
rdpclip.exe	3464	Running	dadmin	00	1,404 K	Disabled	RDP Clipboard Monitor
rdpoinput.exe	3604	Running	dadmin	00	704 K	Disabled	RDP Session Input Handler

If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Application Name [Type = UnicodeString]:** full path and the name of the executable for the process.

Logical disk is displayed in format \device\harddiskvolume#. You can get all local volume numbers by using **diskpart** utility. The command to get volume numbers using diskpart is “**list volume**”:

```
C:\Windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (C) 1999-2013 Microsoft corporation.
On computer: DC01
DISKPART> list volume
Volume ### Ltr Label Fs Type Size Status Info
----- -- -- -- -- -- --
Volume 0 D System Rese NTFS DVD-ROM 0 B No Media
Volume 1 System Rese NTFS Partition 350 MB Healthy System
Volume 2 C NTFS Partition 126 GB Healthy Boot
```

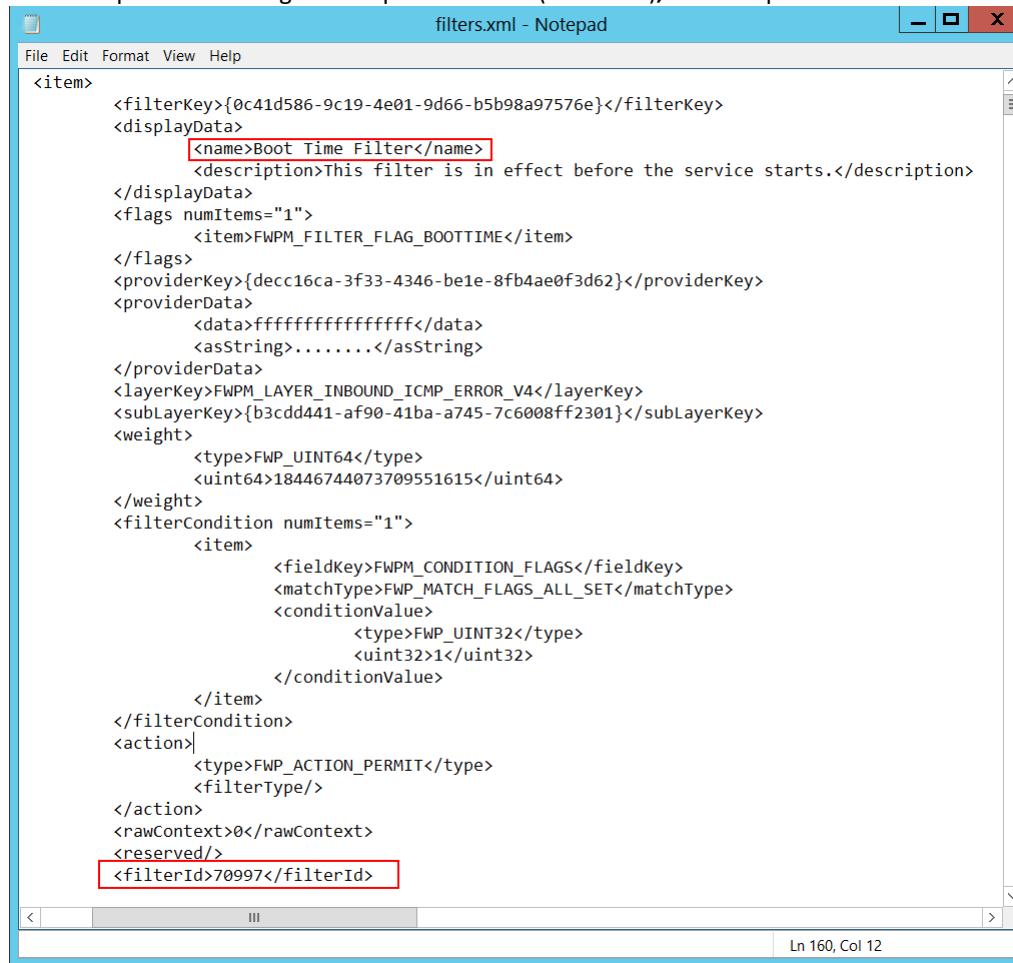
Network Information:

- **Source Address** [Type = UnicodeString]: local IP address on which application was bind the port.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format
 - 127.0.0.1 , ::1 - localhost
- **Source Port** [Type = UnicodeString]: port number which application was bind.
- **Protocol** [Type = UInt32]: number of protocol which was used.

Service	Protocol Number
Internet Control Message Protocol (ICMP)	1
Transmission Control Protocol (TCP)	6
User Datagram Protocol (UDP)	17
General Routing Encapsulation (PPTP data over GRE)	47
Authentication Header (AH) IPSec	51
Encapsulation Security Payload (ESP) IPSec	50
Exterior Gateway Protocol (EGP)	8
Gateway-Gateway Protocol (GGP)	3
Host Monitoring Protocol (HMP)	20
Internet Group Management Protocol (IGMP)	88
MIT Remote Virtual Disk (RVD)	66
OSPF Open Shortest Path First	89
PARC Universal Packet Protocol (PUP)	12
Reliable Datagram Protocol (RDP)	27
Reservation Protocol (RSVP) QoS	46

Filter Information:

- **Filter Run-Time ID** [Type = UInt64]: unique filter ID which allows application to bind the port. By default Windows firewall won't prevent a port from being binded by an application and if this application doesn't match any filters you will get value 0 in this field.
To find specific Windows Filtering Platform filter by ID you need to execute the following command: **netsh wfp show filters**. As result of this command **filters.xml** file will be generated. You need to open this file and find specific substring with required filter ID (**<filterId>**), for example:



The screenshot shows a Notepad window titled "filters.xml - Notepad". The XML code contains several highlighted sections: the filter key ("0c41d586-9c19-4e01-9d66-b5b98a97576e"), the filter name ("Boot Time Filter"), the provider key ("decc16ca-3f33-4346-be1e-8fb4ae0f3d62"), and the filter ID ("70997").

```

<item>
    <filterKey>{0c41d586-9c19-4e01-9d66-b5b98a97576e}</filterKey>
    <displayData>
        <name>Boot Time Filter</name>
        <description>This filter is in effect before the service starts.</description>
    </displayData>
    <flags numItems="1">
        <item>FWPM_FILTER_FLAG_BOOTTIME</item>
    </flags>
    <providerKey>{decc16ca-3f33-4346-be1e-8fb4ae0f3d62}</providerKey>
    <providerData>
        <data>ffffffffffff</data>
        <asString>.....</asString>
    </providerData>
    <layerKey>FWPM_LAYER_INBOUND_ICMP_ERROR_V4</layerKey>
    <subLayerKey>{b3cd441-af90-41ba-a745-7c6008ff2301}</subLayerKey>
    <weight>
        <type>FWP_UINT64</type>
        <uint64>18446744073709551615</uint64>
    </weight>
    <filterCondition numItems="1">
        <item>
            <fieldKey>FWPM_CONDITION_FLAGS</fieldKey>
            <matchType>FWP_MATCH_FLAGS_ALL_SET</matchType>
            <conditionValue>
                <type>FWP_UINT32</type>
                <uint32>1</uint32>
            </conditionValue>
        </item>
    </filterCondition>
    <action>
        <type>FWP_ACTION_PERMIT</type>
        <filterType/>
    </action>
    <rawContext>0</rawContext>
    <reserved/>
    <filterId>70997</filterId>

```

- **Layer Name** [Type = UnicodeString]: [Application Layer Enforcement](#) layer name.
- **Layer Run-Time ID** [Type = UInt64]: Windows Filtering Platform layer identifier. To find specific Windows Filtering Platform layer ID you need to execute the following command: **netsh wfp show state**. As result of this command **wfpstate.xml** file will be generated. You need to open this file and find specific substring with required layer ID (**<layerId>**), for example:

wfpstate.xml - Notepad

```

</item>
<item>
    <fieldKey>FWPM_CONDITION_INTERFACE_QUARANTINE_EPOCH</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_UINT64</dataType>
</item>
<item>
    <fieldKey>FWPM_CONDITION_ALE_PACKAGE_ID</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_SID</dataType>
</item>
</field>
<defaultSubLayerKey>FWPM_SUBLAYER_UNIVERSAL</defaultSubLayerKey>
<layerId>44</layerId>
</layer>
<callouts numItems="6">
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V4</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v4 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4</applicableLayer>
        <calloutId>13</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V6</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v6 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V6</applicableLayer>
        <calloutId>14</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V5</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v5 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V5</applicableLayer>
        <calloutId>15</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V7</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v7 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V7</applicableLayer>
        <calloutId>16</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V8</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v8 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V8</applicableLayer>
        <calloutId>17</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V9</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v9 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V9</applicableLayer>
        <calloutId>18</calloutId>
    </item>
</callouts>

```

Ln 4584, Col 37

Security Monitoring Recommendations:

For 5158(S): The Windows Filtering Platform has permitted a bind to a local port.

- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”
- Check that “**Source Address**” is one of the addresses assigned to the computer.
- If you need to monitor all actions with a specific local port, monitor for **5158** events with that “**Source Port**.”
- Monitor for all connections with a “**Protocol Number**” that is not typical for this device or computer, for example, anything other than 6 or 17.
- If the computer’s communication with “**Destination Address**” should always use a specific “**Destination Port**,” monitor for any other “**Destination Port**.”

5159(F): The Windows Filtering Platform has blocked a bind to a local port.

This event is logged if the Windows Filtering Platform has blocked a bind to a local port.

There is no example of this event in this document.

Event Schema:

The Windows Filtering Platform has blocked a bind to a local port.

Application Information:

Process ID:%1

Application Name:%2

Network Information:

Source Address:%3

Source Port:%4

Protocol:%5

Filter Information:

Filter Run-Time ID:%6

Layer Name:%7

Layer Run-Time ID:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

Audit Filtering Platform Packet Drop

Audit Filtering Platform Packet Drop determines whether the operating system generates audit events when packets are dropped by the [Windows Filtering Platform](#).

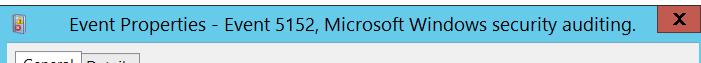
Windows Filtering Platform (WFP) enables independent software vendors (ISVs) to filter and modify TCP/IP packets, monitor or authorize connections, filter Internet Protocol security (IPSec)-protected traffic, and filter remote procedure calls (RPCs).

A high rate of dropped packets may indicate that there have been attempts to gain unauthorized access to computers on your network.

Event volume: High.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	No	No	No	No	Failure events volume typically is very high for this subcategory and typically used for troubleshooting. If you need to monitor blocked connections, it is better to use “ 5157(F) : The Windows Filtering Platform has blocked a connection,” because it contains almost the same information and generates per-connection, not per-packet. There is no recommendation to enable Success auditing, because Success events in this subcategory rarely occur.
Member Server	No	No	No	No	Failure events volume typically is very high for this subcategory and typically used for troubleshooting. If you need to monitor blocked connections, it is better to use “ 5157(F) : The Windows Filtering Platform has blocked a connection,” because it contains almost the same information and generates per-connection, not per-packet. There is no recommendation to enable Success auditing, because Success events in this subcategory rarely occur.
Workstation	No	No	No	No	Failure events volume typically is very high for this subcategory and typically used for troubleshooting. If you need to monitor blocked connections, it is better to use “ 5157(F) : The Windows Filtering Platform has blocked a connection,” because it contains almost the same information and generates per-connection, not per-packet. There is no recommendation to enable Success auditing, because Success events in this subcategory rarely occur.

 Event Properties - Event 5152, Microsoft Windows security auditing.

[General](#) [Details](#)

The Windows Filtering Platform has blocked a packet.

Application Information:

- Process ID: 4556
- Application Name: \device\harddiskvolume2\documents\listener.exe

Network Information:

- Direction: Inbound

Destination Port: 3333
Protocol: 6

Filter Information:

- Filter Run-Time ID: 0
- Layer Name: Receive/Accept
- Layer Run-Time ID: 44

Log Name: Security
Source: Microsoft Windows sec
Event ID: 5152
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 9/22/2015 9:52:37 AM
Task Category: Filtering Platform Packet Drop
Keywords: Audit Failure
Computer: DC01.contoso.local

[Copy](#) [Close](#)

Events List:

- [5152\(F\)](#): The Windows Filtering Platform blocked a packet.
- [5153\(S\)](#): A more restrictive Windows Filtering Platform filter has blocked a packet.

5152(F): The Windows Filtering Platform blocked a packet.

Event Description:

This event generates when [Windows Filtering Platform](#) has blocked a network packet.
This event is generated for every received network packet.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5152</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12809</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-09-22T16:52:37.274367300Z" />
```

```
<EventRecordID>321323</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="4456" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ProcessId">4556</Data>
<Data Name="Application">\device\harddiskvolume2\documents\listener.exe</Data>
<Data Name="Direction">%%14592</Data>
<Data Name="SourceAddress">10.0.0.100</Data>
<Data Name="SourcePort">49278</Data>
<Data Name="DestAddress">10.0.0.10</Data>
<Data Name="DestPort">3333</Data>
<Data Name="Protocol">6</Data>
<Data Name="FilterRTID">0</Data>
<Data Name="LayerName">%%14610</Data>
<Data Name="LayerRTID">44</Data>
</EventData>
</Event>
```

Required Server Roles: None.

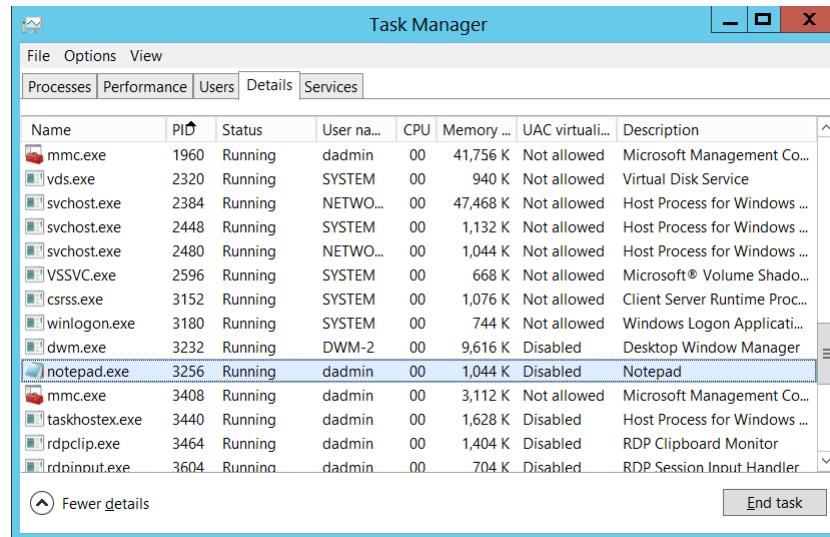
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Application Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process to which blocked network packet was sent. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Application Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Logical disk is displayed in format \device\harddiskvolume#. You can get all local volume numbers by using **diskpart** utility. The command to get volume numbers using diskpart is “list volume”:

```
C:\Windows\system32>diskpart
Microsoft DiskPart version 6.3.9600
Copyright (C) 1999-2013 Microsoft Corporation.
On computer: DC01

DISKPART> list volume

  volume ###  Ltr  Label        Fs  Type     size    status     Info
  -----  --  --  -----  -----  -----  -----  -----
  Volume 0      D                DVD-ROM   0 B  No Media
  Volume 1      System Rese  NTFS  Partition 350 MB  Healthy  System
  Volume 2      C                NTFS  Partition 126 GB  Healthy  Boot
```

Network Information:

- **Direction** [Type = UnicodeString]: direction of blocked connection.
 - Inbound – for inbound connections.
 - Outbound – for outbound connections.
- **Source Address** [Type = UnicodeString]: local IP address on which application received the packet.
 - IPv4 Address

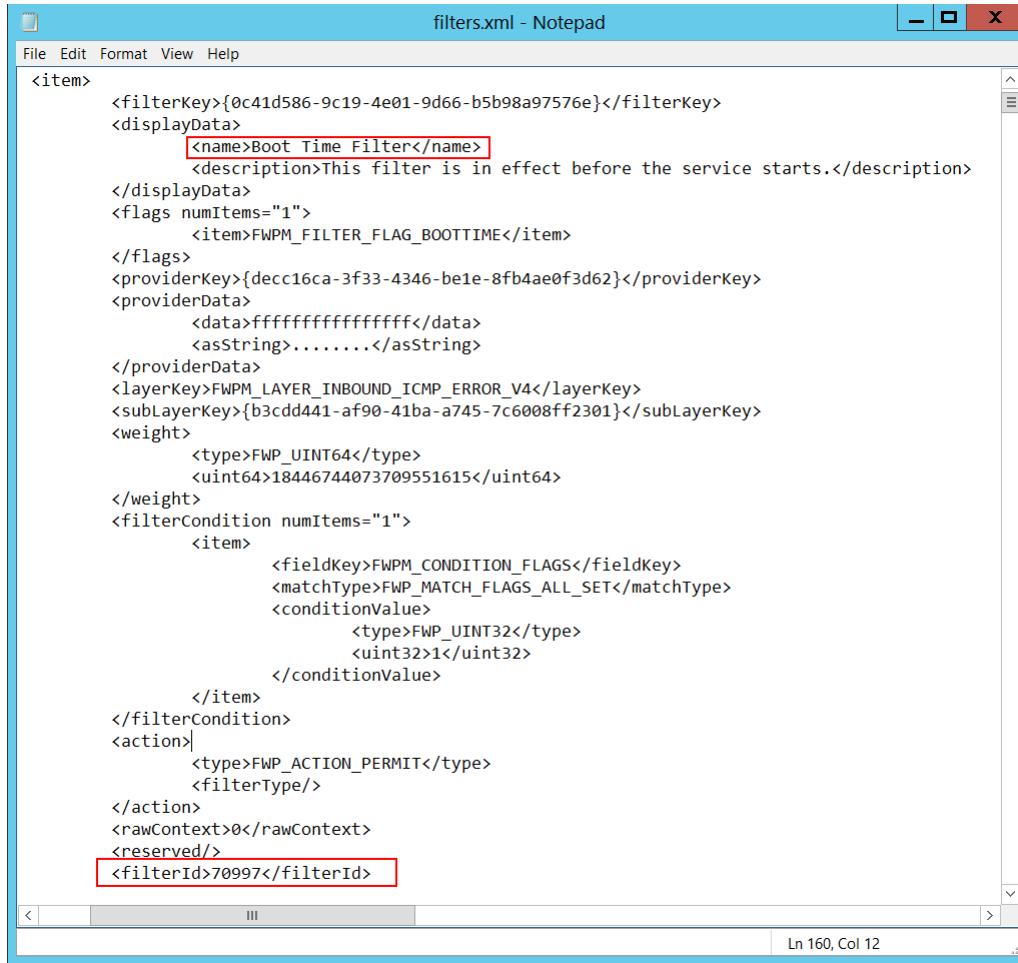
- IPv6 Address
- :: - all IP addresses in IPv6 format
- 0.0.0.0 - all IP addresses in IPv4 format
- 127.0.0.1 , ::1 - localhost
- **Source Port** [Type = UnicodeString]: port number on which application received the packet.
- **Destination Address** [Type = UnicodeString]: IP address from which packet was received or initiated.
 - IPv4 Address
 - IPv6 Address
 - :: - all IP addresses in IPv6 format
 - 0.0.0.0 - all IP addresses in IPv4 format
 - 127.0.0.1 , ::1 - localhost
- **Destination Port** [Type = UnicodeString]: port number which was used from remote machine to send the packet.
- **Protocol** [Type = UInt32]: number of protocol which was used.

Service	Protocol Number
Internet Control Message Protocol (ICMP)	1
Transmission Control Protocol (TCP)	6
User Datagram Protocol (UDP)	17
General Routing Encapsulation (PPTP data over GRE)	47
Authentication Header (AH) IPSec	51
Encapsulation Security Payload (ESP) IPSec	50
Exterior Gateway Protocol (EGP)	8
Gateway-Gateway Protocol (GGP)	3
Host Monitoring Protocol (HMP)	20
Internet Group Management Protocol (IGMP)	88
MIT Remote Virtual Disk (RVD)	66
OSPF Open Shortest Path First	89
PARC Universal Packet Protocol (PUP)	12
Reliable Datagram Protocol (RDP)	27
Reservation Protocol (RSVP) QoS	46

Filter Information:

- **Filter Run-Time ID** [Type = UInt64]: unique filter ID which blocked the packet.

To find specific Windows Filtering Platform filter by ID you need to execute the following command: **netsh wfp show filters**. As result of this command **filters.xml** file will be generated. You need to open this file and find specific substring with required filter ID (**<filterId>**), for example:



The screenshot shows a Notepad window titled "filters.xml - Notepad". The XML code defines a filter named "Boot Time Filter" with provider key {decc16ca-3f33-4346-be1e-8fb4ae0f3d62} and sub-layer key {b3cdd441-af90-41ba-a745-7c6008ff2301}. The filter has a weight of 18446744073709551615 and a condition that matches all flags. It permits traffic and has a layer ID of 70997.

```

<item>
    <filterKey>{0c41d586-9c19-4e01-9d66-b5b98a97576e}</filterKey>
    <displayData>
        <name>Boot Time Filter</name>
        <description>This filter is in effect before the service starts.</description>
    </displayData>
    <flags numItems="1">
        <item>FWPM_FILTER_FLAG_BOOTTIME</item>
    </flags>
    <providerKey>{decc16ca-3f33-4346-be1e-8fb4ae0f3d62}</providerKey>
    <providerData>
        <data>ffffffffffff</data>
        <asString>.....</asString>
    </providerData>
    <layerKey>FWPM_LAYER_INBOUND_ICMP_ERROR_V4</layerKey>
    <subLayerKey>{b3cdd441-af90-41ba-a745-7c6008ff2301}</subLayerKey>
    <weight>
        <type>FWP_UINT64</type>
        <uint64>18446744073709551615</uint64>
    </weight>
    <filterCondition numItems="1">
        <item>
            <fieldKey>FWPM_CONDITION_FLAGS</fieldKey>
            <matchType>FWP_MATCH_FLAGS_ALL_SET</matchType>
            <conditionValue>
                <type>FWP_UINT32</type>
                <uint32>1</uint32>
            </conditionValue>
        </item>
    </filterCondition>
    <action>
        <type>FWP_ACTION_PERMIT</type>
        <filterType/>
    </action>
    <rawContext>0</rawContext>
    <reserved/>
    <filterId>70997</filterId>

```

- **Layer Name** [Type = UnicodeString]: [Application Layer Enforcement](#) layer name.
- **Layer Run-Time ID** [Type = UInt64]: Windows Filtering Platform layer identifier. To find specific Windows Filtering Platform layer ID you need to execute the following command: **netsh wfp show state**. As result of this command **wfpstate.xml** file will be generated. You need to open this file and find specific substring with required layer ID (**<layerId>**), for example:

wfpstate.xml - Notepad

```

</item>
<item>
    <fieldKey>FWPM_CONDITION_INTERFACE_QUARANTINE_EPOCH</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_UINT64</dataType>
</item>
<item>
    <fieldKey>FWPM_CONDITION_ALE_PACKAGE_ID</fieldKey>
    <type>FWPM_FIELD_RAW_DATA</type>
    <dataType>FWP_SID</dataType>
</item>
</field>
<defaultSubLayerKey>FWPM_SUBLAYER_UNIVERSAL</defaultSubLayerKey>
<layerId>44</layerId>
</layer>
<callouts numItems="6">
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V4</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v4 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V4</applicableLayer>
        <calloutId>13</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V6</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v6 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V6</applicableLayer>
        <calloutId>14</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V5</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v5 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V5</applicableLayer>
        <calloutId>15</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V7</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v7 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V7</applicableLayer>
        <calloutId>16</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V8</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v8 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V8</applicableLayer>
        <calloutId>17</calloutId>
    </item>
    <item>
        <calloutKey>FWPM_CALLOUT_IPSEC_INBOUND_INITIATE_SECURE_V9</calloutKey>
        <displayData>
            <name>WFP Built-in IPsec Inbound Initiate Secure v9 Layer Callout</name>
            <description>Verifies that each incoming connection that is supposed to arrive secure arrives securely.</description>
        </displayData>
        <flags numItems="1">
            <item>FWPM_CALLOUT_FLAG_REGISTERED</item>
        </flags>
        <providerKey/>
        <providerData/>
        <applicableLayer>FWPM_LAYER_ALE_AUTH_RECV_ACCEPT_V9</applicableLayer>
        <calloutId>18</calloutId>
    </item>
</callouts>

```

Ln 4584, Col 37

Security Monitoring Recommendations:

For 5152(F): The Windows Filtering Platform blocked a packet.

- If you have a pre-defined application which should be used to perform the operation that was reported by this event, monitor events with “**Application**” not equal to your defined application.
- You can monitor to see if “**Application**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in application names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Application**.”
- Check that **Source Address** is one of the addresses assigned to the computer.
- If the computer or device should not have access to the Internet, or contains only applications that don’t connect to the Internet, monitor for [5152](#) events where **Destination Address** is an IP address from the Internet (not from private IP ranges).
- If you know that the computer should never contact or be contacted by certain network IP addresses, monitor for these addresses in “**Destination Address**.”
- If you have a “whitelist” of IP addresses that the computer or device is expected to contact or be contacted by, monitor for IP addresses in “**Destination Address**” that are not in the whitelist.
- If you need to monitor all inbound connections to a specific local port, monitor for [5152](#) events with that “**Source Port**.”

- Monitor for all connections with a “**Protocol Number**” that is not typical for this device or computer, for example, anything other than 1, 6, or 17.
- If the computer’s communication with “**Destination Address**” should always use a specific “**Destination Port**,” monitor for any other “**Destination Port**.”

5153(S): A more restrictive Windows Filtering Platform filter has blocked a packet.

This event is logged if a more restrictive Windows Filtering Platform filter has blocked a packet.

There is no example of this event in this document.

Event Schema:

A more restrictive Windows Filtering Platform filter has blocked a packet.

Application Information:

Process ID:%1

Application Name:%2

Network Information:

Source Address:%3

Source Port:%4

Protocol:%5

Filter Information:

Filter Run-Time ID:%6

Layer Name:%7

Layer Run-Time ID:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

Audit Handle Manipulation

Audit Handle Manipulation enables generation of “4658: The handle to an object was closed” in [Audit File System](#), [Audit Kernel Object](#), [Audit Registry](#), [Audit Removable Storage](#) and [Audit SAM](#) subcategories, and shows object’s handle duplication and close actions.

Event volume: High.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	

Domain Controller	No	No	No	No	Typically, information about the duplication or closing of an object handle has little to no security relevance and is hard to parse or analyze. There is no recommendation to enable this subcategory for Success or Failure auditing, unless you know exactly what you need to monitor in Object's Handles level.
Member Server	No	No	No	No	Typically, information about the duplication or closing of an object handle has little to no security relevance and is hard to parse or analyze. There is no recommendation to enable this subcategory for Success or Failure auditing, unless you know exactly what you need to monitor in Object's Handles level.
Workstation	No	No	No	No	Typically, information about the duplication or closing of an object handle has little to no security relevance and is hard to parse or analyze. There is no recommendation to enable this subcategory for Success or Failure auditing, unless you know exactly what you need to monitor in Object's Handles level.

Events List:

- [4658\(S\)](#): The handle to an object was closed.
- [4690\(S\)](#): An attempt was made to duplicate a handle to an object.

4658(S): The handle to an object was closed.

Event Properties - Event 4690, Microsoft Windows security audit... X

General Details

An attempt was made to duplicate a handle to an object.

Subject: Security ID: SYSTEM
Account Name: DC01\$

This event doesn't generate in this subcategory, but you can use this subcategory to enable it. For a description of the event, see "[4658\(S\)](#): The handle to an object was closed" in the Audit File System subcategory.

4690(S): An attempt was made to duplicate a handle to an object.

Event Description:

This event generates if an attempt was made to duplicate a handle to an object.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Source Handle Information:
Source Handle ID:0x438
Source Process ID: 0x674

New Handle Information:
Target Handle ID:0xd9c
Target Process ID:0x4

Log Name: Security
Source: Microsoft Windows sec Logged: 9/22/2015 5:17:41 PM
Event ID: 4690 Task Category: Handle Manipulation
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4690</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12807</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T00:17:41.755998800Z" />
```

```
<EventRecordID>338632</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="1100" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="SourceHandleId">0x438</Data>
<Data Name="SourceProcessId">0x674</Data>
<Data Name="TargetHandleId">0xd9c</Data>
<Data Name="TargetProcessId">0x4</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made an attempt to duplicate a handle to an object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

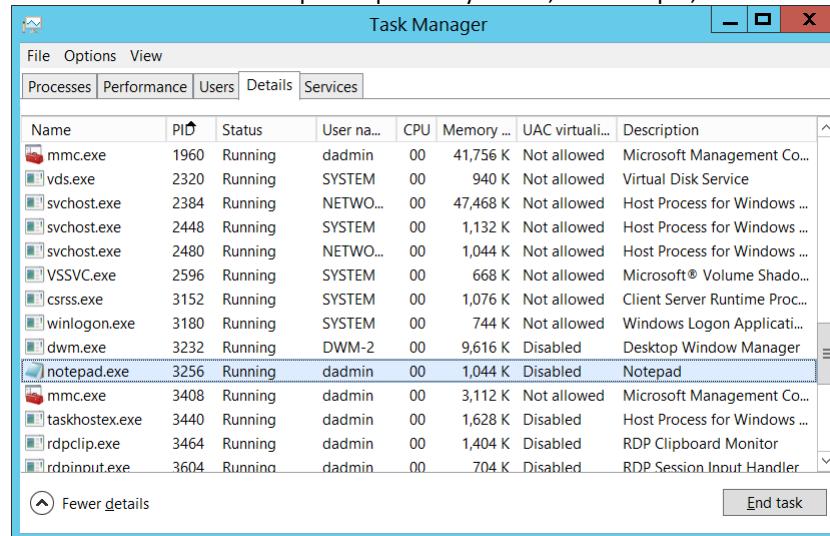
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to duplicate a handle to an object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624: An account was successfully logged on.](#)”

Source Handle Information:

- **Source Handle ID** [Type = Pointer]: hexadecimal value of a handle which was duplicated. This field can help you correlate this event with other events, for example “4663: An attempt was made to access an object” in [Audit File System](#), [Audit Kernel Object](#), [Audit Registry](#), [Audit Removable Storage](#) or [Audit SAM](#) subcategories.
- **Source Process ID** [Type = Pointer]: hexadecimal Process ID of the process which opened the **Source Handle ID** before it was duplicated. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

New Handle Information:

- **Target Handle ID** [Type = Pointer]: hexadecimal value of the new handle (the copy of **Source Handle ID**). This field can help you correlate this event with other events, for example “4663: An attempt was made to access an object” in [Audit File System](#), [Audit Kernel Object](#), [Audit Registry](#), [Audit Removable Storage](#) or [Audit SAM](#) subcategories.
- **Target Process ID** [Type = Pointer]: hexadecimal Process ID of the process which opened the **Target Handle ID**. Process ID (PID) is a number used by the operating system to uniquely identify an active process. You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID** field.

Security Monitoring Recommendations:

For 4690(S): An attempt was made to duplicate a handle to an object.

- Typically this event has little to no security relevance and is hard to parse or analyze. There is no recommendation for this event, unless you know exactly what you need to monitor with it.
- This event can be used to track all actions or operations related to a specific object handle.

Audit Kernel Object

Audit Kernel Object determines whether the operating system generates audit events when users attempt to access the system kernel, which includes mutexes and semaphores.

Only kernel objects with a matching system access control list ([SACL](#)) generate security audit events. The audits generated are usually useful only to developers.

Typically, kernel objects are given SACLs only if the AuditBaseObjects or AuditBaseDirectories auditing options are enabled.

The "[Audit: Audit the access of global system objects](#)" policy setting controls the default SACL of kernel objects.

Event volume: High.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	No	No	Typically Kernel object auditing events have little to no security relevance and are hard to parse or analyze. Also, the volume of these events is typically very high. There is no recommendation to enable this subcategory, unless you know exactly what you need to monitor at the Kernel objects level.
Member Server	No	No	No	No	Typically Kernel object auditing events have little to no security relevance and are hard to parse or analyze. Also, the volume of these events is typically very high. There is no recommendation to enable this subcategory, unless you know exactly what you need to monitor at the Kernel objects level.
Workstation	No	No	No	No	Typically Kernel object auditing events have little to no security relevance and are hard to parse or analyze. Also, the volume of these events is typically very high. There is no recommendation to enable this subcategory, unless you know exactly what you need to monitor at the Kernel objects level.

Events List:

- [4656](#)(S, F): A handle to an object was requested.
- [4658](#)(S): The handle to an object was closed.
- [4660](#)(S): An object was deleted.
- [4663](#)(S): An attempt was made to access an object.

4656(S, F): A handle to an object was requested.

This event also belongs in the Audit File System subcategory, and is described there. See "[4656](#)(S, F): A handle to an object was requested."

4658(S): The handle to an object was closed.

This event also belongs in the Audit File System subcategory, and is described there. See "[4658](#)(S): The handle to an object was closed."

4660(S): An object was deleted.

This event also belongs in the Audit File System subcategory, and is described there. See "[4660](#)(S): An object was deleted."

4663(S): An attempt was made to access an object.

This event also belongs in the Audit File System subcategory, and is described there. See "[4663\(S\): An attempt was made to access an object.](#)"

Audit Other Object Access Events

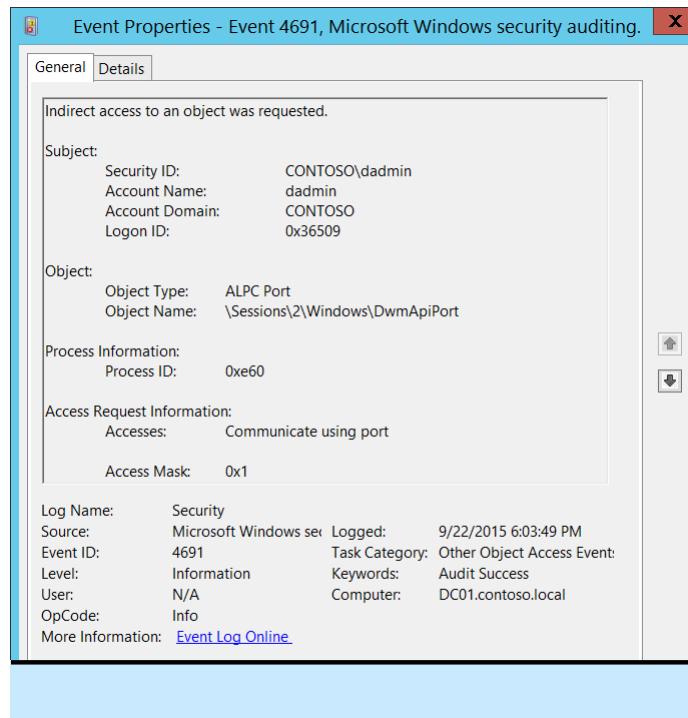
Audit Other Object Access Events allows you to monitor operations with scheduled tasks, COM+ objects and indirect object access requests.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	We recommend Success auditing first of all because of scheduled tasks events. We recommend Failure auditing to get events about possible ICPM DoS attack.
Member Server	Yes	Yes	Yes	Yes	We recommend Success auditing first of all because of scheduled tasks events. We recommend Failure auditing to get events about possible ICPM DoS attack.
Workstation	Yes	Yes	Yes	Yes	We recommend Success auditing first of all because of scheduled tasks events. We recommend Failure auditing to get events about possible ICPM DoS attack.

Events List:

- [4671\(-\)](#): An application attempted to access a blocked ordinal through the TBS.

 Event Properties - Event 4691, Microsoft Windows security auditing.

General Details

Indirect access to an object was requested.

Subject:

- Security ID: CONTOSO\damain
- Account Name: damain
- Account Domain: CONTOSO
- Logon ID: 0x36509

Object:

- Object Type: ALPC Port
- Object Name: \Sessions\2\Windows\DwmApiPort

Process Information:

- Process ID: 0xe60

Access Request Information:

- Accesses: Communicate using port
- Access Mask: 0x1

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4691
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

- [4691\(S\)](#): Indirect access to an object was requested.
- [5148\(F\)](#): The Windows Filtering Platform has detected a DoS attack and entered a defensive mode; packets associated with this attack will be discarded.
- [5149\(F\)](#): The DoS attack has subsided and normal processing is being resumed.
- [4698\(S\)](#): A scheduled task was created.
- [4699\(S\)](#): A scheduled task was deleted.
- [4700\(S\)](#): A scheduled task was enabled.
- [4701\(S\)](#): A scheduled task was disabled.
- [4702\(S\)](#): A scheduled task was updated.
- [5888\(S\)](#): An object in the COM+ Catalog was modified.
- [5889\(S\)](#): An object was deleted from the COM+ Catalog.
- [5890\(S\)](#): An object was added to the COM+ Catalog.

4671(-): An application attempted to access a blocked ordinal through the TBS.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system.

4691(S): Indirect access to an object was requested.

Event Description:

This event indicates that indirect access to an object was requested.
These events are generated for [ALPC Ports](#) access request actions.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4691</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12804</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T01:03:49.834912100Z" />
<EventRecordID>344382</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="2928" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x36509</Data>
<Data Name="ObjectType">ALPC Port</Data>
<Data Name="ObjectName">\Sessions\2\Windows\DsPort</Data>
<Data Name="AccessList">%4464</Data>
<Data Name="AccessMask">0x1</Data>
<Data Name="ProcessId">0xe60</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested an access to the object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested an access to the object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Type** [Type = UnicodeString]: The type of an object for which access was requested.

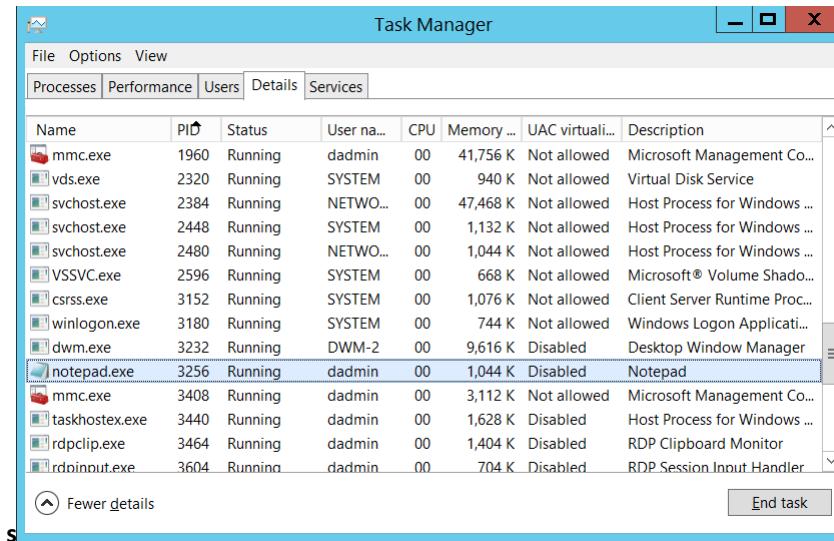
The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: full path and name of the object for which access was requested.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the access was requested. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

Access Request Information:

- **Accesses** [Type = UnicodeString]: the list of access rights which were requested by **Subject\Security ID**. These access rights depend on **Object Type**. “Table 13. File access codes.” contains information about the most common access rights for file system objects. For information about ALPC ports access rights, use <https://technet.microsoft.com/> or other informational resources.
- **Access Mask** [Type = HexInt32]: hexadecimal mask for the operation that was requested or performed. See “Table 13. File access codes.” for more information about file access rights. For information about ALPC ports access rights, use <https://technet.microsoft.com/> or other informational resources.

Security Monitoring Recommendations:

For 4691(S): Indirect access to an object was requested.

- Typically this event has little to no security relevance and is hard to parse or analyze. There is no recommendation for this event, unless you know exactly what you need to monitor with ALPC Ports.

5148(F): The Windows Filtering Platform has detected a DoS attack and entered a defensive mode; packets associated with this attack will be discarded.

In most circumstances, this event occurs very rarely. It is designed to be generated when an ICPM DoS attack starts or was detected.

There is no example of this event in this document.

Event Schema:

The Windows Filtering Platform has detected a DoS attack and entered a defensive mode; packets associated with this attack will be discarded.

Network Information:

Type:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of ICMP DoS attack or, among other things, hardware or network device related problems. In both cases, we recommend triggering an alert and investigating the reason the event was generated.

5149(F): The DoS attack has subsided and normal processing is being resumed.

In most circumstances, this event occurs very rarely. It is designed to be generated when an ICMP DoS attack ended.

There is no example of this event in this document.

Event Schema:

The DoS attack has subsided and normal processing is being resumed.

Network Information:

Type:%1

Packets Discarded:%2

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of ICMP DoS attack or, among other things, hardware or network device related problems. In both cases, we recommend triggering an alert and investigating the reason the event was generated.

4698(S): A scheduled task was created.

Event Properties - Event 4698, Microsoft Windows security auditing. X

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x364EB

Task Information:

Task Name:	\Microsoft\StartListener
Task Content:	<?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo> <Date>2015-09-22T19:03:06.9258653</Date> <Author>CONTOSO\dadmin</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author"> <RunLevel>LeastPrivilege</RunLevel> <UserId>CONTOSO\dadmin</UserId> <LogonType>InteractiveToken</LogonType> </Principal> </Principals> <Settings> <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries> <AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings> <StopOnIdleEnd>true</StopOnIdleEnd> <RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>true</Enabled> <Hidden>false</Hidden> <RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings> <Actions Context="Author"> <Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task>

Event Description:

This event generates every time a new scheduled task is created.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4698</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12804</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-23T02:03:06.944522200Z" />
  <EventRecordID>344740</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="5048" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x364eb</Data>
  <Data Name="TaskName">\Microsoft\StartListener</Data>
  <Data Name="TaskContent"><?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
<RegistrationInfo>
<Date>2015-09-22T19:03:06.9258653</Date>
<Author>CONTOSO\dadmin</Author>
</RegistrationInfo>
<Triggers />
<Principals>
<Principal id="Author">
<RunLevel>LeastPrivilege</RunLevel>
<UserId>CONTOSO\dadmin</UserId>
<LogonType>InteractiveToken</LogonType>
</Principal>
</Principals>
<Settings>
<MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
<DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries>
<StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
<AllowHardTerminate>true</AllowHardTerminate>
<StartWhenAvailable>false</StartWhenAvailable>
<RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
<IdleSettings>
<StopOnIdleEnd>true</StopOnIdleEnd>
<RestartOnIdle>false</RestartOnIdle>
</IdleSettings>
<AllowStartOnDemand>true</AllowStartOnDemand>
<Enabled>true</Enabled>
<Hidden>false</Hidden>
<RunOnlyIfIdle>false</RunOnlyIfIdle>
<WakeToRun>false</WakeToRun>
<ExecutionTimeLimit>P3D</ExecutionTimeLimit>
<Priority>7</Priority>
</Settings>
<Actions Context="Author">
<Exec>
<Command>C:\Documents\listener.exe</Command>
</Exec>
</Actions>
</Task>

```

<DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries> <AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings> <StopOnIdleEnd>true</StopOnIdleEnd>

```
<RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>true</Enabled> <Hidden>false</Hidden>
<RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings> <Actions Context="Author">
<Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

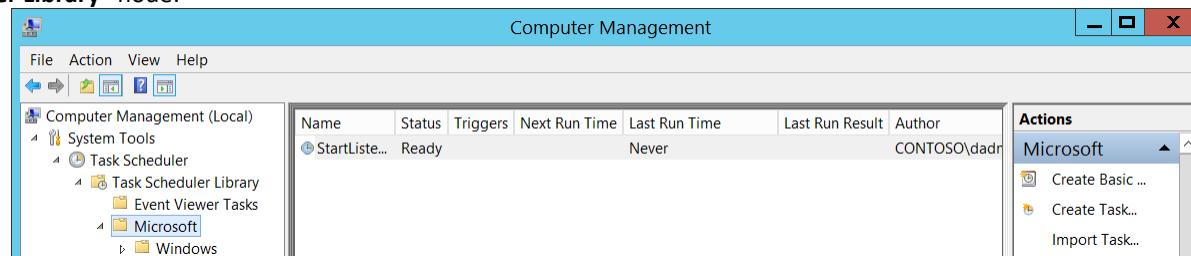
- **Security ID** [Type = SID]: SID of account that requested the “create scheduled task” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create scheduled task” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Task Information:

- **Task Name** [Type = UnicodeString]: new scheduled task name. The format of this value is “[task_path](#)\task_name”, where **task_path** is a path in Microsoft **Task Scheduler** tree starting from “[Task Scheduler Library](#)” node:



- **Task Content** [Type = UnicodeString]: the [XML](#) content of the new task. For more information about the XML format for scheduled tasks, see “[XML Task Definition Format](#).”

Security Monitoring Recommendations:

For 4698(S): A scheduled task was created.

[Appendix A: Security monitoring recommendations for many audit events](#)



- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. We recommend monitoring all scheduled task creation events, especially on critical computers or devices. Scheduled tasks are often used by malware to stay in the system after reboot or for other malicious actions.
- Monitor for new tasks located in the **Task Scheduler Library** root node, that is, where **Task Name** looks like '\TASK_NAME'. Scheduled tasks that are created manually or by malware are often located in the **Task Scheduler Library** root node.
- In the new task, if the **Task Content: XML** contains <LogonType>Password</LogonType> value, trigger an alert. In this case, the password for the account that will be used to run the scheduled task will be saved in Credential Manager in cleartext format, and can be extracted using Administrative privileges.

4699(S): A scheduled task was deleted.

Event Description:

This event generates every time a scheduled task was deleted.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4699</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12804</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-23T02:13:30.044244500Z" />
  <EventRecordID>344827</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="5048" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />

```

```
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x364eb</Data>
<Data Name="TaskName">\Microsoft\My</Data>
<Data Name="TaskContent"><?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo>
<Date>2015-08-25T13:56:10.5315552</Date> <Author>CONTOSO\dadmin</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author">
<RunLevel>LeastPrivilege</RunLevel> <UserId>CONTOSO\dadmin</UserId> <LogonType>Password</LogonType> </Principal> </Principals> <Settings>
<MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>false</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
<AllowHardTerminate>false</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings>
<StopOnIdleEnd>true</StopOnIdleEnd> <RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>true</Enabled>
<Hidden>false</Hidden> <RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>PT0S</ExecutionTimeLimit> <Priority>7</Priority> </Settings>
<Actions Context="Author"> <Exec> <Command>C:\Windows\notepad.exe</Command> </Exec> </Actions> </Task></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete scheduled task” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete scheduled task” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Task Information:

- **Task Name** [Type = UnicodeString]: deleted scheduled task name. The format of this value is “\task_path\task_name”, where **task_path** is a path in Microsoft **Task Scheduler** tree starting from “**Task Scheduler Library**” node:

Event Properties - Event 4700, Microsoft Windows security auditing.

General **Details**

A scheduled task was enabled.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x364EB

Task Information:

Task Name:	\Microsoft\StartListener
Task Content:	<?xml version="1.0" encoding="UTF-16"?><Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"><Registration><Triggers /><Principals><Principal id="Author"><RunLevel>LeastPrivilege</RunLevel><UserId>CONTOSO\dadmin</UserId><LogonType>InteractiveToken</LogonType></Principal></Principals><Settings><MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy><DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries><StopIfGoingOnBatteries>true</StopIfGoingOnBatteries><AllowHardTerminate>true</AllowHardTerminate><StartWhenAvailable>false</StartWhenAvailable><RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable><IdleSettings><StopOnIdleEnd>true</StopOnIdleEnd><RestartOnIdle>false</RestartOnIdle></IdleSettings><AllowStartOnDemand>true</AllowStartOnDemand><Enabled>true</Enabled>

Event XML:

```

<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
<System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4700</EventID>

```

Computer Management

File Action View Help

Computer Management (Local)

- System Tools
 - Task Scheduler
 - Task Scheduler Library
 - Event Viewer Tasks
 - Microsoft
 - Windows

Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result	Author
StartLister...	Ready			Never		CONTOSO\dadmin

Actions

Microsoft

- Create Basic ...
- Create Task...
- Import Task...

- **Task Content** [Type = UnicodeString]: the [XML](#) of the deleted task. Here “[XML Task Definition Format](#)” you can read more about the XML format for scheduled tasks.

Security Monitoring Recommendations:

For 4699(S): A scheduled task was deleted.

Appendix A: Security monitoring recommendations for many audit events

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. We recommend monitoring all scheduled task deletion events, especially on critical computers or devices. Scheduled tasks are often used by malware to stay in the system after reboot or for other malicious actions. However, this event does not often happen.
- Monitor for deleted tasks located in the **Task Scheduler Library** root node, that is, where **Task Name** looks like ‘\TASK_NAME’. Scheduled tasks that are created manually or by malware are often located in the **Task Scheduler Library** root node. Deletion of such tasks can be a sign of malicious activity.
- If a highly critical scheduled task exists on some computers, and it should never be deleted, monitor for [4699](#) events with the corresponding **Task Name**.

4700(S): A scheduled task was enabled.

Event Description:

This event generates every time a scheduled task is enabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
<System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4700</EventID>

```

```
<Version>0</Version>
<Level>0</Level>
<Task>12804</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T02:32:47.606423000Z" />
<EventRecordID>344861</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="756" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x364eb</Data>
<Data Name="TaskName">\Microsoft\StartListener</Data>
<Data Name="TaskContent"><?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo>
<Date>2015-09-22T19:03:06.9258653</Date> <Author>CONTOSO\dadmin</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author">
<RunLevel>LeastPrivilege</RunLevel> <UserId>CONTOSO\dadmin</UserId> <LogonType>InteractiveToken</LogonType> </Principal> </Principals> <Settings>
<MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
<AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings>
<StopOnIdleEnd>true</StopOnIdleEnd> <RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>true</Enabled>
<Hidden>false</Hidden> <RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings>
<Actions Context="Author"> <Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

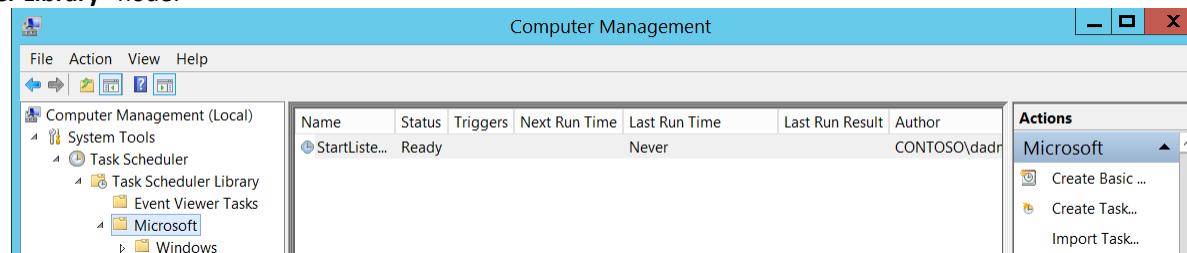
- **Security ID [Type = SID]:** SID of account that requested the “enable scheduled task” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enable scheduled task” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Task Information:

- **Task Name** [Type = UnicodeString]: enabled scheduled task name. The format of this value is “\task_path\task_name”, where **task_path** is a path in Microsoft **Task Scheduler** tree starting from “**Task Scheduler Library**” node:



- **Task Content** [Type = UnicodeString]: the [XML](#) of the enabled task. Here “[XML Task Definition Format](#)” you can read more about the XML format for scheduled tasks.

Security Monitoring Recommendations:

For 4700(S): A scheduled task was enabled.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If a highly critical scheduled task exists on some computers, and for some reason it should never be enabled, monitor for [4700](#) events with the corresponding **Task Name**.

4701(S): A scheduled task was disabled.

Event Properties - Event 4701, Microsoft Windows security auditing. X

Subject:	Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x364EB
Task Information:	Task Name: \Microsoft\StartListener Task Content: <?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo> <Date>2015-09-22T19:03:06.9258653</Date> <Author>CONTOSO\dadmin</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author"> <RunLevel>LeastPrivilege</RunLevel> <UserId>CONTOSO\dadmin</UserId> <LogonType>InteractiveToken</LogonType> </Principal> </Principals> <Settings> <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries> <AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings> <StopOnIdleEnd>true</StopOnIdleEnd> <RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>false</Enabled> <Hidden>false</Hidden> <RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings> <Actions Context="Author"> <Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task>

Event Description:

This event generates every time a scheduled task is disabled.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4701</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12804</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-23T02:32:45.844066600Z" />
  <EventRecordID>344860</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="4364" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x364eb</Data>
  <Data Name="TaskName">\Microsoft\StartListener</Data>
  <Data Name="TaskContent"><?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo> <Date>2015-09-22T19:03:06.9258653</Date> <Author>CONTOSO\dadmin</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author"> <RunLevel>LeastPrivilege</RunLevel> <UserId>CONTOSO\dadmin</UserId> <LogonType>InteractiveToken</LogonType> </Principal> </Principals> <Settings> <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries> <AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings> <StopOnIdleEnd>true</StopOnIdleEnd> </IdleSettings> </Settings> </Task>
```

<DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries> <AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings> <StopOnIdleEnd>true</StopOnIdleEnd>

```
<RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>false</Enabled> <Hidden>false</Hidden>
<RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings> <Actions Context="Author">
<Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

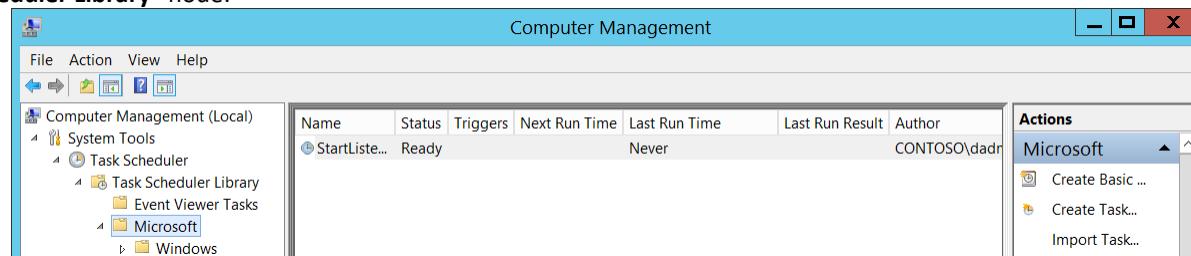
- **Security ID** [Type = SID]: SID of account that requested the “enable scheduled task” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enable scheduled task” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Task Information:

- **Task Name** [Type = UnicodeString]: disabled scheduled task name. The format of this value is “\task_path\task_name”, where **task_path** is a path in Microsoft **Task Scheduler** tree starting from “**Task Scheduler Library**” node:

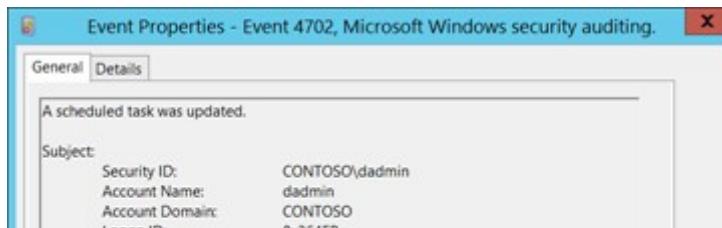


- **Task Content** [Type = UnicodeString]: the [XML](#) of the disabled task. Here “[XML Task Definition Format](#)” you can read more about the XML format for scheduled tasks.

Security Monitoring Recommendations:

For 4701(S): A scheduled task was disabled.

[Appendix A: Security monitoring recommendations for many audit events](#)



- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If a highly critical scheduled task exists on some computers, and it should never be disabled, monitor for [4701](#) events with the corresponding **Task Name**.

4702(S): A scheduled task was updated.

Event Description:

This event generates every time scheduled task was updated/changed.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4702</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12804</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T03:00:59.343820000Z" />
<EventRecordID>344863</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="596" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x364eb</Data>
<Data Name="TaskName">\Microsoft\StartListener</Data>

```

```
<Data Name="TaskContentNew"><?xml version="1.0" encoding="UTF-16"?> <Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"> <RegistrationInfo>
<Date>2015-09-22T19:03:06.9258653</Date> <Author>CONTOSO\admind</Author> </RegistrationInfo> <Triggers /> <Principals> <Principal id="Author">
<RunLevel>HighestAvailable</RunLevel> <UserId>CONTOSO\admind</UserId> <LogonType>InteractiveToken</LogonType> </Principal> </Principals> <Settings>
<MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy> <DisallowStartIfOnBatteries>true</DisallowStartIfOnBatteries> <StopIfGoingOnBatteries>true</StopIfGoingOnBatteries>
<AllowHardTerminate>true</AllowHardTerminate> <StartWhenAvailable>false</StartWhenAvailable> <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable> <IdleSettings>
<StopOnIdleEnd>true</StopOnIdleEnd> <RestartOnIdle>false</RestartOnIdle> </IdleSettings> <AllowStartOnDemand>true</AllowStartOnDemand> <Enabled>true</Enabled>
<Hidden>false</Hidden> <RunOnlyIfIdle>false</RunOnlyIfIdle> <WakeToRun>false</WakeToRun> <ExecutionTimeLimit>P3D</ExecutionTimeLimit> <Priority>7</Priority> </Settings>
<Actions Context="Author"> <Exec> <Command>C:\Documents\listener.exe</Command> </Exec> </Actions> </Task></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

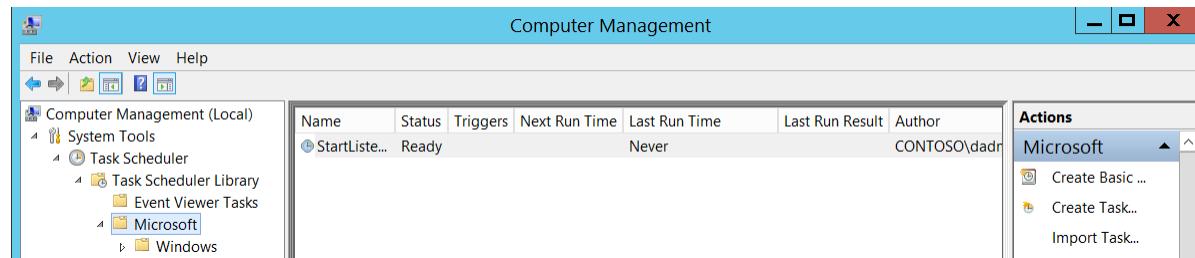
- **Security ID** [Type = SID]: SID of account that requested the “change/update scheduled task” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change/update scheduled task” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Task Information:

- **Task Name** [Type = UnicodeString]: updated/changed scheduled task name. The format of this value is “\task_path\task_name”, where **task_path** is a path in Microsoft **Task Scheduler** tree starting from “**Task Scheduler Library**” node:



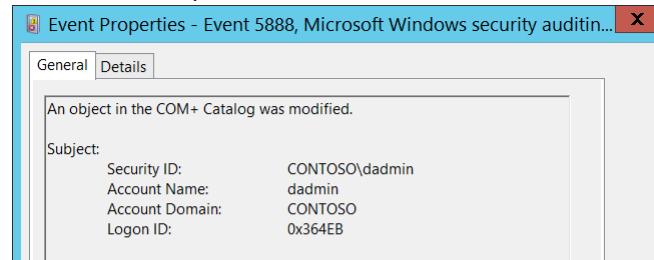
- **Task New Content** [Type = UnicodeString]: the new [XML](#) for the updated task. Here "[XML Task Definition Format](#)" you can read more about the XML format for scheduled tasks.

Security Monitoring Recommendations:

For 4702(S): A scheduled task was updated.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Monitor for updated scheduled tasks located in the **Task Scheduler Library** root node, that is, where **Task Name** looks like '\TASK_NAME'. Scheduled tasks that are created manually or by malware are often located in the **Task Scheduler Library** root node.
- In the updated scheduled task, if the **Task Content: XML** contains <LogonType>Password</LogonType> value, trigger an alert. In this case, the password for the account that will be used to run the scheduled task will be saved in Credential Manager in cleartext format, and can be extracted using Administrative privileges.

 Event Properties - Event 5888, Microsoft Windows security auditin... X

General	Details
An object in the COM+ Catalog was modified.	
Subject:	
Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x364EB

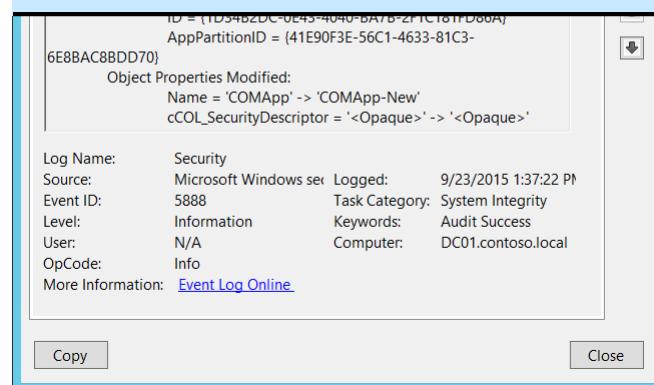
5888(S): An object in the COM+ Catalog was modified.

Event Description:

This event generates when the object in [COM+ Catalog](#) was modified.

For some reason this event belongs to [Audit System Integrity](#) subcategory, but generation of this event enables in this subcategory.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

 Event Properties - Event 5888, Microsoft Windows security auditin... X

ID = {FD34B2DC-0E43-4040-BA7B-2F1C181FD80A}	AppPartitionID = {41E90F3E-56C1-4633-81C3-6E8BAC8BDD70}
Object Properties Modified:	
Name = 'COMApp' -> 'COMApp-New'	
cCOL_SecurityDescriptor = '<Opaque>' -> '<Opaque>'	
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	5888
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>5888</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>12290</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
```

```
<TimeCreated SystemTime="2015-09-23T20:37:22.400120200Z" />
<EventRecordID>344994</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="1352" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectUserDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">222443</Data>
<Data Name="ObjectCollectionName">Applications</Data>
<Data Name="ObjectIdentifyingProperties">ID = {1D34B2DC-0E43-4040-BA7B-2F1C181FD86A} AppPartitionID = {41E90F3E-56C1-4633-81C3-6E8BAC8BDD70}</Data>
<Data Name="ModifiedObjectProperties">Name = 'COMApp' -> 'COMApp-New' cCOL_SecurityDescriptor = '<Opaque>' -> '<Opaque>'</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested the “modify/change object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]:** the name of the account that requested the “modify/change object” operation.
- **Account Domain [Type = UnicodeString]:** subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **COM+ Catalog Collection** [Type = UnicodeString]: the name of COM+ collection in which the object was modified. Here is the list of possible collection values with descriptions:

Collection	Description
ApplicationCluster	Contains a list of the servers in the application cluster.
ApplicationInstances	Contains an object for each instance of a running COM+ application.
Applications	Contains an object for each COM+ application installed on the local computer.
Components	Contains an object for each component in the application to which it is related.
ComputerList	Contains a list of the computers found in the Computers folder of the Component Services administration tool.
DCOMProtocols	Contains a list of the protocols to be used by DCOM. It contains an object for each protocol.
ErrorInfo	Retrieves extended error information regarding methods that deal with multiple objects.
EventClassesForIID	Retrieves information regarding event classes.
FilesForImport	Retrieves information from its MSI file about an application that can be imported.
InprocServers	Contains a list of the in-process servers registered with the system. It contains an object for each component.
InterfacesForComponent	Contains an object for each interface exposed by the component to which the collection is related.
LegacyComponents	Contains an object for each unconfigured component in the application to which it is related.
LegacyServers	Identical to the InprocServers collection except that this collection also includes local servers.
LocalComputer	Contains a single object that holds computer level settings information for the computer whose catalog you are accessing.
MethodsForInterface	Contains an object for each method on the interface to which the collection is related.
Partitions	Used to specify the applications contained in each partition.
PartitionUsers	Used to specify the users contained in each partition.
PropertyInfo	Retrieves information about the properties that a specified collection supports.
PublisherProperties	Contains an object for each publisher property for the parent SubscriptionsForComponent collection.
RelatedCollectionInfo	Retrieves information about other collections related to the collection from which it is called.
Roles	Contains an object for each role assigned to the application to which it is related.
RolesForComponent	Contains an object for each role assigned to the component to which the collection is related.
RolesForInterface	Contains an object for each role assigned to the interface to which the collection is related.
RolesForMethod	Contains an object for each role assigned to the method to which the collection is related.
RolesForPartition	Contains an object for each role assigned to the partition to which the collection is related.
Root	Contains the top-level collections on the catalog.
SubscriberProperties	Contains an object for each subscriber property for the parent SubscriptionsForComponent collection.
SubscriptionsForComponent	Contains an object for each subscription for the parent Components collection.
TransientPublisherProperties	Contains an object for each publisher property for the parent TransientSubscriptions collection.
TransientSubscriberProperties	Contains an object for each subscriber property for the parent TransientSubscriptions collection.
TransientSubscriptions	Contains an object for each transient subscription.
UsersInPartitionRole	Contains an object for each user in the partition role to which the collection is related.

UsersInRole	Contains an object for each user in the role to which the collection is related.
WOWInprocServers	Contains a list of the in-process servers registered with the system for 32-bit components on 64-bit computers.
WOWLegacyServers	Identical to the LegacyServers collection except that this collection is drawn from the 32-bit registry on 64-bit computers.

- **Object Name** [Type = UnicodeString]: object-specific fields with the names and identifiers for the modified object. It depends on **COM+ Catalog Collection** value, for example, if **COM+ Catalog Collection** = [Applications](#), then you can find that:
 - **ID** - A GUID representing the application. This property is returned when the [Key](#) property method is called on an object of this collection.
 - **AppPartitionID** - A GUID representing the application partition ID.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Event Properties - Event 5889, Microsoft Windows security auditing. X

General	Details
An object was deleted from the COM+ Catalog.	
Subject:	Security ID: CONTOSO\dadmin
Object:	COM+ Catalog Collection: Applications Object Name: ID = {1D34B2DC-0E43-4040-BA7B-2F1C181FD86A} AppPartitionID = {41E90F3E-56C1-4633-81C3-6E8BAC8BDD70} Object Details: Name = COMApp-New ApplicationProxyServerName = ProcessType = 2 CommandLine = ServiceName = <null> RunAsUserType = 1 Identity = Interactive User Description =

- **Object Properties Modified** [Type = UnicodeString]: the list of object's (**Object Name**) properties which were modified.

The items have the following format: Property_Name = 'OLD_VALUE' -> 'NEW_VALUE'

Check description for specific **COM+ Catalog Collection** to see the list of object's properties and descriptions.

Security Monitoring Recommendations:

For 5888(S): An object in the COM+ Catalog was modified.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you have a specific COM+ object for which you need to monitor all modifications, monitor all [5888](#) events with the corresponding **Object Name**.

5889(S): An object was deleted from the COM+ Catalog.

Event Description:

This event generates when the object in the [COM+ Catalog](#) was deleted.

For some reason this event belongs to [Audit System Integrity](#) subcategory, but generation of this event enables in this subcategory.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

ShutdownAfter = 3 RunForever = N Password = ***** Activation = Local	
Log Name: Security Source: Microsoft Windows sec Event ID: 5889 Level: Information User: N/A OpCode: Info More Information: Event Log Online	Logged: 9/23/2015 1:44:42 PM Task Category: System Integrity Keywords: Audit Success Computer: DC01.contoso.local
Copy Close	

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5889</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12290</Task>
<Opcode>0</Opcode>
```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T20:44:42.948569400Z" />
<EventRecordID>344998</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="4756" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectUserDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">222443</Data>
  <Data Name="ObjectCollectionName">Applications</Data>
  <Data Name="ObjectIdentifyingProperties">ID = {1D34B2DC-0E43-4040-BA7B-2F1C181FD86A} AppPartitionID = {41E90F3E-56C1-4633-81C3-6E8BAC8BDD70}</Data>
  <Data Name="ObjectProperties">Name = COMApp-New ApplicationProxyServerName = ProcessType = 2 CommandLine = ServiceName = <null> RunAsUserType = 1 Identity = Interactive User Description = IsSystem = N Authentication = 4 ShutdownAfter = 3 RunForever = N Password = ***** Activation = Local Changeable = Y Deleteable = Y CreatedBy = AccessChecksLevel = 1 ApplicationAccessChecksEnabled = 1 cCOL_SecurityDescriptor = <Opaque> ImpersonationLevel = 3 AuthenticationCapability = 64 CRMEnabled = 0 3GigSupportEnabled = 0 QueuingEnabled = 0 QueueListenerEnabled = N EventsEnabled = 1 ProcessFlags = 0 ThreadMax = 0 ApplicationProxy = 0 CRMLLogFile = DumpEnabled = 0 DumpOnException = 0 DumpOnFailfast = 0 MaxDumpCount = 5 DumpPath = %systemroot%\system32\com\dmp IsEnabled = 1 AppPartitionID = {41E90F3E-56C1-4633-81C3-6E8BAC8BDD70} ConcurrentApps = 1 RecycleLifetimeLimit = 0 RecycleCallLimit = 0 RecycleActivationLimit = 0 RecycleMemoryLimit = 0 RecycleExpirationTimeout = 15 QCListenerMaxThreads = 0 QCAuthenticateMsgs = 0 ApplicationDirectory = SRPTrustLevel = 262144 SRPEnabled = 0 SoapActivated = 0 SoapVRoot = SoapMailTo = SoapBaseUrl = Replicable = 1</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “delete object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “delete object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **COM+ Catalog Collection** [Type = UnicodeString]: the name of COM+ collection in which COM+ object was deleted. Here is the list of possible collection values with descriptions:

Collection	Description
ApplicationCluster	Contains a list of the servers in the application cluster.
ApplicationInstances	Contains an object for each instance of a running COM+ application.
Applications	Contains an object for each COM+ application installed on the local computer.
Components	Contains an object for each component in the application to which it is related.
ComputerList	Contains a list of the computers found in the Computers folder of the Component Services administration tool.
DCOMProtocols	Contains a list of the protocols to be used by DCOM. It contains an object for each protocol.
ErrorInfo	Retrieves extended error information regarding methods that deal with multiple objects.
EventClassesForIID	Retrieves information regarding event classes.
FilesForImport	Retrieves information from its MSI file about an application that can be imported.
InprocServers	Contains a list of the in-process servers registered with the system. It contains an object for each component.
InterfacesForComponent	Contains an object for each interface exposed by the component to which the collection is related.
LegacyComponents	Contains an object for each unconfigured component in the application to which it is related.
LegacyServers	Identical to the InprocServers collection except that this collection also includes local servers.
LocalComputer	Contains a single object that holds computer level settings information for the computer whose catalog you are accessing.
MethodsForInterface	Contains an object for each method on the interface to which the collection is related.
Partitions	Used to specify the applications contained in each partition.
PartitionUsers	Used to specify the users contained in each partition.
PropertyInfo	Retrieves information about the properties that a specified collection supports.
PublisherProperties	Contains an object for each publisher property for the parent SubscriptionsForComponent collection.
RelatedCollectionInfo	Retrieves information about other collections related to the collection from which it is called.
Roles	Contains an object for each role assigned to the application to which it is related.
RolesForComponent	Contains an object for each role assigned to the component to which the collection is related.
RolesForInterface	Contains an object for each role assigned to the interface to which the collection is related.
RolesForMethod	Contains an object for each role assigned to the method to which the collection is related.
RolesForPartition	Contains an object for each role assigned to the partition to which the collection is related.

Root	Contains the top-level collections on the catalog.
SubscriberProperties	Contains an object for each subscriber property for the parent SubscriptionsForComponent collection.
SubscriptionsForComponent	Contains an object for each subscription for the parent Components collection.
TransientPublisherProperties	Contains an object for each publisher property for the parent TransientSubscriptions collection.
TransientSubscriberProperties	Contains an object for each subscriber property for the parent TransientSubscriptions collection.
TransientSubscriptions	Contains an object for each transient subscription.
UsersInPartitionRole	Contains an object for each user in the partition role to which the collection is related.
UsersInRole	Contains an object for each user in the role to which the collection is related.
WOWInprocServers	Contains a list of the in-process servers registered with the system for 32-bit components on 64-bit computers.
WOWLegacyServers	Identical to the LegacyServers collection except that this collection is drawn from the 32-bit registry on 64-bit computers.

- **Object Name** [Type = UnicodeString]: object-specific fields with the names and identifiers for the deleted object. It depends on **COM+ Catalog Collection** value, for example, if **COM+ Catalog Collection** = [Applications](#), then you can find that:
 - **ID** - A GUID representing the application. This property is returned when the [Key](#) property method is called on an object of this collection.
 - **AppPartitionID** - A GUID representing the application partition ID.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Object Details** [Type = UnicodeString]: the list of deleted object's (**Object Name**) properties.

The items have the following format: Property_Name = VALUE

Check description for specific **COM+ Catalog Collection** to see the list of object's properties and descriptions.

Security Monitoring Recommendations:

For 5889(S): An object was deleted from the COM+ Catalog.

[Appendix A: Security monitoring recommendations for many audit events](#)

Event Properties - Event 5890, Microsoft Windows security audit... X

General	Details
An object was added to the COM+ Catalog.	
Subject:	
Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x364EB	
Object:	
COM+ Catalog Collection: Roles Object Name: ApplId = {1D34B2DC-0E43-4040-BA7B-2F1C181FD86A} Name = CreatorOwner	
Object Details:	
Description =	
Log Name:	Security
Source:	Microsoft Windows se
Event ID:	5890
Logged:	9/23/2015 12:45:04 I
Task Category:	System Integrity

Specified time range: More Information: [Event Log Online](#)

Copy Close

5890(S): An object was added to the COM+ Catalog.

Event Description:

This event generates when new object was added to the [COM+ Catalog](#).

For some reason this event belongs to [Audit System Integrity](#) subcategory, but generation of this event enables in this subcategory.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
```

```
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5890</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12290</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-23T19:45:04.239886800Z" />
<EventRecordID>344980</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="2856" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectUserDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">222443</Data>
<Data Name="ObjectCollectionName">Roles</Data>
<Data Name="ObjectIdentifyingProperties">ApplId = {1D34B2DC-0E43-4040-BA7B-2F1C181FD86A} Name = CreatorOwner</Data>
<Data Name="ObjectProperties">Description =</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “add object” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “add object” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **COM+ Catalog Collection** [Type = UnicodeString]: the name of COM+ collection to which the new object was added. Here is the list of possible collection values with descriptions:

Collection	Description
ApplicationCluster	Contains a list of the servers in the application cluster.
ApplicationInstances	Contains an object for each instance of a running COM+ application.
Applications	Contains an object for each COM+ application installed on the local computer.
Components	Contains an object for each component in the application to which it is related.
ComputerList	Contains a list of the computers found in the Computers folder of the Component Services administration tool.
DCOMProtocols	Contains a list of the protocols to be used by DCOM. It contains an object for each protocol.
ErrorInfo	Retrieves extended error information regarding methods that deal with multiple objects.
EventClassesForIID	Retrieves information regarding event classes.
FilesForImport	Retrieves information from its MSI file about an application that can be imported.
InprocServers	Contains a list of the in-process servers registered with the system. It contains an object for each component.
InterfacesForComponent	Contains an object for each interface exposed by the component to which the collection is related.
LegacyComponents	Contains an object for each unconfigured component in the application to which it is related.
LegacyServers	Identical to the InprocServers collection except that this collection also includes local servers.
LocalComputer	Contains a single object that holds computer level settings information for the computer whose catalog you are accessing.
MethodsForInterface	Contains an object for each method on the interface to which the collection is related.
Partitions	Used to specify the applications contained in each partition.
PartitionUsers	Used to specify the users contained in each partition.
PropertyInfo	Retrieves information about the properties that a specified collection supports.
PublisherProperties	Contains an object for each publisher property for the parent SubscriptionsForComponent collection.
RelatedCollectionInfo	Retrieves information about other collections related to the collection from which it is called.
Roles	Contains an object for each role assigned to the application to which it is related.
RolesForComponent	Contains an object for each role assigned to the component to which the collection is related.
RolesForInterface	Contains an object for each role assigned to the interface to which the collection is related.
RolesForMethod	Contains an object for each role assigned to the method to which the collection is related.
RolesForPartition	Contains an object for each role assigned to the partition to which the collection is related.

Root	Contains the top-level collections on the catalog.
SubscriberProperties	Contains an object for each subscriber property for the parent SubscriptionsForComponent collection.
SubscriptionsForComponent	Contains an object for each subscription for the parent Components collection.
TransientPublisherProperties	Contains an object for each publisher property for the parent TransientSubscriptions collection.
TransientSubscriberProperties	Contains an object for each subscriber property for the parent TransientSubscriptions collection.
TransientSubscriptions	Contains an object for each transient subscription.
UsersInPartitionRole	Contains an object for each user in the partition role to which the collection is related.
UsersInRole	Contains an object for each user in the role to which the collection is related.
WOWInprocServers	Contains a list of the in-process servers registered with the system for 32-bit components on 64-bit computers.
WOWLegacyServers	Identical to the LegacyServers collection except that this collection is drawn from the 32-bit registry on 64-bit computers.

- **Object Name** [Type = UnicodeString]: object-specific fields with the names and identifiers for the new object. It depends on **COM+ Catalog Collection** value, for example, if **COM+ Catalog Collection** = [Applications](#), then you can find that:
 - **ID** - A GUID representing the application. This property is returned when the [Key](#) property method is called on an object of this collection.
 - **AppPartitionID** - A GUID representing the application partition ID.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Object Details** [Type = UnicodeString]: the list of new object's (**Object Name**) properties.

The items have the following format: Property_Name = VALUE

Check description for specific **COM+ Catalog Collection** to see the list of object's properties and descriptions.

Security Monitoring Recommendations:

For 5890(S): An object was added to the COM+ Catalog.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. If you need to monitor for creation of new COM+ objects within specific COM+ collection, monitor all [5890](#) events with the corresponding **COM+ Catalog Collection** field value.

Audit Registry

Audit Registry allows you to audit attempts to access registry objects. A security audit event is generated only for objects that have system access control lists ([SACLs](#)) specified, and only if the type of access requested, such as Read, Write, or Modify, and the account making the request match the settings in the SACL.

If success auditing is enabled, an audit entry is generated each time any account successfully accesses a registry object that has a matching SACL. If failure auditing is enabled, an audit entry is generated each time any user unsuccessfully attempts to access a registry object that has a matching SACL.

Event volume: Low to Medium, depending on how registry SACLs are configured.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	IF	IF	IF	We strongly recommend that you develop a Registry Objects Security Monitoring policy and define appropriate SACLs for registry objects for different operating system templates and roles. Do not enable this subcategory if you have not planned how to use and analyze the collected information. It is also important to delete non-effective, excess SACLs . Otherwise the auditing log will be overloaded with useless information.
Member Server	IF	IF	IF	IF	Failure events can show you unsuccessful attempts to access specific registry objects.
Workstation	IF	IF	IF	IF	Consider enabling this subcategory for critical computers first, after you develop a Registry Objects Security Monitoring policy for them.

Events List:

- [4663\(S\)](#): An attempt was made to access an object.
- [4656\(S, F\)](#): A handle to an object was requested.
- [4658\(S\)](#): The handle to an object was closed.
- [4660\(S\)](#): An object was deleted.
- [4657\(S\)](#): A registry value was modified.
- [5039\(-\)](#): A registry key was virtualized.
- [4670\(S\)](#): Permissions on an object were changed.

[4663\(S\)](#): An attempt was made to access an object.

This event also belongs in the Audit File System subcategory, and is described there. See "[4663\(S\)](#): An attempt was made to access an object."

[4656\(S, F\)](#): A handle to an object was requested.

This event also belongs in the Audit File System subcategory, and is described there. See "[4656\(S, F\)](#): A handle to an object was requested."

[4658\(S\)](#): The handle to an object was closed.

This event also belongs in the Audit File System subcategory, and is described there. See "[4658\(S\)](#): The handle to an object was closed."

[4660\(S\)](#): An object was deleted.

This event also belongs in the Audit File System subcategory, and is described there. See "[4660\(S\)](#): An object was deleted."

4657(S): A registry value was modified.

Event Properties - Event 4657, Microsoft Windows security auditi... X

General	Details
Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x364EB	
Object: Object Name: \REGISTRY\MACHINE Object Value Name: Name_New Handle ID: 0x54 Operation Type: Existing registry value modified	Up Down
Process Information: Process ID: 0xce4 Process Name: C:\Windows\regedit.exe	
Change Information: Old Value Type: REG_SZ Old Value: New Value Type: REG_SZ New Value: Andrei	
Log Name: Security Source: Microsoft Windows se... Logged: 9/23/2015 6:28:43 PM Event ID: 4657 Task Category: Registry Level: Information Keywords: Audit Success User: N/A Computer: DC01.contoso.local OpCode: Info More Information: Event Log Online	
<input type="button" value="Copy"/> <input type="button" value="Close"/>	

Event Description:

This event generates when a registry key **value** was modified. It doesn't generate when a registry key was modified. This event generates only if "Set Value" auditing is set in registry key's **SACL**.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4657</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12801</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-24T01:28:43.639634100Z" />
<EventRecordID>744725</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="4824" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x364eb</Data>
<Data Name="ObjectName">\REGISTRY\MACHINE</Data>
<Data Name="ObjectValueName">Name_New</Data>
<Data Name="HandleId">0x54</Data>
<Data Name="OperationType">%1905</Data>
<Data Name="OldValueType">%1873</Data>
<Data Name="OldValue" />
<Data Name="newValueType">%1873</Data>
<Data Name="newValue">Andrei</Data>
```

```
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x364eb</Data>
<Data Name="ObjectName">\REGISTRY\MACHINE</Data>
<Data Name="ObjectValueName">Name_New</Data>
<Data Name="HandleId">0x54</Data>
<Data Name="OperationType">%1905</Data>
<Data Name="OldValueType">%1873</Data>
<Data Name="OldValue" />
<Data Name="newValueType">%1873</Data>
<Data Name="newValue">Andrei</Data>
```

```
<Data Name="ProcessId">0xce4</Data>
<Data Name="ProcessName">C:\Windows\regedit.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “modify registry value” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “modify registry value” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

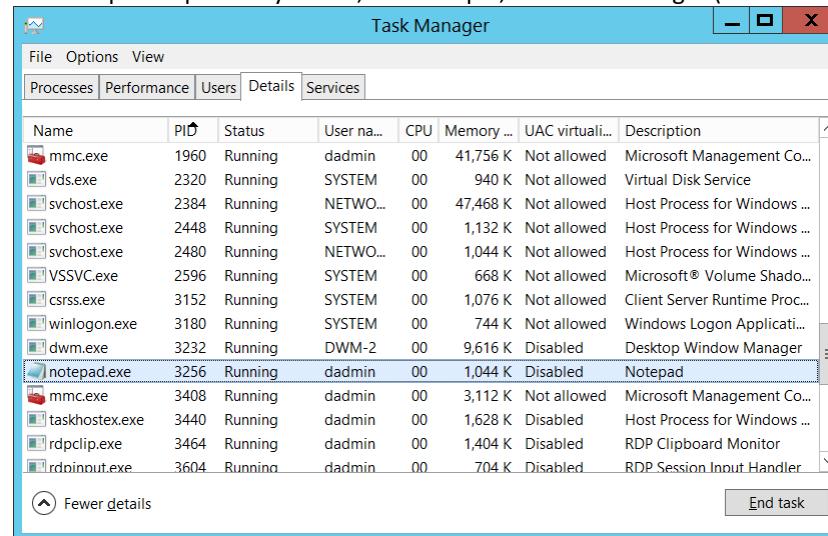
Object:

- **Object Name** [Type = UnicodeString]: full path and name of the registry key which value was modified. The format is: \REGISTRY\HIVE\PATH where:
 - HIVE:
 - HKEY_LOCAL_MACHINE = \REGISTRY\MACHINE
 - HKEY_CURRENT_USER = \REGISTRY\USER\[USER_SID], where [USER_SID] is the SID of current user.
 - HKEY_CLASSES_ROOT = \REGISTRY\MACHINE\SOFTWARE\Classes
 - HKEY_USERS = \REGISTRY\USER
 - HKEY_CURRENT_CONFIG = \REGISTRY\MACHINE\SYSTEM\ControlSet001\Hardware Profiles\Current
 - PATH – path to the registry key.
- **Object Value Name** [Type = UnicodeString]: the name of modified registry key value.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4656](#): A handle to an object was requested.” This parameter might not be captured in the event, and in that case appears as “0x0”.
- **Operation Type** [Type = UnicodeString]: the type of performed operation with registry key value. Most common operations are:

- New registry value created
- Registry value deleted
- Existing registry value modified

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the registry key value was modified. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Change Information:

- **Old Value Type** [Type = UnicodeString]: old type of changed registry key value. Registry key value types:

Value Type	Description
REG_SZ	String
REG_BINARY	Binary
REG_DWORD	DWORD (32-bit) Value
REG_QWORD	QWORD (64-bit) Value
REG_MULTI_SZ	Multi-String Value
REG_EXPAND_SZ	Expandable String Value

- **Old Value** [Type = UnicodeString]: old value for changed registry key value.
- **New Value Type** [Type = UnicodeString]: new type of changed registry key value. See table above for possible values.
- **New Value** [Type = UnicodeString]: new value for changed registry key value.

Security Monitoring Recommendations:

For 4657(S): A registry value was modified.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If **Object Name** is a sensitive or critical registry key for which you need to monitor any modification of its values, monitor all [4657](#) events.
- If **Object Name** has specific values (**Object Value Name**) and you need to monitor modifications of these values, monitor for all [4657](#) events.

5039(-): A registry key was virtualized.

This event should be generated when registry key was virtualized using [LUAFV](#).

This event occurs very rarely during standard LUAFV registry key virtualization.

There is no example of this event in this document.

Event Schema:

A registry key was virtualized.

Subject:

Security ID:%1%
Account Name:%2
Account Domain:%3
Logon ID:%4

Object:

Key Name:%5
Virtual Key Name:%6

Process Information:

Process ID:%7
Process Name%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4670(S): Permissions on an object were changed.

This event also belongs in the Audit File System subcategory, and is described there. See "[4670\(S\): Permissions on an object were changed.](#)"

Audit Removable Storage

Audit Removable Storage allows you to audit user attempts to access file system objects on a removable storage device. A security audit event is generated for all objects and all types of access requested, with no dependency on object's [SACL](#).

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	This subcategory will help identify when and which files or folders were accessed or modified on removable devices.
Member Server	Yes	Yes	Yes	Yes	It is often useful to track actions with removable storage devices and the files or folders on them, because malicious software very often uses removable devices as a method to get into the system. At the same time, you will be able to track which files were written or executed from a removable storage device.
Workstation	Yes	Yes	Yes	Yes	You can track, for example, actions with files or folders on USB flash drives or sticks that were inserted into domain controllers or high value servers, which is typically not allowed. We recommend Failure auditing to track failed access attempts.

Events List:

- [4656](#)(S, F): A handle to an object was requested.
- [4658](#)(S): The handle to an object was closed.
- [4663](#)(S): An attempt was made to access an object.

[4656](#)(S, F): A handle to an object was requested.

This event also belongs in the Audit File System subcategory, and is described there. See "[4656](#)(S, F): A handle to an object was requested."

[4658](#)(S): The handle to an object was closed.

This event also belongs in the Audit File System subcategory, and is described there. See "[4658](#)(S): The handle to an object was closed."

[4663](#)(S): An attempt was made to access an object.

This event also belongs in the Audit File System subcategory, and is described there. See "[4663](#)(S): An attempt was made to access an object."

Audit SAM

Audit SAM, which enables you to audit events that are generated by attempts to access Security Account Manager ([SAM](#)) objects.

The Security Account Manager (SAM) is a database that is present on computers running Windows operating systems that stores user accounts and security descriptors for users on the local computer.

- SAM objects include the following:
 - SAM_ALIAS: A local group
 - SAM_GROUP: A group that is not a local group
 - SAM_USER: A user account
 - SAM_DOMAIN: A domain
 - SAM_SERVER: A computer account

If you configure this policy setting, an audit event is generated when a SAM object is accessed. Success audits record successful attempts, and failure audits record unsuccessful attempts. Only a [SACL](#) for SAM_SERVER can be modified.

Changes to user and group objects are tracked by the Account Management audit category. However, user accounts with enough privileges could potentially alter the files in which the account and password information is stored in the system, bypassing any Account Management events.

Event volume: High on domain controllers.

For information about reducing the number of events generated in this subcategory, see [KB841001](#).

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at Security Account Manager level.
Member Server	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at Security Account Manager level.
Workstation	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at Security Account Manager level.

Events List:

- [4661](#)(S, F): A handle to an object was requested.

4661(S, F): A handle to an object was requested.

This event also belongs in the Audit Directory Service Access subcategory, and is described there. See "[4661](#)(S, F): A handle to an object was requested."

Audit Central Policy Staging

Audit Central Policy Staging allows you to audit access requests where a permission granted or denied by a proposed policy differs from the current central access policy on an object. If you configure this policy setting, an audit event is generated each time a user accesses an object and the permission granted by the current central access policy on the object differs from that granted by the proposed policy. The resulting audit event is generated as follows:

- Success audits, when configured, record access attempts when the current central access policy grants access, but the proposed policy denies access.
- Failure audits, when configured, record access attempts when:
 - The current central access policy does not grant access, but the proposed policy grants access.
 - A principal requests the maximum access rights they are allowed and the access rights granted by the current central access policy are different than the access rights granted by the proposed policy.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	No	IF	No	<p>IF - Enable this subcategory if you need to test or troubleshoot Dynamic Access Control Proposed Central Access Policies.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	IF	No	IF	No	<p>IF - Enable this subcategory if you need to test or troubleshoot Dynamic Access Control Proposed Central Access Policies.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	IF	No	IF	No	<p>IF - Enable this subcategory if you need to test or troubleshoot Dynamic Access Control Proposed Central Access Policies.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4818\(S\)](#): Proposed Central Access Policy does not grant the same access permissions as the current Central Access Policy.

4818(S): Proposed Central Access Policy does not grant the same access permissions as the current Central Access Policy.

Event Properties - Event 4818, Microsoft Windows security auditing.

General Details

Account Domain: CONTOSO
Logon ID: 0x1E5F21

Object:

- Object Server: Security
- Object Type: File
- Object Name: C:\Finance Documents\desktop.ini
- Handle ID: 0xc64

Process Information:

- Process ID: 0x4
- Process Name:

Current Central Access Policy results:

Access Reasons:	READ_CONTROL: Granted by D:(A;ID;0x1200a9;;BU)
	SYNCHRONIZE: Granted by D:(A;ID;0x1200a9;;BU)
	ReadData (or ListDirectory): Granted by D:(A;ID;0x1200a9;;BU)
	ReadEA: Granted by D:(A;ID;0x1200a9;;BU)
	ReadAttributes: Granted by D:(A;ID;0x1200a9;;BU)

Proposed Central Access Policy results that differ from the current Central Access Policy results:

Access Reasons:	READ_CONTROL: NOT Granted by Central Access Rule Finance Documents Rule
	SYNCHRONIZE: NOT Granted by Central Access Rule Finance Documents Rule
	ReadData (or ListDirectory): NOT Granted by Central Access Rule Finance Documents Rule
	ReadEA: NOT Granted by Central Access Rule Finance Documents Rule
	ReadAttributes: NOT Granted by Central Access Rule Finance Documents Rule

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4818
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates when Dynamic Access Control Proposed [Central Access Policy](#) is enabled and access was not granted by Proposed Central Access Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4818</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12813</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-30T16:37:29.473472100Z" />
<EventRecordID>1049324</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="SubjectUserName">Auditor</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x1e5f21</Data>
```

```
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Finance Documents\desktop.ini</Data>
<Data Name="HandleId">0xc64</Data>
<Data Name="ProcessId">0x4</Data>
```

```

<Data Name="ProcessName" />
<Data Name="AccessReason">%&1538: %&1801 D:(A;ID;0x1200a9;;BU) %&1541: %&1801 D:(A;ID;0x1200a9;;BU) %&4416: %&1801 D:(A;ID;0x1200a9;;BU) %&4419: %&1801 D:(A;ID;0x1200a9;;BU) %&4423: %&1801 D:(A;ID;0x1200a9;;BU)</Data>
<Data Name="StagingReason">%&1538: %&1814Finance Documents Rule %&1541: %&1814Finance Documents Rule %&4416: %&1814Finance Documents Rule %&4419: %&1814Finance Documents Rule %&4423: %&1814Finance Documents Rule</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made an access request. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an access request.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString]: has "**Security**" value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation. Always "**File**" for this event.

The following table contains the list of the most common **Object Types**:

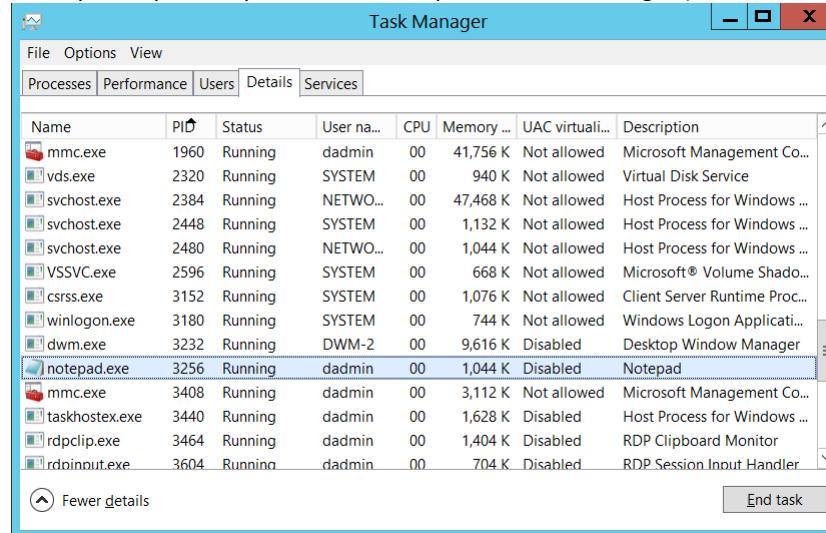
Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter

Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: full path and name of the file or folder for which access was requested.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the access was requested. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Current Central Access Policy results:

- **Access Reasons** [Type = UnicodeString]: the list of access check results for Current Access Policy. The format of the result is:

REQUESTED_ACCESS: RESULT ACE_WICH_PROVDED_OR_DENIED_ACCESS.

- REQUESTED_ACCESS – the name of requested access. See the possible REQUESTED_ACCESS values in the table below:

Access	Hexadecimal Value	Description
ReadData (or ListDirectory)	0x1	ReadData - For a file object, the right to read the corresponding file data. For a directory object, the right to read the corresponding directory data. ListDirectory - For a directory, the right to list the contents of the directory.
WriteData (or AddFile)	0x2	WriteData - For a file object, the right to write data to the file. For a directory object, the right to create a file in the directory (FILE_ADD_FILE).

		AddFile - For a directory, the right to create a file in the directory.
AppendData (or AddSubdirectory or CreatePipeInstance)	0x4	AppendData - For a file object, the right to append data to the file. (For local files, write operations will not overwrite existing data if this flag is specified without FILE_WRITE_DATA .) For a directory object, the right to create a subdirectory (FILE_ADD_SUBDIRECTORY). AddSubdirectory - For a directory, the right to create a subdirectory. CreatePipeInstance - For a named pipe, the right to create a pipe.
ReadEA	0x8	The right to read extended file attributes.
WriteEA	0x10	The right to write extended file attributes.
Execute/Traverse	0x20	Execute - For a native code file, the right to execute the file. This access right given to scripts may cause the script to be executable, depending on the script interpreter. Traverse - For a directory, the right to traverse the directory. By default, users are assigned the BYPASS_TRAVERSE_CHECKING privilege, which ignores the FILE_TRAVERSE access right. See the remarks in File Security and Access Rights for more information.
DeleteChild	0x40	For a directory, the right to delete a directory and all the files it contains, including read-only files.
ReadAttributes	0x80	The right to read file attributes.
WriteAttributes	0x100	The right to write file attributes.
DELETE	0x10000	The right to delete the object.
READ_CONTROL	0x20000	The right to read the information in the object's security descriptor, not including the information in the system access control list (SACL).
WRITE_DAC	0x40000	The right to modify the discretionary access control list (DACL) in the object's security descriptor.
WRITE_OWNER	0x80000	The right to change the owner in the object's security descriptor
SYNCHRONIZE	0x100000	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.
ACCESS_SYS_SEC	0x1000000	The ACCESS_SYS_SEC access right controls the ability to get or set the SACL in an object's security descriptor.

- RESULT:
 - Granted by
 - Denied by
 - Granted by ACE on parent folder
 - Not granted due to missing – after this sentence you will typically see missing user rights, for example SeSecurityPrivilege.
 - Unknown or unchecked
- ACE WHICH PROVEDDED OR DENIED ACCESS:
 - Ownership – if access was granted because of ownership of an object.
 - User Right name, for example SeSecurityPrivilege.
 - The [Security Descriptor Definition Language](#) (SDDL) value for the Access Control Entry (ACE) that granted or denied access.

Proposed Central Access Policy results that differ from the current Central Access Policy results:

- **Access Reasons** [Type = UnicodeString]: the list of access check results for Proposed Central Access Policy. Here you will see only denied requests. The format of the result is:
REQUESTED_ACCESS: NOT Granted by RULE_NAME Rule.

- REQUESTED_ACCESS – the name of requested access. See the possible REQUESTED_ACCESS values in the table below:

Access	Hexadecimal Value	Description
ReadData (or ListDirectory)	0x1	ReadData - For a file object, the right to read the corresponding file data. For a directory object, the right to read the corresponding directory data. ListDirectory - For a directory, the right to list the contents of the directory.
WriteData (or AddFile)	0x2	WriteData - For a file object, the right to write data to the file. For a directory object, the right to create a file in the directory (FILE_ADD_FILE). AddFile - For a directory, the right to create a file in the directory.
AppendData (or AddSubdirectory or CreatePipeInstance)	0x4	AppendData - For a file object, the right to append data to the file. (For local files, write operations will not overwrite existing data if this flag is specified without FILE_WRITE_DATA .) For a directory object, the right to create a subdirectory (FILE_ADD_SUBDIRECTORY). AddSubdirectory - For a directory, the right to create a subdirectory. CreatePipeInstance - For a named pipe, the right to create a pipe.
ReadEA	0x8	The right to read extended file attributes.
WriteEA	0x10	The right to write extended file attributes.
Execute/Traverse	0x20	Execute - For a native code file, the right to execute the file. This access right given to scripts may cause the script to be executable, depending on the script interpreter. Traverse - For a directory, the right to traverse the directory. By default, users are assigned the BYPASS_TRAVERSE_CHECKING privilege, which ignores the FILE_TRAVERSE access right. See the remarks in File Security and Access Rights for more information.
DeleteChild	0x40	For a directory, the right to delete a directory and all the files it contains, including read-only files.
ReadAttributes	0x80	The right to read file attributes.
WriteAttributes	0x100	The right to write file attributes.
DELETE	0x10000	The right to delete the object.
READ_CONTROL	0x20000	The right to read the information in the object's security descriptor, not including the information in the system access control list (SACL).
WRITE_DAC	0x40000	The right to modify the discretionary access control list (DACL) in the object's security descriptor.
WRITE_OWNER	0x80000	The right to change the owner in the object's security descriptor
SYNCHRONIZE	0x100000	The right to use the object for synchronization. This enables a thread to wait until the object is in the signaled state. Some object types do not support this access right.
ACCESS_SYS_SEC	0x1000000	The ACCESS_SYS_SEC access right controls the ability to get or set the SACL in an object's security descriptor.

- RULE_NAME: the name of Central Access Rule which denied the access.

Security Monitoring Recommendations:

For 4818(S): Proposed Central Access Policy does not grant the same access permissions as the current Central Access Policy.

- This event typically used for troubleshooting and testing of Proposed Central Access Policies for Dynamic Access Control.

Policy Change

Audit Policy Change

Audit Policy Change determines whether the operating system generates audit events when changes are made to audit policy.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	Almost all events in this subcategory have security relevance and should be monitored. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	Yes	No	Yes	No	Almost all events in this subcategory have security relevance and should be monitored. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	Yes	No	Yes	No	Almost all events in this subcategory have security relevance and should be monitored. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Changes to audit policy that are audited include:

- Changing permissions and audit settings on the audit policy object (by using "auditpol /set /sd" command).
- Changing the system audit policy.
- Registering and unregistering security event sources.
- Changing per-user audit settings.
- Changing the value of CrashOnAuditFail.
- Changing audit settings on an object (for example, modifying the system access control list ([SACL](#)) for a file or registry key).

[SACL](#) change auditing is performed when a SACL for an object has changed and the Policy Change category is configured. Discretionary access control list (DACL) and owner change auditing are performed when Object Access auditing is configured and the object's SACL is set for auditing of the DACL or owner change.

- Changing anything in the Special Groups list.

The following events will be enabled with Success auditing in this subcategory:

- 4902(S): The Per-user audit policy table was created.
- 4907(S): Auditing settings on object were changed.
- 4904(S): An attempt was made to register a security event source.
- 4905(S): An attempt was made to unregister a security event source.

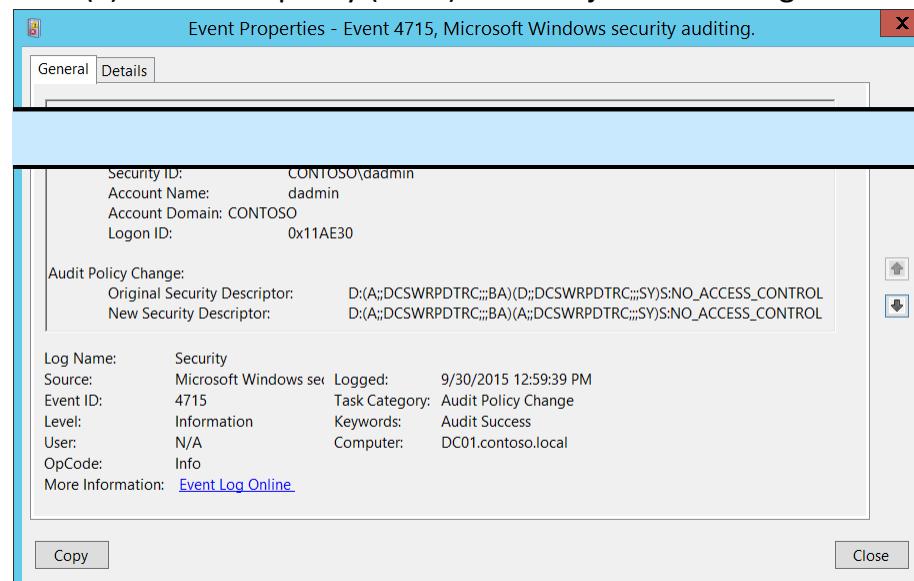
All other events in this subcategory will be logged regardless of the "Audit Policy Change" setting.

Events List:

- [4715](#)(S): The audit policy (SACL) on an object was changed.
- [4719](#)(S): System audit policy was changed.

- [4817\(S\)](#): Auditing settings on object were changed.
- [4902\(S\)](#): The Per-user audit policy table was created.
- [4906\(S\)](#): The CrashOnAuditFail value has changed.
- [4907\(S\)](#): Auditing settings on object were changed.
- [4908\(S\)](#): Special Groups Logon table modified.
- [4912\(S\)](#): Per User Audit Policy was changed.
- [4904\(S\)](#): An attempt was made to register a security event source.
- [4905\(S\)](#): An attempt was made to unregister a security event source.

4715([S](#)): The audit policy (SACL) on an object was changed.



Event Description:

This event generates every time local audit policy security descriptor changes.

This event is always logged regardless of the "Audit Policy Change" sub-category setting.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4715</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-30T19:59:39.964601800Z" />
```

```
<EventRecordID>1049425</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="4668" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
```

```
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x11ae30</Data>
<Data Name="OldSd">D:(A;;DCSWRPDTRC;;;BA)(D;;DCSWRPDTRC;;;SY)S:NO_ACCESS_CONTROL</Data>
<Data Name="NewSd">D:(A;;DCSWRPDTRC;;;BA)(A;;DCSWRPDTRC;;;SY)S:NO_ACCESS_CONTROL</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “change local audit policy security descriptor (SACL)” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change local audit policy security descriptor (SACL)” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Audit Policy Change:

- **Original Security Descriptor** [Type = UnicodeString]: the old Security Descriptor Definition Language (SDDL) value for the audit policy.
- **New Security Descriptor** [Type = UnicodeString]: new Security Descriptor Definition Language (SDDL) value for the audit policy.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

[O:BAG:SYD:\(D;;0xf0007;;;AN\)\(D;;0xf0007;;;BG\)\(A;;0xf0007;;;SY\)\(A;;0x7;;;BA\)S:ARAI\(AU;SAFA;DCLCRPCRSWDWO;;;WD\)](#)

- [O](#): Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators

"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- **G:** = Primary Group.
- **D:** = DACL Entries.
- **S:** = SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: **D:(A;;FA;;;WD)**

- **entry_type:**
 - "D" - DACL
 - "S" - SACL
- **inheritance_flags:**
 - "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.
 - "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" Is not also set.
 - "AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.
- **ace_type:**
 - "A" - ACCESS ALLOWED
 - "D" - ACCESS DENIED
 - "OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).
 - "OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).
 - "AU" - SYSTEM AUDIT

"A" - SYSTEM ALARM
 "OU" - OBJECT SYSTEM AUDIT
 "OL" - OBJECT SYSTEM ALARM

- ace_flags:

"CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.
 "OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.
 "NP" - NO PROPAGATE: only immediate children inherit this ace.
 "IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.
 "ID" - ACE IS INHERITED
 "SA" - SUCCESSFUL ACCESS AUDIT
 "FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A
 - inherit_object_guid: N/A
 - account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

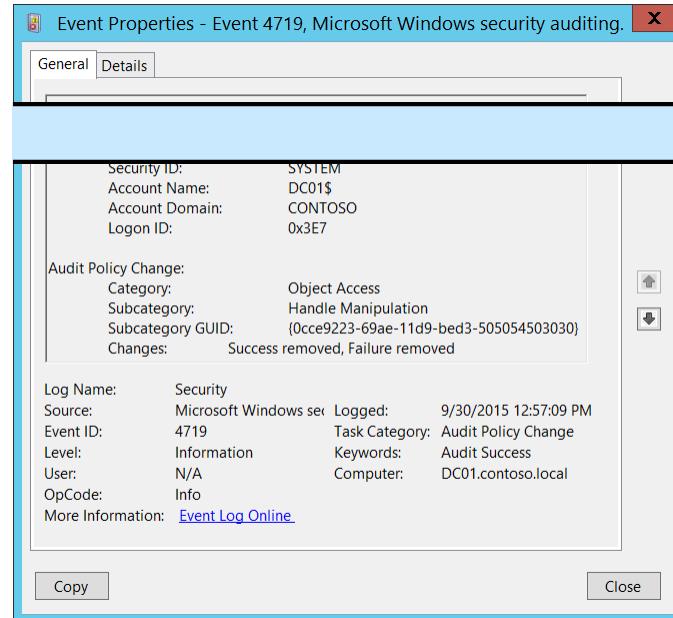
For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,
[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 4715(S): The audit policy (SACL) on an object was changed.

- Monitor for all events of this type, especially on high value assets or computers, because any change of the local audit policy security descriptor should be planned. If this action was not planned, investigate the reason for the change.

4719(S): System audit policy was changed.

 Event Properties - Event 4719, Microsoft Windows security auditing.

General		Details	
Security ID:	SYSTEM		
Account Name:	DC01\$		
Account Domain:	CONTOSO		
Logon ID:	0x3E7		
Audit Policy Change:			
Category:	Object Access		
Subcategory:	Handle Manipulation		
Subcategory GUID:	{0cce9223-69ae-11d9-bed3-505054503030}		
Changes:	Success removed, Failure removed		
Log Name:	Security		
Source:	Microsoft Windows security	Logged:	9/30/2015 12:57:09 PM
Event ID:	4719	Task Category:	Audit Policy Change
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info		
More Information: Event Log Online			
Copy		Close	

Event Description:
This event generates when the computer's audit policy changes.
This event is always logged regardless of the "Audit Policy Change" sub-category setting.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4719</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-09-30T19:57:09.668217100Z" />
<EventRecordID>1049418</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="4668" />

```

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="CategoryId">%%8274</Data>
<Data Name="SubcategoryId">%%12807</Data>
<Data Name="SubcategoryGuid">{0CCE9223-69AE-11D9-BED3-505054503030}</Data>

```

```
<Data Name="AuditPolicyChanges">%&8448, %&8450</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to local audit policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to local audit policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Audit Policy Change:

- **Category:** the name of auditing Category which subcategory was changed. Possible values:
 - Account Logon
 - Account Management
 - Detailed Tracking
 - DS Access
 - Logon/Logoff
 - Object Access
 - Policy Change
 - Privilege Use
 - System
- **Subcategory:** the name of auditing Subcategory which was changed. Possible values:

Credential Validation	Process Termination	Network Policy Server
-----------------------	---------------------	-----------------------

Kerberos Authentication Service	RPC Events	Other Logon/Logoff Events
Kerberos Service Ticket Operations	Detailed Directory Service Replication	Special Logon
Other Logon/Logoff Events	Directory Service Access	Application Generated
Application Group Management	Directory Service Changes	Certification Services
Computer Account Management	Directory Service Replication	Detailed File Share
Distribution Group Management	Account Lockout	File Share
Other Account Management Events	IPsec Extended Mode	File System
Security Group Management	IPsec Main Mode	Filtering Platform Connection
User Account Management	IPsec Quick Mode	Filtering Platform Packet Drop
DPAPI Activity	Logoff	Handle Manipulation
Process Creation	Logon	Kernel Object
Other Object Access Events	Filtering Platform Policy Change	IPsec Driver
Registry	MPSSVC Rule-Level Policy Change	Other System Events
SAM	Other Policy Change Events	Security State Change
Policy Change	Non-Sensitive Privilege Use	Security System Extension
Authentication Policy Change	Sensitive Privilege Use	System Integrity
Authorization Policy Change	Other Privilege Use Events	Plug and Play Events
Group Membership		

- **Subcategory GUID:** the unique subcategory GUID. To see Subcategory GUIDs you can use this command: `auditpol /list /subcategory:*` /v.

Administrator: Command Prompt

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Windows\system32>auditpol /list /subcategory:*
Category/subcategory          GUID
System                          {69979848-797A-11D9-BED3-505054503030}
    Security State Change      {0CCE9210-69AE-11D9-BED3-505054503030}
    Security System Extension  {0CCE9211-69AE-11D9-BED3-505054503030}
    System Integrity           {0CCE9212-69AE-11D9-BED3-505054503030}
    IPsec Driver               {0CCE9213-69AE-11D9-BED3-505054503030}
    Other System Events        {0CCE9214-69AE-11D9-BED3-505054503030}
Logon/Logoff                     {69979849-797A-11D9-BED3-505054503030}
    Logon                      {0CCE9215-69AE-11D9-BED3-505054503030}
    Logoff                     {0CCE9216-69AE-11D9-BED3-505054503030}
    Account Lockout            {0CCE9217-69AE-11D9-BED3-505054503030}
    IPsec Main Mode            {0CCE9218-69AE-11D9-BED3-505054503030}
    IPsec Quick Mode           {0CCE9219-69AE-11D9-BED3-505054503030}
    IPsec Extended Mode        {0CCE921A-69AE-11D9-BED3-505054503030}
    Special Logon              {0CCE921B-69AE-11D9-BED3-505054503030}
    Other Logon/Logoff Events  {0CCE921C-69AE-11D9-BED3-505054503030}
    Network Policy Server      {0CCE9243-69AE-11D9-BED3-505054503030}
    User / Device Claims       {0CCE9247-69AE-11D9-BED3-505054503030}
Object Access                    {6997984A-797A-11D9-BED3-505054503030}
    File System                {0CCE921D-69AE-11D9-BED3-505054503030}
    Registry                   {0CCE921E-69AE-11D9-BED3-505054503030}
    Kernel Object               {0CCE921F-69AE-11D9-BED3-505054503030}
    SAM                        {0CCE9220-69AE-11D9-BED3-505054503030}
    Certification Services     {0CCE9221-69AE-11D9-BED3-505054503030}
```

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

- **Changes:** changes which were made for “**Subcategory**”. Possible values:
 - Success removed
 - Failure removed
 - Success added
 - Failure added

It can be also a combination of any of the items above, separated by coma.

Security Monitoring Recommendations:

For 4719(S): System audit policy was changed.

- Monitor for all events of this type, especially on high value assets or computers, because any change in local audit policy should be planned. If this action was not planned, investigate the reason for the change.

4817(S): Auditing settings on object were changed.

Event Properties - Event 4817, Microsoft Windows security auditing.

General **Details**

Object:

Security ID:	SYSTEM
Account Name:	DC01\$
Account Domain:	CONTOSO
Logon ID:	0x3E7

Auditing Settings:

Original Security Descriptor:	S:(AU;SA;RC;;S-1-5-21-3457937927-2839227994-823803824-1104)
New Security Descriptor:	S:(AU;SA;RC;;S-1-5-21-3457937927-2839227994-823803824-1104)

Log Name: Security
Source: Microsoft Windows security
Event ID: 4817
Level: Information
User: N/A
OpCode: Info

More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates when the [Global Object Access Auditing](#) policy is changed on a computer. Separate events will be generated for "Registry" and "File system" policy changes.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4817</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-10T01:26:33.191368500Z" />
<EventRecordID>1192270</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="3048" />
<Channel>Security</Channel>
```

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="ObjectServer">LSA</Data>
<Data Name="ObjectType">Global SACL</Data>
<Data Name="ObjectName">Key</Data>
<Data Name="OldSd" />
<Data Name="NewSd">S:(AU;SA;RC;;S-1-5-21-3457937927-2839227994-823803824-1104)</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to Global Object Access Auditing policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to Global Object Access Auditing policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString]: has "LSA" value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object to which this event applies. Always "**Global SACL**" for this event.

The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Central Access Policies
Key	WaitablePort	Callback	Global SACL
Job	Port	FilterConnectionPort	
ALPC Port	Semaphore	Adapter	

- **Object Name:**

- Key – if "Registry" Global Object Access Auditing policy was changed.
- File – if "File system" Global Object Access Auditing policy was changed.

Auditing Settings:

- **Original Security Descriptor** [Type = UnicodeString]: the old Security Descriptor Definition Language (SDDL) value for the Global Object Access Auditing policy. Empty if Global Object Access Auditing policy SACL was not set.
- **New Security Descriptor** [Type = UnicodeString]: the new Security Descriptor Definition Language (SDDL) value for the Global Object Access Auditing policy.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

`O:BAG:SYD:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)`

- `O`: Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- `G`: Primary Group.
- `D`: DACL Entries.
- `S`: SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: `D:(A;;FA;;;WD)`

- `entry_type`:

"D" - DACL
 "S" - SACL
 - inheritance_flags:
 "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.
 "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" Is not also set.
 "AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.
 - ace_type:
 "A" - ACCESS ALLOWED
 "D" - ACCESS DENIED
 "OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).
 "OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).
 "AU" - SYSTEM AUDIT
 "A" - SYSTEM ALARM
 "OU" - OBJECT SYSTEM AUDIT
 "OL" - OBJECT SYSTEM ALARM
 - ace_flags:
 "CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.
 "OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.
 "NP" - NO PROPAGATE: only immediate children inherit this ace.
 "IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.
 "ID" - ACE IS INHERITED
 "SA" - SUCCESSFUL ACCESS AUDIT
 "FA" - FAILED ACCESS AUDIT
 - rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes

"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A
- inherit_object_guid: N/A
- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,
[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 4817(S): Auditing settings on object were changed.

- If you use Global Object Access Auditing policies, then this event should be always monitored, especially on high value assets or computers. If this change was not planned, investigate the reason for the change.
- If you don't use Global Object Access Auditing policies, then this event should be always monitored because it indicates use of Global Object Access Auditing policies outside of your standard procedures.

4902(S): The Per-user audit policy table was created.

 Event Properties - Event 4902, Microsoft Windows security audit... X

General Details

Event Description:
This event generates during system startup if Per-user audit policy is defined on the computer.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Number of Elements: 1 Policy ID: 0x703E Log Name: Security Source: Microsoft Windows security audit Event ID: 4902 Level: Information User: N/A OpCode: Info More Information: Event Log Online	 	Event XML: <pre> - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>4902</EventID> <Version>0</Version> <Level>0</Level> <Task>13568</Task> <Opcode>0</Opcode></pre> <pre> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-10-01T00:05:25.814466500Z" /> <EventRecordID>1049490</EventRecordID> <Correlation /> <Execution ProcessID="520" ThreadID="556" /></pre>
---	--	--

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="PuaCount">1</Data>
  <Data Name="PuaPolicyId">0x703e</Data>
</EventData>
</Event>
```

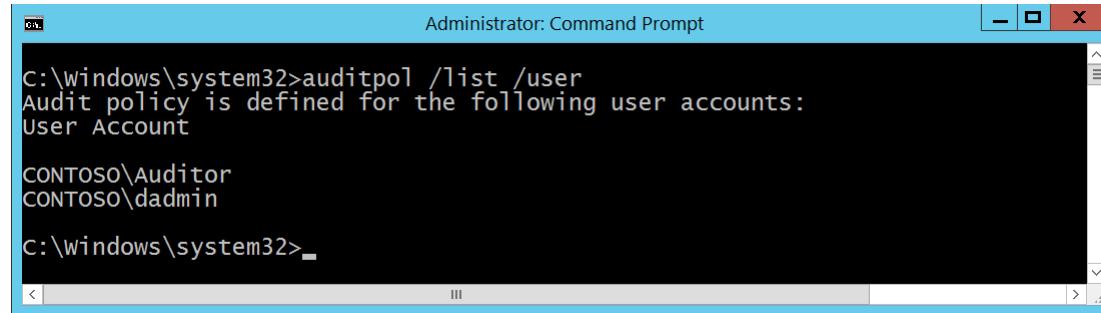
Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

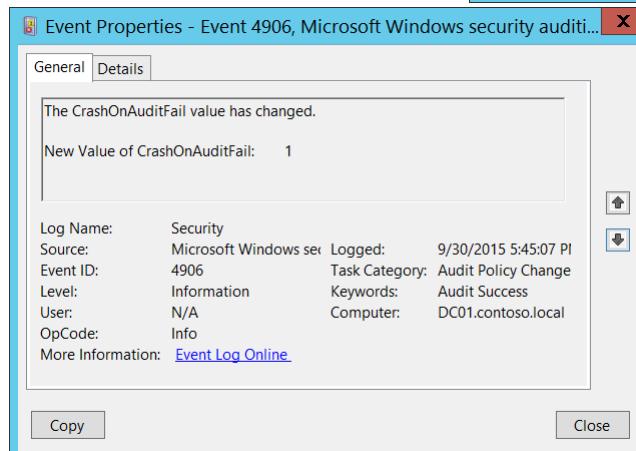
Number of Elements [Type = UInt32]: number of users for which Per-user policies were defined (number of unique users). You can get the list of users for which Per-user policies are defined using "auditpol /list /user" command:



```
C:\windows\system32>auditpol /list /user
Audit policy is defined for the following user accounts:
User Account

CONTOSO\Auditor
CONTOSO\dadmin

C:\windows\system32>_
```



Policy ID [Type = HexInt64]: unique per-User Audit Policy hexadecimal identifier.

Security Monitoring Recommendations:

For 4902(S): The Per-user audit policy table was created.

- If you don't expect to see any per-User Audit Policies enabled on specific computers (**Computer**), monitor for these events.
- If you don't use per-User Audit Policies in your network, monitor for these events.
- Typically this is an informational event and has little to no security relevance.

4906(S): The CrashOnAuditFail value has changed.

Event Description:

This event generates every time **CrashOnAuditFail** audit flag value was modified.

This event is always logged regardless of the "Audit Policy Change" sub-category setting.

More information about **CrashOnAuditFail** flag can be found [here](#).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4906</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T00:45:07.048458800Z" />
<EventRecordID>1049529</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="532" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="CrashOnAuditFailValue">1</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

New Value of CrashOnAuditFail [Type = UInt32]: contains new value of **CrashOnAuditFail** flag. Possible values are:

- 0 - The feature is off. The system does not halt, even when it cannot record events in the Security Log.
- 1 - The feature is on. The system halts when it cannot record an event in the Security Log.
- 2 - The feature is on and has been triggered. The system halted because it could not record an auditable event in the Security Log. Only members of the Administrators group can log on.

Security Monitoring Recommendations:

For 4906(S): The CrashOnAuditFail value has changed.

- Any changes of **CrashOnAuditFail** audit flag that are reported by this event must be monitored, and an alert should be triggered. If this change was not planned, investigate the reason for the change.

4907(S): Auditing settings on object were changed.

Event Properties - Event 4907, Microsoft Windows security auditing.

General Details

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x138EB0
Object:	
Object Server:	Security
Object Type:	Key
Object Name:	\REGISTRY\MACHINE\SYSTEM\ControlSet001\Services\EventLog\Internet Explorer
Handle ID:	0x2f8
Process Information:	
Process ID:	0x120c
Process Name:	C:\Windows\regedit.exe
Auditing Settings:	
Original Security Descriptor:	S:AI
New Security Descriptor:	S:ARAI(AU;CISA;KA;;;S-1-5-21-3457937927-2839227994-823803824-1104)
Log Name:	Security
Source:	Microsoft Windows se
Event ID:	4907
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy **Close**

Event Description:

This event generates when the [SACL](#) of an object (for example, a registry key or file) was changed. This event doesn't generate for Active Directory objects.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-
A5BA-3E3B0328C30D}" />
  <EventID>4907</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13568</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-10-01T18:18:19.458828800Z" />
  <EventRecordID>1049732</EventRecordID>
  <Correlation />
  <Execution ProcessID="500" ThreadID="508" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
```

```

- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x138eb0</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">Key</Data>
<Data Name="ObjectName">\REGISTRY\MACHINE\SYSTEM\ControlSet001\Services\EventLog\Internet Explorer</Data>
<Data Name="HandleId">0x2f8</Data>
```

```
<Data Name="OldSd">S:AI</Data>
<Data Name="NewSd">S:ARAI(AU;CISA;KA;;S-1-5-21-3457937927-2839227994-823803824-1104)</Data>
<Data Name="ProcessId">0x120c</Data>
<Data Name="ProcessName">C:\Windows\regedit.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to object's auditing settings. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to object's auditing settings.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

Object:

- **Object Server** [Type = UnicodeString]: has "Security" value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation.

The following table contains the list of the most common **Object Types**:

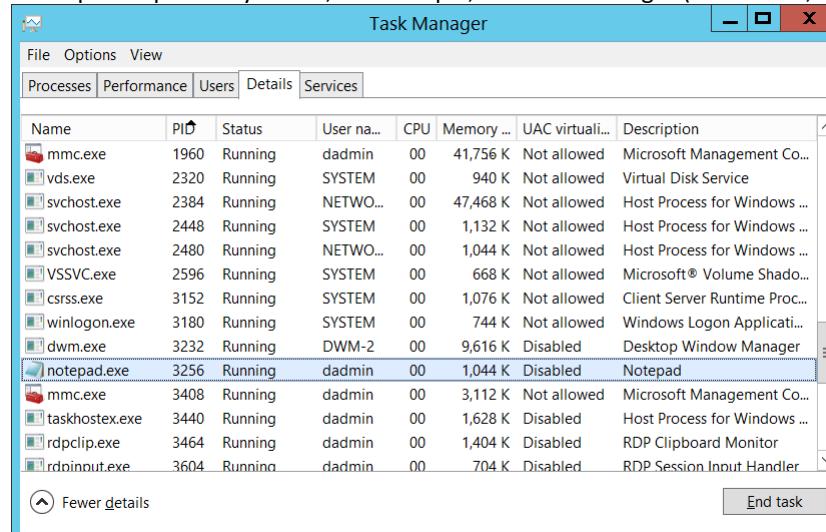
Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	SC_MANAGER OBJECT
Key	WaitablePort	Callback	

Job	Port	FilterConnectionPort	
ALPC Port	Semaphore	Adapter	

- **Object Name** [Type = UnicodeString]: full path and name of the object for which the [SACL](#) was modified. Depends on **Object Type**. Here are some examples:
 - The format for **Object Type** = “Key” is: \REGISTRY\HIVE\PATH where:
 - HIVE:
 - HKEY_LOCAL_MACHINE = \REGISTRY\MACHINE
 - HKEY_CURRENT_USER = \REGISTRY\USER\[USER_SID], where [USER_SID] is the SID of current user.
 - HKEY_CLASSES_ROOT = \REGISTRY\MACHINE\SOFTWARE\Classes
 - HKEY_USERS = \REGISTRY\USER
 - HKEY_CURRENT_CONFIG = \REGISTRY\MACHINE\SYSTEM\ControlSet001\Hardware Profiles\Current
 - PATH – path to the registry key.
 - The format for **Object Type** = “File” is: full path and name of the file or folder for which [SACL](#) was modified.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4656](#): A handle to an object was requested.” Event for registry keys or with **Handle ID** field in “[4656](#)(S, F): A handle to an object was requested.” Event for file system objects. This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the object’s [SACL](#) was changed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Auditing Settings:

- **Original Security Descriptor** [Type = UnicodeString]: the old Security Descriptor Definition Language (SDDL) value for the object.
- **New Security Descriptor** [Type = UnicodeString]: the new Security Descriptor Definition Language (SDDL) value for the object.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

O:BAG:SYD:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)

- **O**: = Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- **G**: = Primary Group.

- **D**: = DACL Entries.

- **S**: = SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: **D:(A;;FA;;;WD)**

- `entry_type`:

"D" - DACL

"S" - SACL

- inheritance_flags:

"P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.

"AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" is not also set.

"AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.

- ace_type:

"A" - ACCESS ALLOWED

"D" - ACCESS DENIED

"OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).

"OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).

"AU" - SYSTEM AUDIT

"A" - SYSTEM ALARM

"OU" - OBJECT SYSTEM AUDIT

"OL" - OBJECT SYSTEM ALARM

- ace_flags:

"CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.

"OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.

"NP" - NO PROPAGATE: only immediate children inherit this ace.

"IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.

"ID" - ACE IS INHERITED

"SA" - SUCCESSFUL ACCESS AUDIT

"FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object

"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A
- inherit_object_guid: N/A
- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,
[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

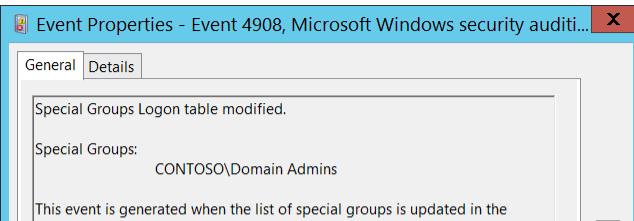
Security Monitoring Recommendations:

For 4907(S): Auditing settings on object were changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
- If you need to monitor events related to specific Windows object types ("Object Type"), for example File or Key, monitor this event for the corresponding "Object Type." If you need to monitor all SACL changes for specific files, folders, registry keys, or other object types, monitor for "**Object Name**" field value which has specific object name.
- If you have critical file or registry objects and you need to monitor all modifications (especially changes in SACL), monitor for specific "**Object\Object Name**".
- If you have high-value computers for which you need to monitor all changes for all or specific file or registry objects, monitor for all [4907](#) events on these computers.

4908(S): Special Groups Logon table modified.

 Event Properties - Event 4908, Microsoft Windows security audit... X

General Details

Special Groups Logon table modified.

Special Groups:
CONTOSO\Domain Admins

This event is generated when the list of special groups is updated in the registry or through security policy. The updated list of special groups is

Source: Microsoft Windows sec	Logged: 9/30/2015 5:20:40 PM
Event ID: 4908	Task Category: Audit Policy Change
Level: Information	Keywords: Audit Success
User: N/A	Computer: DC01.contoso.local
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time Special Groups logon table was modified.

This event also generates during system startup.

This event is always logged regardless of the "Audit Policy Change" sub-category setting.

More information about Special Groups auditing can be found here:

<http://blogs.technet.com/b/askds/archive/2008/03/11/special-groups-auditing-via-group-policy-preferences.aspx>

<https://support.microsoft.com/en-us/kb/947223>

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4908</EventID>
<Version>0</Version>
```

<Level>0</Level>

```

<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T00:20:40.210246600Z" />
<EventRecordID>1049511</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="532" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SidList">%{S-1-5-21-3457937927-2839227994-823803824-512}</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

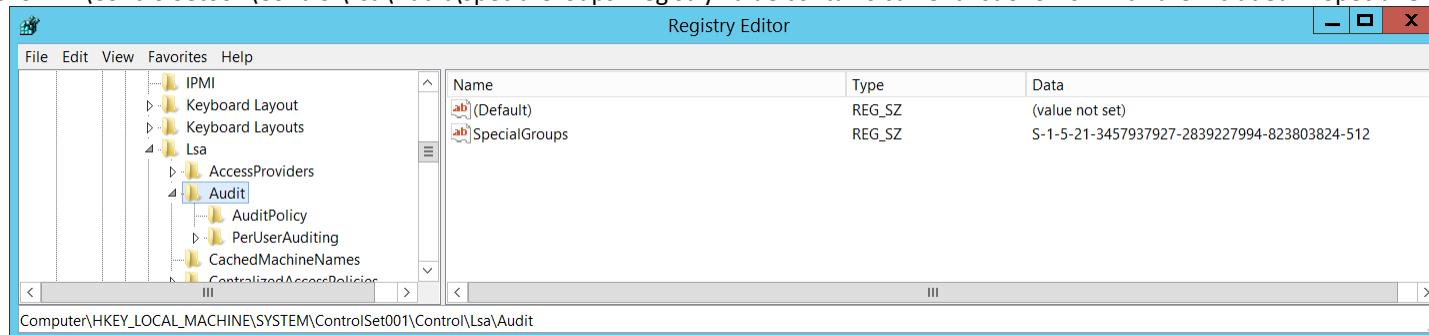
Event Versions: 0.

Field Descriptions:

Special Groups [Type = UnicodeString]: contains current list of SIDs (groups or accounts) which are members of Special Groups. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

"HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Lsa\Audit\SpecialGroups" registry value contains current list of SIDs which are included in Special Groups:

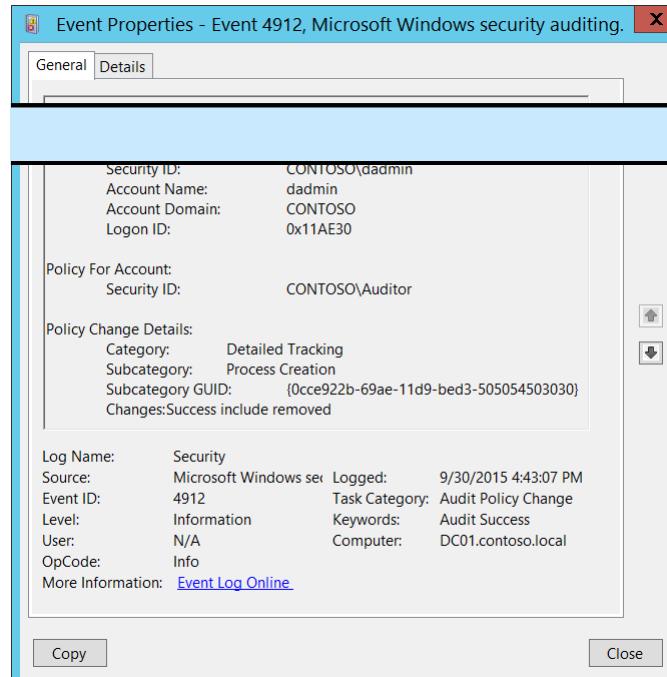


Security Monitoring Recommendations:

For 4908(S): Special Groups Logon table modified.

- If you use the Special Groups feature, then this event should be always monitored, especially on high value assets or computers. If this change was not planned, investigate the reason for the change.
- If you don't use the Special Groups feature, then this event should be always monitored because it indicates use of the Special Groups feature outside of your standard procedures.

4912(S): Per User Audit Policy was changed.

 Event Properties - Event 4912, Microsoft Windows security auditing. X

General Details

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x11AE30
Policy For Account:	
Security ID:	CONTOSO\Auditor
Policy Change Details:	
Category:	Detailed Tracking
Subcategory:	Process Creation
Subcategory GUID:	{0cce922b-69ae-11d9-bed3-505054503030}
Changes:	Success include removed
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4912
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time [Per User Audit Policy](#) was changed.

This event is always logged regardless of the "Audit Policy Change" sub-category setting.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4912</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13568</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-09-30T23:43:07.363195100Z" />
  <EventRecordID>1049452</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="1660" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>

```

```

<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x11ae30</Data>
<Data Name="TargetUserSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>

```

```
<Data Name="CategoryId">%&8276</Data>
<Data Name="SubcategoryId">%&13312</Data>
<Data Name="SubcategoryGuid">{0CCE922B-69AE-11D9-BED3-505054503030}</Data>
<Data Name="AuditPolicyChanges">%&8452</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to per-user audit policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to per-user audit policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Policy For Account:

- **Security ID** [Type = SID]: SID of account for which the Per User Audit Policy was changed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Policy Change Details:

- **Category** [Type = UnicodeString]: the name of auditing category which subcategory state was changed. Possible values are:
 - Account Logon
 - Account Management
 - Detailed Tracking
 - DS Access
 - Logon/Logoff

- Object Access
 - Policy Change
 - Privilege Use
 - System
- **Subcategory** [Type = UnicodeString]: the name of auditing subcategory which state was changed. Possible values:

Audit Credential Validation	Audit Process Termination	Audit Other Logon/Logoff Events
Audit Kerberos Authentication Service	Audit RPC Events	Audit Special Logon
Audit Kerberos Service Ticket Operations	Audit Detailed Directory Service Replication	Audit Application Generated
Audit Other Logon/Logoff Events	Audit Directory Service Access	Audit Certification Services
Audit Application Group Management	Audit Directory Service Changes	Audit Detailed File Share
Audit Computer Account Management	Audit Directory Service Replication	Audit File Share
Audit Distribution Group Management	Audit Account Lockout	Audit File System
Audit Other Account Management Events	Audit IPsec Extended Mode	Audit Filtering Platform Connection
Audit Security Group Management	Audit IPsec Main Mode	Audit Filtering Platform Packet Drop
Audit User Account Management	Audit IPsec Quick Mode	Audit Handle Manipulation
Audit DPAPI Activity	Audit Logoff	Audit Kernel Object
Audit Process Creation	Audit Logon	Audit IPsec Driver
Audit Other Object Access Events	Audit Filtering Platform Policy Change	Audit Other System Events
Audit Registry	Audit MPSSVC Rule-Level Policy Change	Audit Security State Change
Audit SAM	Audit Other Policy Change Events	Audit Security System Extension
Audit Policy Change	Audit Non-Sensitive Privilege Use	Audit System Integrity
Audit Authentication Policy Change	Audit Sensitive Privilege Use	Audit PNP Activity
Audit Authorization Policy Change	Audit Other Privilege Use Events	
Group Membership	Audit Network Policy Server	

- **Subcategory GUID** [Type = GUID]: the unique GUID of changed subcategory.

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

To see subcategory GUID you can use the following command: "**auditpol /list /subcategory:*** /v":

Administrator: Command Prompt

```
C:\Windows\system32>auditpol /list /subcategory:*
Category/Subcategory          GUID
System                          {69979848-797A-11D9-BED3-505054503030}
    Security State Change      {0CCE9210-69AE-11D9-BED3-505054503030}
    Security System Extension  {0CCE9211-69AE-11D9-BED3-505054503030}
    System Integrity           {0CCE9212-69AE-11D9-BED3-505054503030}
    IPsec Driver                {0CCE9213-69AE-11D9-BED3-505054503030}
    Other System Events        {0CCE9214-69AE-11D9-BED3-505054503030}
Logon/Logoff                    {69979849-797A-11D9-BED3-505054503030}
    Logon                         {0CCE9215-69AE-11D9-BED3-505054503030}
    Logoff                        {0CCE9216-69AE-11D9-BED3-505054503030}
    Account Lockout              {0CCE9217-69AE-11D9-BED3-505054503030}
    IPsec Main Mode              {0CCE9218-69AE-11D9-BED3-505054503030}
    IPsec Quick Mode             {0CCE9219-69AE-11D9-BED3-505054503030}
    IPsec Extended Mode          {0CCE921A-69AE-11D9-BED3-505054503030}
    Special Logon                 {0CCE921B-69AE-11D9-BED3-505054503030}
    Other Logon/Logoff Events    {0CCE921C-69AE-11D9-BED3-505054503030}
    Network Policy Server         {0CCE9243-69AE-11D9-BED3-505054503030}
```

- **Changes** [Type = UnicodeString]: changes which were made for the subcategory. Possible values are:
 - Success include removed

Event Properties - Event 4904, Microsoft Windows security audit...

General Details

An attempt was made to register a security event source.

Subject:

- Security ID: SYSTEM
- Account Name: DC01\$
- Account Domain: CONTOSO
- Logon ID: 0x3E7

Process:

- Process ID: 0x688
- Process Name: C:\Windows\System32\svchost.exe

Event Source:

- Source Name: FSRM Audit
- Event Source ID: 0x1CC4E

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4904
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

- Success include added
- Failure include removed
- Failure include added
- Success exclude removed
- Success exclude added
- Failure exclude removed
- Failure exclude added

Security Monitoring Recommendations:

For 4912(S): Per User Audit Policy was changed.

- If you use the Per-user audit feature, then this event should be always monitored, especially on high value assets or computers. If this change was not planned, investigate the reason for the change.
- If you don't use the Per-user audit feature, then this event should be always monitored because it indicates use of the Per-user audit feature outside of your standard procedures.

4904(S): An attempt was made to register a security event source.

Event Description:

This event generates every time a new [security event source](#) is registered.

You can typically see this event during system startup, if specific roles (Internet Information Services, for example) are installed in the system.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4904</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T00:53:01.030688000Z" />
<EventRecordID>1049538</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="548" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="AuditSourceName">FSRM Audit</Data>
<Data Name="EventSourceId">0xcc4e</Data>
<Data Name="ProcessId">0x688</Data>
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

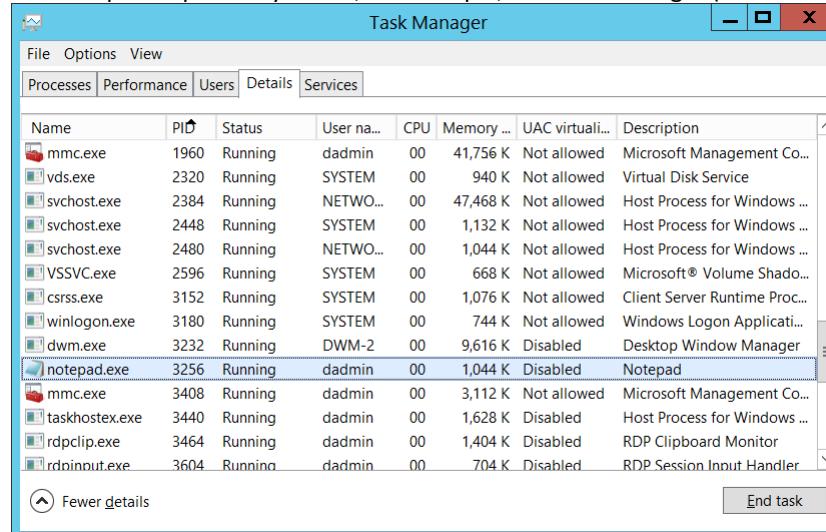
- **Security ID [Type = SID]**: SID of account that made an attempt to register a security event source. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name [Type = UnicodeString]**: the name of the account that made an attempt to register a security event source.
- **Account Domain [Type = UnicodeString]**: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID [Type = HexInt64]**: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Process:

- **Process ID [Type = Pointer]**: hexadecimal Process ID of the process that attempted to register the security event source. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



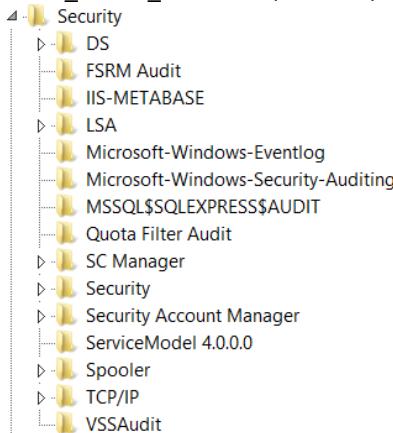
If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, "[4688](#): A new process has been created" **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Event Source:

- **Source Name** [Type = UnicodeString]: the name of registered security event source. You can see all registered security event source names in this registry path: "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Security". Here is an example:



- **Event Source ID** [Type = HexInt64]: the unique hexadecimal identifier of registered security event source.

 Event Properties - Event 4905, Microsoft Windows security auditi... X

General		Details																
<p>An attempt was made to register a security event source.</p> <p>Appendix A: Security monitoring recommendations for many audit events</p>																		
<p>Subject:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">Security ID:</td> <td>SYSTEM</td> </tr> <tr> <td>Account Name:</td> <td>DC01\$</td> </tr> <tr> <td>Account Domain:</td> <td>CONTOSO</td> </tr> <tr> <td>Logon ID:</td> <td>0x3E7</td> </tr> </table> <p>Process:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">Process ID:</td> <td>0xd90</td> </tr> <tr> <td>Process Name:</td> <td>-</td> </tr> </table> <p>Event Source:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">Source Name:</td> <td>IIS-METABASE</td> </tr> <tr> <td>Event Source ID:</td> <td>0x20C15F</td> </tr> </table> <p>Log Name: Security Source: Microsoft Windows security Event ID: 4905 Level: Information User: N/A OpCode: Info More Information: Event Log Online</p>			Security ID:	SYSTEM	Account Name:	DC01\$	Account Domain:	CONTOSO	Logon ID:	0x3E7	Process ID:	0xd90	Process Name:	-	Source Name:	IIS-METABASE	Event Source ID:	0x20C15F
Security ID:	SYSTEM																	
Account Name:	DC01\$																	
Account Domain:	CONTOSO																	
Logon ID:	0x3E7																	
Process ID:	0xd90																	
Process Name:	-																	
Source Name:	IIS-METABASE																	
Event Source ID:	0x20C15F																	
Copy	Close																	

Security Monitoring Recommendations:
For 4904(S): An attempt was made to register a security event source.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever "Subject\Security ID" is not SYSTEM.

-
-
- If you have a pre-defined "Process Name" for the process reported in this event, monitor all events with "Process Name" not equal to your defined value.
- If you have a pre-defined list of allowed security event sources for specific computers or computer types, then you can use this event and check whether "Event Source\Source Name" is in your defined list.
- Typically this event has an informational purpose.

4905(S): An attempt was made to unregister a security event source.

Event Description:
This event generates every time a [security event source](#) is unregistered.
You typically see this event if specific roles were removed, for example, Internet Information Services.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4905</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13568</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T17:39:12.039825000Z" />
<EventRecordID>1049718</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="1888" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="AuditSourceName">IIS-METABASE</Data>
<Data Name="EventSourceId">0x20c15f</Data>
<Data Name="ProcessId">0xd90</Data>
<Data Name="ProcessName">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

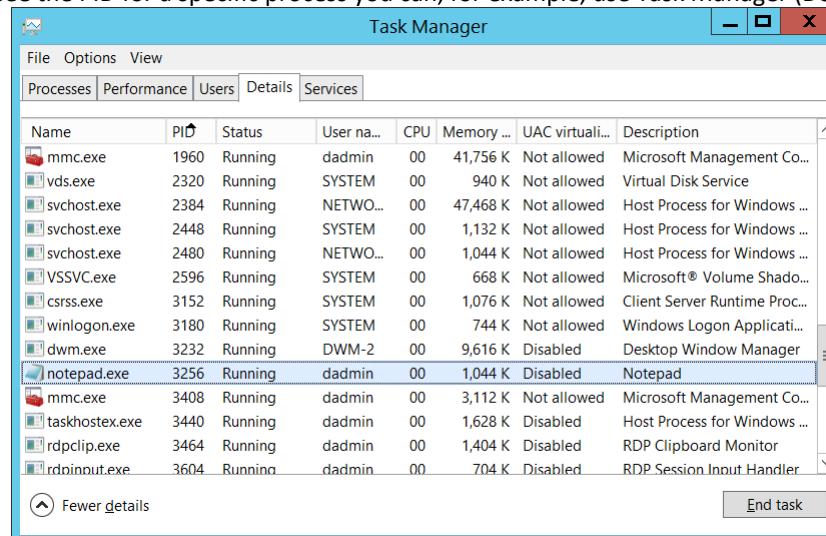
- **Security ID** [Type = SID]: SID of account that made an attempt to unregister a security event source. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made an attempt to unregister a security event source.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted to unregister the security event source. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



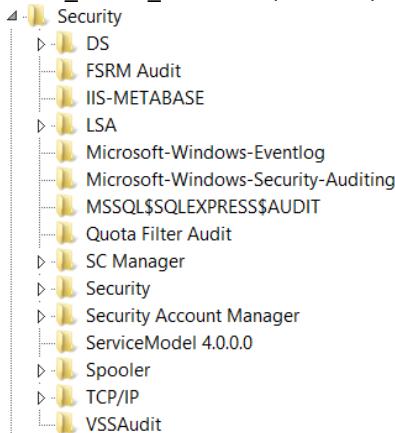
If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, "[4688](#): A new process has been created" **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Event Source:

- **Source Name** [Type = UnicodeString]: the name of unregistered security event source. You can see all registered security event source names in this registry path: "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Security". Here is an example:



- **Event Source ID** [Type = HexInt64]: the unique hexadecimal identifier of unregistered security event source.

Security Monitoring Recommendations:

For 4905(S): An attempt was made to unregister a security event source.

[**Appendix A: Security monitoring recommendations for many audit events**](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever "**Subject\Security ID**" is not SYSTEM.
-
-
- If you have a pre-defined "Process Name" for the process reported in this event, monitor all events with "Process Name" not equal to your defined value.
- If you have a list of critical security event sources which should never have been unregistered, then you can use this event and check the "**Event Source\Source Name**."
- Typically this event has an informational purpose.

Audit Authentication Policy Change

Audit Authentication Policy Change determines whether the operating system generates audit events when changes are made to authentication policy.

Changes made to authentication policy include:

- Creation, modification, and removal of forest and domain trusts.
- Changes to Kerberos policy under Computer Configuration\Windows Settings\Security Settings\Account Policies\Kerberos Policy.
- When any of the following user logon rights is granted to a user or group:
 - Access this computer from the network
 - Allow logon locally
 - Allow logon through Remote Desktop
 - Logon as a batch job
 - Logon as a service
- Namespace collision, such as when an added trust collides with an existing namespace name.

This setting is useful for tracking changes in domain-level and forest-level trust and privileges that are granted to user accounts or groups.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>On domain controllers, it is important to enable Success audit for this subcategory to be able to get information related to operations with domain and forest trusts, changes in Kerberos policy and some other events included in this subcategory.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	Yes	No	Yes	No	<p>On member servers it is important to enable Success audit for this subcategory to be able to get information related to changes in user logon rights policies and password policy changes.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	Yes	No	Yes	No	<p>On workstations it is important to enable Success audit for this subcategory to be able to get information related to changes in user logon rights policies and password policy changes.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

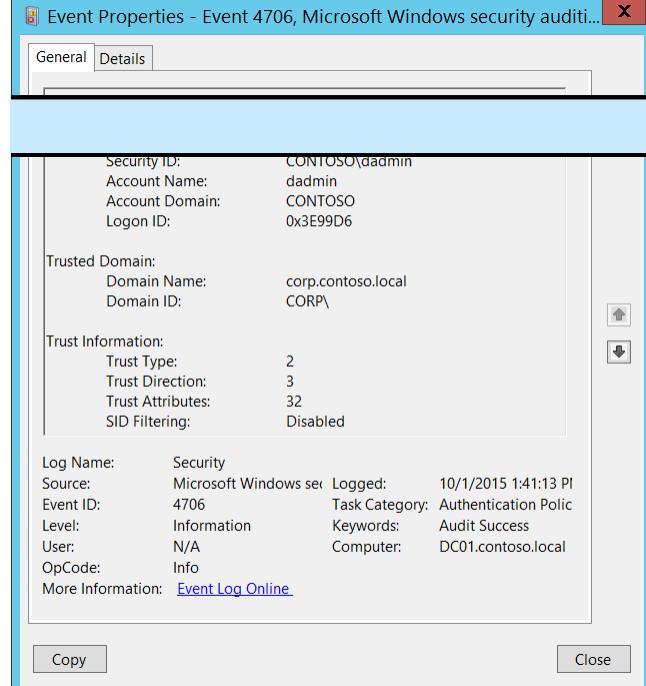
- [4670\(S\)](#): Permissions on an object were changed
- [4706\(S\)](#): A new trust was created to a domain.
- [4707\(S\)](#): A trust to a domain was removed.
- [4716\(S\)](#): Trusted domain information was modified.
- [4713\(S\)](#): Kerberos policy was changed.

- [4717\(S\)](#): System security access was granted to an account.
- [4718\(S\)](#): System security access was removed from an account.
- [4739\(S\)](#): Domain Policy was changed.
- [4864\(S\)](#): A namespace collision was detected.
- [4865\(S\)](#): A trusted forest information entry was added.
- [4866\(S\)](#): A trusted forest information entry was removed.
- [4867\(S\)](#): A trusted forest information entry was modified.

4670(S): Permissions on an object were changed.

This event also belongs in the **Audit File System** subcategory, and is described there. See "[4670\(S\): Permissions on an object were changed.](#)"

4706(S): A new trust was created to a domain.

 Event Properties - Event 4706, Microsoft Windows security audit... X

General	Details
---------	---------

Event Description:
This event generates when new trust was created to a domain.
This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4706</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T20:41:13.189445500Z" />
<EventRecordID>1049759</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4900" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```

<Security />
</System>
- <EventData>
<Data Name="DomainName">corp.contoso.local</Data>
```

```
<Data Name="DomainSid">S-1-5-21-2226861337-2836268956-2433141405</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e99d6</Data>
<Data Name="TdoType">2</Data>
<Data Name="TdoDirection">3</Data>
<Data Name="TdoAttributes">32</Data>
<Data Name="SidFilteringEnabled">%%1796</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “create domain trust” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “create domain trust” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Trusted Domain:

- **Domain Name** [Type = UnicodeString]: the name of new trusted domain.
- **Domain ID** [Type = SID]: SID of new trusted domain. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Trust Information:

- **Trust Type** [Type = UInt32]: the type of new trust. The following table contains possible values for this field:

Value	Attribute Value	Description
1	TRUST_TYPE_DOWNLEVEL	The domain controller of the trusted domain is a computer running an operating system earlier than Windows 2000.
2	TRUST_TYPE_UPLEVEL	The domain controller of the trusted domain is a computer running Windows 2000 or later.
3	TRUST_TYPE_MIT	The trusted domain is running a non-Windows, RFC4120-compliant Kerberos distribution. This type of trust is distinguished in that (1) a SID is not required for the TDO , and (2) the default key types include the DES-CBC and DES-CRC encryption types (see [RFC4120] section 8.1).
4	TRUST_TYPE_DCE	The trusted domain is a DCE realm. Historical reference, this value is not used in Windows.

- **Trust Direction** [Type = UInt32]: the direction of new trust. The following table contains possible values for this field:

Value	Attribute Value	Description
0	TRUST_DIRECTION_DISABLED	The trust relationship exists, but it has been disabled.
1	TRUST_DIRECTION_INBOUND	The trusted domain trusts the primary domain to perform operations such as name lookups and authentication.
2	TRUST_DIRECTION_OUTBOUND	The primary domain trusts the trusted domain to perform operations such as name lookups and authentication.
3	TRUST_DIRECTION_BIDIRECTIONAL	Both domains trust one another for operations such as name lookups and authentication.

- **Trust Attributes** [Type = UInt32]: the decimal value of attributes for new trust. You need convert decimal value to hexadecimal and find it in the table below. The following table contains possible values for this field:

Value	Attribute Value	Description
0x1	TRUST_ATTRIBUTE_NON_TRANSITIVE	If this bit is set, then the trust cannot be used transitively. For example, if domain A trusts domain B, which in turn trusts domain C, and the A<-->B trust has this attribute set, then a client in domain A cannot authenticate to a server in domain C over the A<-->B<-->C trust linkage.
0x2	TRUST_ATTRIBUTE_UPLEVEL_ONLY	If this bit is set in the attribute, then only Windows 2000 operating system and newer clients may use the trust link. Netlogon does not consume trust objects that have this flag set.
0x4	TRUST_ATTRIBUTE_QUARANTINED_DOMAIN	If this bit is set, the trusted domain is quarantined and is subject to the rules of SID Filtering as described in [MS-PAC] section 4.1.2.2 .
0x8	TRUST_ATTRIBUTE_FOREST_TRANSITIVE	If this bit is set, the trust link is a cross-forest trust [MS-KILE] between the root domains of two forests , both of which are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater. Only evaluated on Windows Server 2003 operating system, Windows Server 2008 operating system, Windows Server 2008 R2 operating system, Windows Server 2012 operating system, Windows Server 2012 R2 operating system, and Windows Server 2016 Technical Preview operating system. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.
0x10	TRUST_ATTRIBUTE_CROSS_ORGANIZATION	If this bit is set, then the trust is to a domain or forest that is not part of the organization . The behavior controlled by this bit is explained in [MS-KILE] section 3.3.5.7.5 and [MS-APDS] section 3.1.5 . Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.

		greater.
0x20	TRUST_ATTRIBUTE_WITHIN_FOREST	If this bit is set, then the trusted domain is within the same forest. Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview.
0x40	TRUST_ATTRIBUTE_TREAT_AS_EXTERNAL	If this bit is set, then a cross-forest trust to a domain is to be treated as an external trust for the purposes of SID Filtering. Cross-forest trusts are more stringently filtered than external trusts. This attribute relaxes those cross-forest trusts to be equivalent to external trusts. For more information on how each trust type is filtered, see [MS-PAC] section 4.1.2.2. Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview. Only evaluated if SID Filtering is used. Only evaluated on cross-forest trusts having TRUST_ATTRIBUTE_FOREST_TRANSITIVE. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.
0x80	TRUST_ATTRIBUTEUSES_RC4_ENCRYPTION	This bit is set on trusts with the trustType set to TRUST_TYPE_MIT, which are capable of using RC4 keys. Historically, MIT Kerberos distributions supported only DES and 3DES keys ([RFC4120] , [RFC3961]). MIT 1.4.1 adopted the RC4HMAC encryption type common to Windows 2000 [MS-KILE] , so trusted domains deploying later versions of the MIT distribution required this bit. For more information, see "Keys and Trusts", section 6.1.6.9.1 . Only evaluated on TRUST_TYPE_MIT
0x200	TRUST_ATTRIBUTE_CROSS_ORGANIZATION_NO_TGT_DELEGATION	If this bit is set, tickets granted under this trust MUST NOT be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5. Only supported on Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview.
0x400	TRUST_ATTRIBUTE_PIM_TRUST	If this bit and the TATE bit are set, then a cross-forest trust to a domain is to be treated as Privileged Identity Management trust for the purposes of SID Filtering. For more information on how each trust type is filtered, see [MS-PAC] section 4.1.2.2. Evaluated only on Windows Server 2016 Technical Preview Evaluated only if SID Filtering is used. Evaluated only on cross-forest trusts having TRUST_ATTRIBUTE_FOREST_TRANSITIVE. Can be set only if the forest and the trusted forest are running in a forest functional level of DS_BEHAVIOR_WINTHRESHOLD or greater.

- **SID Filtering** [Type = UnicodeString]: [SID Filtering](#) state for the new trust:

- Enabled
- Disabled

Security Monitoring Recommendations:

For 4706(S): A new trust was created to a domain.

- Any changes related to Active Directory domain trusts (especially creation of the new trust) must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.

4707(S): A trust to a domain was removed.

Event Properties - Event 4707, Microsoft Windows security auditi... X

General Details

Security ID:	CONTOSO\damain
Account Name:	damain
Account Domain:	CONTOSO
Logon ID:	0x3E99D6
Domain Information:	
Domain Name:	FABRIKAM
Domain ID:	CORP\
Log Name:	Security
Source:	Microsoft Windows se
Event ID:	4707
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy Close

Event Description:

This event generates when a domain trust was removed.
This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4707</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T20:41:13.080444700Z" />
<EventRecordID>1049754</EventRecordID>
```

```
<Correlation />
<Execution ProcessID="500" ThreadID="580" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="DomainName">FABRIKAM</Data>
<Data Name="DomainSid">S-1-5-21-2226861337-2836268956-2433141405</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">damain</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e99d6</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “remove domain trust” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “remove domain trust” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

 Event Properties - Event 4716, Microsoft Windows security audit... X

General		Details	
Trusted domain information was modified.			
Subject:		Security ID: CONTOSO\dadmin Account Name: dadmin Account Domain: CONTOSO Logon ID: 0x138EB0	
Trusted Domain:		Domain Name: - Domain ID: CORP\	
New Trust Information:		Trust Type: 2 Trust Direction: 3 Trust Attributes: 32 SID Filtering: -	

Domain Information:

- **Domain Name** [Type = UnicodeString]: the name of removed trusted domain.
- **Domain ID** [Type = SID]: SID of removed trusted domain. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Security Monitoring Recommendations:

For 4707(S): A trust to a domain was removed.

- Any changes related to Active Directory domain trusts (especially trust removal) must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.

4716(S): Trusted domain information was modified.
Event Description:

This event generates when the trust was modified.

This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event ID:	4716	Task Category:	Authentication Policy
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info	More Information: Event Log Online	
<input type="button" value="Copy"/>		<input type="button" value="Close"/>	

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
```

```
<EventID>4716</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T22:55:54.560735500Z" />
<EventRecordID>1049763</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4920" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x138eb0</Data>
  <Data Name="DomainName">-</Data>
  <Data Name="DomainSid">S-1-5-21-2226861337-2836268956-2433141405</Data>
  <Data Name="TdoType">2</Data>
  <Data Name="TdoDirection">3</Data>
  <Data Name="TdoAttributes">32</Data>
  <Data Name="SidFilteringEnabled">-</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested the “modify domain trust settings” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it

in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “modify domain trust settings” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Trusted Domain:

- **Domain Name** [Type = UnicodeString]: the name of changed trusted domain. If this attribute was not changed, then it will have “-“ value.
- **Domain ID** [Type = SID]: SID of changed trusted domain. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

New Trust Information:

- **Trust Type** [Type = UInt32]: the type of new trust. If this attribute was not changed, then it will have “-“ value or its old value. The following table contains possible values for this field:

Value	Attribute Value	Description
1	TRUST_TYPE_DOWNLEVEL	The domain controller of the trusted domain is a computer running an operating system earlier than Windows 2000.
2	TRUST_TYPE_UPLEVEL	The domain controller of the trusted domain is a computer running Windows 2000 or later.
3	TRUST_TYPE_MIT	The trusted domain is running a non-Windows, RFC4120-compliant Kerberos distribution. This type of trust is distinguished in that (1) a SID is not required for the TDO , and (2) the default key types include the DES-CBC and DES-CRC encryption types (see [RFC4120] section 8.1).
4	TRUST_TYPE_DCE	The trusted domain is a DCE realm. Historical reference, this value is not used in Windows.

- **Trust Direction** [Type = UInt32]: the direction of new trust. If this attribute was not changed, then it will have “-“ value or its old value. The following table contains possible values for this field:

Value	Attribute Value	Description
0	TRUST_DIRECTION_DISABLED	The trust relationship exists, but it has been disabled.
1	TRUST_DIRECTION_INBOUND	The trusted domain trusts the primary domain to perform operations such as name lookups and authentication.
2	TRUST_DIRECTION_OUTBOUND	The primary domain trusts the trusted domain to perform operations such as name lookups and authentication.
3	TRUST_DIRECTION_BIDIRECTIONAL	Both domains trust one another for operations such as name lookups and authentication.

- **Trust Attributes** [Type = UInt32]: the decimal value of attributes for new trust. You need convert decimal value to hexadecimal and find it in the table below. If this attribute was not changed, then it will have “-“ value or its old value. The following table contains possible values for this field:

Value	Attribute Value	Description
0x1	TRUST_ATTRIBUTE_NON_TRANSITIVE	If this bit is set, then the trust cannot be used transitively. For example, if domain A trusts domain B, which in turn trusts domain C, and the A<-->B trust has this attribute set, then a client in domain A cannot authenticate to a server in domain C over the A<-->B<-->C trust linkage.
0x2	TRUST_ATTRIBUTE_UPLEVEL_ONLY	If this bit is set in the attribute, then only Windows 2000 operating system and newer clients may use the trust link. Netlogon does not consume trust objects that have this flag set.
0x4	TRUST_ATTRIBUTE_QUARANTINED_DOMAIN	If this bit is set, the trusted domain is quarantined and is subject to the rules of SID Filtering as described in [MS-PAC] section 4.1.2.2 .
0x8	TRUST_ATTRIBUTE_FOREST_TRANSITIVE	If this bit is set, the trust link is a cross-forest trust [MS-KILE] between the root domains of two forests , both of which are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater. Only evaluated on Windows Server 2003 operating system, Windows Server 2008 operating system, Windows Server 2008 R2 operating system, Windows Server 2012 operating system, Windows Server 2012 R2 operating system, and Windows Server 2016 Technical Preview operating system. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.
0x10	TRUST_ATTRIBUTE_CROSS_ORGANIZATION	If this bit is set, then the trust is to a domain or forest that is not part of the organization . The behavior controlled by this bit is explained in [MS-KILE] section 3.3.5.7.5 and [MS-APDS] section 3.1.5 . Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.
0x20	TRUST_ATTRIBUTE_WITHIN_FOREST	If this bit is set, then the trusted domain is within the same forest. Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview.
0x40	TRUST_ATTRIBUTE_TREAT_AS_EXTERNAL	If this bit is set, then a cross-forest trust to a domain is to be treated as an external trust for the purposes of SID Filtering. Cross-forest trusts are more stringently filtered than external trusts. This attribute relaxes those cross-forest trusts to be equivalent to external trusts. For more information on how each trust type is filtered, see [MS-PAC] section 4.1.2.2 . Only evaluated on Windows Server 2003, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview. Only evaluated if SID Filtering is used. Only evaluated on cross-forest trusts having TRUST_ATTRIBUTE_FOREST_TRANSITIVE. Can only be set if forest and trusted forest are running in a forest functional level of DS_BEHAVIOR_WIN2003 or greater.
0x80	TRUST_ATTRIBUTEUSES_RC4_ENCRYPTION	This bit is set on trusts with the trustType set to TRUST_TYPE_MIT, which are capable of using RC4 keys. Historically, MIT Kerberos distributions supported only DES and 3DES keys ([RFC4120] , [RFC3961]). MIT 1.4.1 adopted the RC4HMAC encryption type common to Windows 2000 [MS-KILE] , so trusted domains deploying later versions of the MIT distribution required this bit. For more information, see "Keys and Trusts", section 6.1.6.9.1 .

		Only evaluated on TRUST_TYPE_MIT
0x200	TRUST_ATTRIBUTE_CROSS_ORGANIZATION_NO_TGT_DELEGATION	If this bit is set, tickets granted under this trust MUST NOT be trusted for delegation. The behavior controlled by this bit is as specified in [MS-KILE] section 3.3.5.7.5. Only supported on Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 Technical Preview.
0x400	TRUST_ATTRIBUTE_PIM_TRUST	If this bit and the TATE bit are set, then a cross-forest trust to a domain is to be treated as Privileged Identity Management trust for the purposes of SID Filtering. For more information on how each trust type is filtered, see [MS-PAC] section 4.1.2.2. Evaluated only on Windows Server 2016 Technical Preview Evaluated only if SID Filtering is used. Evaluated only on cross-forest trusts having TRUST_ATTRIBUTE_FOREST_TRANSITIVE. Can be set only if the forest and the trusted forest are running in a forest functional level of DS_BEHAVIOR_WINTHRESHOLD or greater.

- **SID Filtering** [Type = UnicodeString]: [SID Filtering](#) state for the new trust:

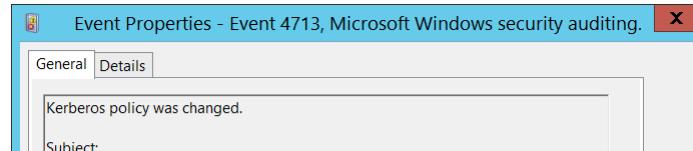
- Enabled
- Disabled

If this attribute was not changed, then it will have “-” value or its old value.

Security Monitoring Recommendations:

For 4716(S): Trusted domain information was modified.

- Any changes in Active Directory domain trust settings must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.



4713([S](#)): Kerberos policy was changed.

Event Description:

This event generates when [Kerberos policy](#) was changed.

This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Logon ID:	0x3E7
Changes Made: (‘-’ means no changes, otherwise each change is shown as: (Parameter Name): (new value) (old value))	
KerMaxT: 0x10c388d000 (0x861c46800); KerMaxR: 0x19254d38000 (0xc92a69c000);	
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4713
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4713</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

```
<TimeCreated SystemTime="2015-10-01T23:15:50.811774300Z" />
<EventRecordID>1049772</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4116" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="KerberosPolicyChange">KerMaxT: 0x10c388d000 (0x861c46800); KerMaxR: 0x19254d38000 (0xc92a69c000);</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to Kerberos policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to Kerberos policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Changes Made [Type = UnicodeString]: '--' means no changes, otherwise each change is shown as: **Parameter_Name**: new_value (**old_value**). Here is a list of possible parameter names:

Parameter Name	Description
----------------	-------------

KerProxy	Maximum tolerance for computer clock synchronization. To convert the KerProxy to minutes you need to: 1. Convert the value to decimal value. 2. Divide value by 6000000000.
KerMaxR	Maximum lifetime for user ticket renewal. To convert the KerProxy to days you need to: 1. Convert the value to decimal value. 2. Divide value by 8640000000000.
KerMaxT	Maximum lifetime for user ticket. To convert the KerMaxT to hours you need to: 1. Convert the value to decimal value. 2. Divide value by 36000000000.
KerMinT	Maximum lifetime for service ticket. To convert the KerMinT to minutes you need to: 1. Convert the value to decimal value. 2. Divide value by 600000000.
KerOpts	Enforce user logon restrictions: <ul style="list-style-type: none">• 0x80 – Enabled• 0x0 – Disabled

Event Properties - Event 4717, Microsoft Windows security audit...

General **Details**

System security access was granted to an account.

Subject:
Security ID: SYSTEM
Account Name: DC01\$
Account Domain: CONTOSO
Logon ID: 0x3E7

Account Modified:
Account Name: CONTOSO\Auditor

Access Granted:
Access Right: SeInteractiveLogonRight

Log Name: Security
Source: Microsoft Windows sec... Logged: 10/1/2015 5:02:33 PM
Event ID: 4717 Task Category: Authentication Policy
Level: Information Keywords: Audit Success
User: N/A Computer: DC01.contoso.local
OpCode: Info
More Information: [Event Log Online](#)

Copy **Close**

This event shows changes in “Kerberos policy”. Here is location of Kerberos policies in Group Policy management console:

Computer Configuration

- ↳ Policies
 - ↳ Software Settings
 - ↳ Windows Settings
 - ↳ Name Resolution Policy
 - ↳ Scripts (Startup/Shutdown)
 - ↳ Security Settings
 - ↳ Account Policies
 - ↳ Password Policy
 - ↳ Account Lockout Policy
 - ↳ Kerberos Policy

Policy	Policy Setting
Enforce user logon restrictions	Not Defined
Maximum lifetime for service ticket	600 minutes
Maximum lifetime for user ticket	10 hours
Maximum lifetime for user ticket renewal	34 days
Maximum tolerance for computer clock synchronization	Not Defined

Security Monitoring Recommendations:

For 4713(S): Kerberos policy was changed.

- Any changes in Kerberos policy reported by current event must be monitored and an alert should be triggered. If this change was not planned, investigate the reason for the change.

4717(S): System security access was granted to an account.

Event Description:

This event generates every time local [logon user right policy](#) is changed and logon right was granted to an account. You will see unique event for every user if logon user rights were granted to multiple accounts.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4717</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T00:02:33.213572000Z" />
<EventRecordID>1049777</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2064" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
<Data Name="AccessGranted">SeInteractiveLogonRight</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to local logon right user policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to local logon right user policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Account Modified:

- **Account Name** [Type = SID]: the SID of the security principal for which logon right was granted. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Access Granted:

- **Access Right** [Type = UnicodeString]: the name of granted logon right. This event generates only for [logon rights](#), which are as follows:

Value	Group Policy Name
SeNetworkLogonRight	Access this computer from the network
SeRemoteInteractiveLogonRight	Allow logon through Terminal Services
SeDenyNetworkLogonRight	Deny access to this computer from the network
SeDenyBatchLogonRight	Deny logon as a batch job
SeDenyServiceLogonRight	Deny logon as a service
SeDenyInteractiveLogonRight	Deny logon locally
SeDenyRemoteInteractiveLogonRight	Deny logon through Terminal Services
SeBatchLogonRight	Log on as a batch job
SeServiceLogonRight	Log on as a service
SeInteractiveLogonRight	Log on locally

Security Monitoring Recommendations:

For 4717(S): System security access was granted to an account.

Type of monitoring required	Recommendation
Actions typically performed by the SYSTEM account: This event and certain other events should be monitored to see if they are triggered by any account other than SYSTEM.	Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever " Subject\Security ID " is not SYSTEM.

High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.

Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.

Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.

Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.

Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.

Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.

External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).

Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should perform only limited actions, or no actions at all.

Logon rights that should be restricted: You might have a list of user logon rights that you

Monitor this event with the “**Subject\Security ID**” and “**Account Modified\Account Name**” that correspond to the high-value account or accounts.

When you monitor for anomalies or malicious actions, use the “**Subject\Security ID**” (with other information) to monitor how or when a particular account is being used.

Monitor this event with the “**Subject\Security ID**” that corresponds to the accounts that should never be used.

If this event corresponds to a “whitelist-only” action, review the “**Subject\Security ID**” for accounts that are outside the whitelist.

If you have specific user logon rights policies, for example, a whitelist of accounts that can log on to certain computers, monitor this event to confirm that any “**Access Right**” was granted only to the appropriate “**Account Modified\Account Name**.”

If this event corresponds to an action you want to monitor for certain account types, review the “**Subject\Security ID**” and “**Account Modified\Account Name**” to see whether the account type is as expected.

For example, if non-service accounts should never be granted certain logon rights (for example, **SeServiceLogonRight**), monitor this event for those accounts and rights.

Monitor this event for the “**Subject\Account Domain**” corresponding to accounts from another domain or “external” accounts.

Monitor the target **Computer**: (or other target device) for actions performed by the “**Subject\Security ID**” that you are concerned about. Also be sure to check “**Account Modified\Account Name**” to see whether logon rights should be granted to that account.

For high-value servers or other computers, we recommend that you track this event and investigate whether the specific “**Access Right**” should be granted to “**Account Modified\Account Name**” in each case.

Monitor this event and compare the “**Access Right**” to your list of restricted rights.

want to monitor (for example, **SeServiceLogonRight**).

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Account Name**” for names that don’t comply with naming conventions.

4718(S): System security access was removed from an account.

Event Properties - Event 4718, Microsoft Windows security audit...

General Details

Security ID:	SYSTEM																				
Account Name:	DC01\$																				
Account Domain:	CONTOSO																				
Logon ID:	0x3E7																				
Account Modified:																					
Account Name:	CONTOSO\Auditor																				
Access Removed:																					
Access Right:	SeInteractiveLogonRight																				
Log Name:	Security																				
Source:	Microsoft Windows sec	Event ID:	4718	Logged:	10/1/2015 4:35:46 PM	Level:	Information	Task Category:	Authentication Policy	User:	N/A	Keywords:	Audit Success	OpCode:	Info	Computer:	DC01.contoso.local	More Information: Event Log Online			
Event ID:	4718	Logged:	10/1/2015 4:35:46 PM																		
Level:	Information	Task Category:	Authentication Policy																		
User:	N/A	Keywords:	Audit Success																		
OpCode:	Info	Computer:	DC01.contoso.local																		
More Information: Event Log Online																					

Copy **Close**

Event Description:

This event generates every time local [logon user right policy](#) is changed and logon right was removed from an account. You will see unique event for every user if logon user rights were removed for multiple accounts.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4718</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-01T23:35:46.375134200Z" />
<EventRecordID>1049773</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="5028" />
```

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-2104</Data>
```

```
<Data Name="AccessRemoved">SeInteractiveLogonRight</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to local logon right user policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to local logon right user policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Account Modified:

- **Account Name** [Type = SID]: the SID of the security principal for which logon right was removed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Access Removed:

- **Access Right** [Type = UnicodeString]: the name of removed logon right. This event generates only for [logon rights](#), which are as follows:

Value	Group Policy Name
SeNetworkLogonRight	Access this computer from the network
SeRemoteInteractiveLogonRight	Allow logon through Terminal Services
SeDenyNetworkLogonRight	Deny access to this computer from the network
SeDenyBatchLogonRight	Deny logon as a batch job
SeDenyServiceLogonRight	Deny logon as a service
SeDenyInteractiveLogonRight	Deny logon locally
SeDenyRemoteInteractiveLogonRight	Deny logon through Terminal Services
SeBatchLogonRight	Log on as a batch job

SeServiceLogonRight	Log on as a service
SeInteractiveLogonRight	Log on locally

Security Monitoring Recommendations:

For 4718(S): System security access was removed from an account.

Type of monitoring required	Recommendation
Actions typically performed by the SYSTEM account: This event and certain other events should be monitored to see if they are triggered by any account other than SYSTEM.	Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “ Subject\Security ID ” is not SYSTEM.
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Subject\Security ID ” and “ Account Modified\Account Name ” that correspond to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Security ID ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist. If you have specific user logon rights policies, for example, a whitelist of accounts that can log on to certain computers, monitor this event to confirm that it was appropriate that the “ Access Right ” was removed from “ Account Modified\Account Name .”
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” and “ Account Modified\Account Name ” to see whether the account type is as expected. For example, if critical remote network service accounts have user logon rights which should never be removed (for example, SeNetworkLogonRight), monitor this event for the “ Account Modified\Account Name ” and the appropriate rights. As another example, if non-service accounts should never be granted certain logon rights (for example, SeServiceLogonRight), you might monitor this event, because a right can be

	removed only after it was previously granted.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should perform only limited actions, or no actions at all.	Monitor the target Computer : (or other target device) for actions performed by the “ Subject\Security ID ” that you are concerned about. Also be sure to check “ Account Modified\Account Name ” to see whether logon rights should be removed from that account. For high-value servers or other computers, we recommend that you track this event and investigate whether the specific “ Access Right ” should be removed from “ Account Modified\Account Name ” in each case.
Logon rights that should be restricted: You might have a list of user logon rights that you want to monitor (for example, SeServiceLogonRight). “Deny” rights that should not be removed: Your organization might use “Deny” rights that should not be removed, for example, SeDenyRemoteInteractiveLogonRight .	Monitor this event and compare the “ Access Right ” to your list of restricted rights. Monitor this event to discover the removal of a right that should never have been granted, so that you can investigate further. You can also monitor this event to discover the removal of “Deny” rights. When these rights are removed, it could be an approved action, done by mistake, or part of malicious activity. These rights include: <ul style="list-style-type: none">• SeDenyNetworkLogonRight:• SeDenyBatchLogonRight• SeDenyServiceLogonRight• SeDenyInteractiveLogonRight• SeDenyRemoteInteractiveLogonRight
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

4739(\$): Domain Policy was changed.

 Event Properties - Event 4739, Microsoft Windows security audit... X

General Details

Domain Policy was changed.

Change Type: Password Policy modified

Subject:

Logon ID: 0x3e7

Domain:
Domain Name: CONTOSO
Domain ID: CONTOSO\

Changed Attributes:
Min. Password Age: -
Max. Password Age: -
Force Logoff: -
Lockout Threshold: -
Lockout Observation Window: -
Lockout Duration: -
Password Properties: -
Min. Password Length: -
Password History Length: 13
Machine Account Quota: -
Mixed Domain Mode: -
Domain Behavior Version: -
OEM Information: -

Additional Information:
Privileges: -

Log Name: Security
Source: Microsoft Windows sev
Event ID: 4739
Level: Information
User: N/A
OpCode: Info
Logged: 10/1/2015 5:45:37 PM
Task Category: Authentication Policy
Keywords: Audit Success
Computer: DC01.contoso.local

More Information: [Event Log Online](#)

Copy Close

Event Description:

This event generates when one of the following changes was made to local computer security policy:

- Computer's "\Security Settings\Account Policies\Account Lockout Policy" settings were modified.
- Computer's "\Security Settings\Account Policies\Password Policy" settings were modified.
- "Network security: Force logoff when logon hours expire" group policy setting was changed.
- Domain functional level was changed or some other attributes changed (see details in event description).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4739</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T00:45:37.587380900Z" />
<EventRecordID>1049781</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="1648" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="DomainPolicyChanged">Password Policy</Data>
<Data Name="DomainName">CONTOSO</Data>
<Data Name="DomainSid">S-1-5-21-3457937927-2839227994-823803824</Data>
```

```
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="PrivilegeList">-</Data>
<Data Name="MinPasswordAge">-</Data>
```

```

<Data Name="MaxPasswordAge">-</Data>
<Data Name="ForceLogoff">-</Data>
<Data Name="LockoutThreshold">-</Data>
<Data Name="LockoutObservationWindow">-</Data>
<Data Name="LockoutDuration">-</Data>
<Data Name="PasswordProperties">-</Data>
<Data Name="MinPasswordLength">-</Data>
<Data Name="PasswordHistoryLength">13</Data>
<Data Name="MachineAccountQuota">-</Data>
<Data Name="MixedDomainMode">-</Data>
<Data Name="DomainBehaviorVersion">-</Data>
<Data Name="OemInformation">-</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Change Type [Type = UnicodeString]: the type of change which was made. The format is “**policy_name** modified”. These are some possible values of **policy_name**:

Value	Group Policy Name \ Description
Lockout Policy	Computer's “\Security Settings\Account Policies\Account Lockout Policy” settings were modified.
Password Policy	Computer's “\Security Settings\Account Policies\Password Policy” settings were modified.
Logoff Policy	“Network security: Force logoff when logon hours expire” group policy setting was changed.
-	Machine Account Quota (ms-DS-MachineAccountQuota) domain attribute was modified.

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to specific local policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to specific local policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.

- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Domain:

- **Domain Name** [Type = UnicodeString]: the name of domain for which policy changes were made.
- **Domain ID** [Type = SID]: the SID of domain for which policy changes were made. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Changed Attributes: For attributes which were not changed the value will be "-".

- **Min. Password Age** [Type = UnicodeString]: "\Security Settings\Account Policies\Password Policy\Minimum password age" group policy. Numeric value.
- **Max. Password Age** [Type = UnicodeString]: "\Security Settings\Account Policies\Password Policy\Maximum password age" group policy. Numeric value.
- **Force Logoff** [Type = UnicodeString]: "\Security Settings\Local Policies\Security Options\Network security: Force logoff when logon hours expire" group policy.
- **Lockout Threshold** [Type = UnicodeString]: "\Security Settings\Account Policies\Account Lockout Policy\Account lockout threshold" group policy. Numeric value.
- **Lockout Observation Window** [Type = UnicodeString]: "\Security Settings\Account Policies\Account Lockout Policy\Reset account lockout counter after" group policy. Numeric value.
- **Lockout Duration** [Type = UnicodeString]: "\Security Settings\Account Policies\Account Lockout Policy\Account lockout duration" group policy. Numeric value.
- **Password Properties** [Type = UnicodeString]:

Value	Group Policy settings
0	\Security Settings\Account Policies\Password Policy\Store passwords using reversible encryption - Disabled. \Security Settings\Account Policies\Password Policy\Password must meet complexity requirements – Disabled.
1	\Security Settings\Account Policies\Password Policy\Store passwords using reversible encryption - Disabled. \Security Settings\Account Policies\Password Policy\Password must meet complexity requirements – Enabled.
16	\Security Settings\Account Policies\Password Policy\Store passwords using reversible encryption - Enabled. \Security Settings\Account Policies\Password Policy\Password must meet complexity requirements – Disabled.
17	\Security Settings\Account Policies\Password Policy\Store passwords using reversible encryption - Enabled. \Security Settings\Account Policies\Password Policy\Password must meet complexity requirements – Enabled.

- **Min. Password Length** [Type = UnicodeString]: "\Security Settings\Account Policies\Password Policy\Minimum password length" group policy. Numeric value.
- **Password History Length** [Type = UnicodeString]: "\Security Settings\Account Policies\Password Policy\Enforce password history" group policy. Numeric value.
- **Machine Account Quota** [Type = UnicodeString]: [ms-DS-MachineAccountQuota](#) domain attribute was modified. Numeric value.
- **Mixed Domain Mode** [Type = UnicodeString]: there is no information about this field in this document.
- **Domain Behavior Version** [Type = UnicodeString]: [msDS-Behavior-Version](#) domain attribute was modified. Numeric value. Possible values:

Value	Identifier	Domain controller operating systems that are allowed in the domain
0	DS_BEHAVIOR_WIN2000	Windows 2000 Server operating system Windows Server 2003 operating system Windows Server 2008 operating system Windows Server 2008 R2 operating system Windows Server 2012 operating system Windows Server 2012 R2 operating system

		Windows Server 2016 Technical Preview operating system
1	DS_BEHAVIOR_WIN2003_WITH_MIXED_DOMAINS	Windows Server 2003 Windows Server 2008 Windows Server 2008 R2 Windows Server 2012 Windows Server 2012 R2 Windows Server 2016 Technical Preview
2	DS_BEHAVIOR_WIN2003	Windows Server 2003 Windows Server 2008 Windows Server 2008 R2 Windows Server 2012 Windows Server 2012 R2 Windows Server 2016 Technical Preview
3	DS_BEHAVIOR_WIN2008	Windows Server 2008 Windows Server 2008 R2 Windows Server 2012 Windows Server 2012 R2 Windows Server 2016 Technical Preview
4	DS_BEHAVIOR_WIN2008R2	Windows Server 2008 R2 Windows Server 2012 Windows Server 2012 R2 Windows Server 2016 Technical Preview
5	DS_BEHAVIOR_WIN2012	Windows Server 2012 Windows Server 2012 R2 Windows Server 2016 Technical Preview
6	DS_BEHAVIOR_WIN2012R2	Windows Server 2012 R2 Windows Server 2016 Technical Preview
7	DS_BEHAVIOR_WINTHRESHOLD	Windows Server 2016 Technical Preview

- **OEM Information** [Type = UnicodeString]: there is no information about this field in this document.

Additional Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were used during the operation, for example, SeBackupPrivilege. This parameter might not be captured in the event, and in that case appears as "-". See full list of user privileges in the table below:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.

SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	<p>Required to perform backup operations.</p> <p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses <code>NtCreateToken()</code> or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a</p>

		client, as long as the account of the client does not have the Account cannot be delegated account control flag set.
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
SeMachineAccountPrivilege	Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE

		<ul style="list-style-type: none"> • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	<p>This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers.</p> <p>With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.</p>
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	<p>Required to gather profiling information for the entire system.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.</p>
SeSystemtimePrivilege	Change the system time	<p>Required to modify the system time.</p> <p>With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.</p>
SeTakeOwnershipPrivilege	Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>
SeTcbPrivilege	Act as part of the operating system	<p>This privilege identifies its holder as part of the trusted computer base.</p> <p>This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.</p>
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivilege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from	Required to undock a laptop.

	docking station	With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

Security Monitoring Recommendations:

For 4739(S): Domain Policy was changed.

- Any settings changes to “**Account Lockout Policy**”, “**Password Policy**”, or “**Network security: Force logoff when logon hours expire**”, plus any **domain functional level and attributes** changes that are reported by this event, must be monitored and an alert should be triggered. If this change was not planned, investigate the reason for the change.

4864(S): A namespace collision was detected.

This event is generated when a namespace collision was detected.

There is no example of this event in this document.

Event Schema:

A namespace collision was detected.

Target Type:%1

 Event Properties - Event 4865, Microsoft Windows security audit... X

General Details

A trusted forest information entry was added.

Subject:

Security ID:	CONTOSO\dadmin
Account Name:	dadmin
Account Domain:	CONTOSO
Logon ID:	0x138E0

Trust Information:

Forest Root:	Fabrikam.local
Forest Root SID:	FABRIKAM\
Operation ID:	0x648620
Entry Type:	2
Flags:	0
Top Level Name:	-
DNS Name:	Fabrikam.local
NetBIOS Name:	FABRIKAM
Domain SID:	FABRIKAM\

Log Name: Security
Source: Microsoft Windows sec... **Logged:** 10/1/2015 8:11:33 PM
Event ID: 4865 **Task Category:** Authentication Policy
Level: Information **Keywords:** Audit Success
User: N/A **Computer:** DC01 contoso.local

Target Name:%2

Forest Root:%3

Top Level Name:%4

DNS Name:%5

NetBIOS Name:%6

Security ID:%7

New Flags:%8

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4865(S): A trusted forest information entry was added.

Event Description:

This event generates when new trusted forest information entry was added.

This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4865</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T03:11:33.397715700Z" />
<EventRecordID>1049810</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4808" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ForestRoot">Fabrikam.local</Data>
<Data Name="ForestRootSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
<Data Name="OperationId">0x648620</Data>
<Data Name="EntryType">2</Data>
<Data Name="Flags">0</Data>
<Data Name="TopLevelName">-</Data>
<Data Name="DnsName">Fabrikam.local</Data>
<Data Name="NetbiosName">FABRIKAM</Data>
<Data Name="DomainSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x138eb0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “add a trusted forest information entry” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “add a trusted forest information entry” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Trust Information:

- **Forest Root** [Type = UnicodeString]: the name of the Active Directory forest for which trusted forest information entry was added.
- **Forest Root SID** [Type = SID]: the SID of the Active Directory forest for which trusted forest information entry was added. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Operation ID** [Type = HexInt64]: unique hexadecimal identifier of the operation. You can correlate this event with other events ([4866](#)(S): A trusted forest information entry was removed, [4867](#)(S): A trusted forest information entry was modified.) using this field.
- **Entry Type** [Type = UInt32]: the type of added entry:

Value	Type Name	Description
0	ForestTrustTopLevelName	The DNS name of the trusted forest . The structure used for this record type is equivalent to LSA_UNICODE_STRING
1	ForestTrustTopLevelNameEx	This type commonly used for name suffix exceptions. The structure used for this record type is equivalent to LSA_UNICODE_STRING .
2	ForestTrustDomainInfo	This field specifies a record containing identification and name information

- **Flags** [Type = UInt32]: The following table specifies the possible flags.

Some flag values are reused for different forest record types. See the “Meaning” column for more information.

Value	Trust Type	Meaning
0	-	No flags were set.
1	ForestTrustTopLevelNameEx	The top-level name trust record is disabled during initial creation.
	ForestTrustTopLevelName	
2	ForestTrustDomainInfo	The domain information trust record is disabled by the domain administrator.
	ForestTrustTopLevelNameEx	The top-level name trust record is disabled by the domain administrator.

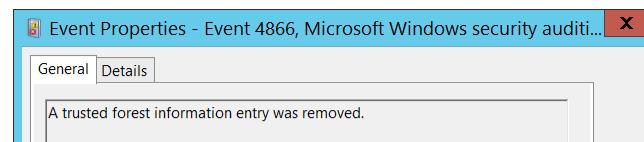
	ForestTrustTopLevelName	
4	ForestTrustDomainInfo	The domain information trust record is disabled due to a conflict.
	ForestTrustTopLevelNameEx	The top-level name trust record is disabled due to a conflict.
	ForestTrustTopLevelName	
8	ForestTrustDomainInfo	The domain information trust record is disabled by the domain administrator.
	ForestTrustDomainInfo	The domain information trust record is disabled due to a conflict.

- **Top Level Name** [Type = UnicodeString]: the name of the new trusted forest information entry.
- **DNS Name** [Type = UnicodeString]: DNS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **NetBIOS Name** [Type = UnicodeString]: NetBIOS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **Domain SID** [Type = SID]: SID of the trust partner. This parameter might not be captured in the event, and in that case appears as “NULL SID”.

Security Monitoring Recommendations:

For 4865(S): A trusted forest information entry was added.

- Any changes related to Active Directory forest trusts (especially creation of the new trust) must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.



4866(S): A trusted forest information entry was removed.

Event Description:

This event generates when the trusted forest information entry was removed.

This event is generated only on domain controllers.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Account Domain:	CONTOSO
Logon ID:	0x138E80
Trust Information:	
Forest Root:	Fabrikam.local
Forest Root SID:	FABRIKAM\
Operation ID:	0x705F4C
Entry Type:	1
Flags:	0
Top Level Name:	my.fabrikam.local
DNS Name:	-
NetBIOS Name:	-
Domain SID:	NULL SID
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4866
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4865</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T03:11:33.397715700Z" />
<EventRecordID>1049810</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4808" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="ForestRoot">Fabrikam.local</Data>
  <Data Name="ForestRootSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
  <Data Name="OperationId">0x648620</Data>
  <Data Name="EntryType">2</Data>
  <Data Name="Flags">0</Data>
  <Data Name="TopLevelName">-</Data>
  <Data Name="DnsName">Fabrikam.local</Data>
  <Data Name="NetbiosName">FABRIKAM</Data>
  <Data Name="DomainSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x138eb0</Data>
</EventData>
</Event>
```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “remove a trusted forest information entry” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “remove a trusted forest information entry” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.

- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Trust Information:

- **Forest Root** [Type = UnicodeString]: the name of the Active Directory forest for which trusted forest information entry was removed.
- **Forest Root SID** [Type = SID]: the SID of the Active Directory forest for which trusted forest information entry was removed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Operation ID** [Type = HexInt64]: unique hexadecimal identifier of the operation. You can correlate this event with other events ([4865](#)(S): A trusted forest information entry was added, [4867](#)(S): A trusted forest information entry was modified.) using this field.
- **Entry Type** [Type = UInt32]: the type of removed entry:

Value	Type Name	Description
0	ForestTrustTopLevelName	The DNS name of the trusted forest . The structure used for this record type is equivalent to LSA_UNICODE_STRING
1	ForestTrustTopLevelNameEx	This type commonly used for name suffix exceptions. The structure used for this record type is equivalent to LSA_UNICODE_STRING .
2	ForestTrustDomainInfo	This field specifies a record containing identification and name information

- **Flags** [Type = UInt32]: The following table specifies the possible flags.

Some flag values are reused for different forest record types. See the “Meaning” column for more information.

Value	Trust Type	Meaning
0	-	No flags were set.
1	ForestTrustTopLevelNameEx ForestTrustTopLevelName	The top-level name trust record is disabled during initial creation.
2	ForestTrustDomainInfo ForestTrustTopLevelNameEx ForestTrustTopLevelName	The domain information trust record is disabled by the domain administrator.
4	ForestTrustDomainInfo ForestTrustTopLevelNameEx ForestTrustTopLevelName	The top-level name trust record is disabled by the domain administrator.
8	ForestTrustDomainInfo ForestTrustDomainInfo	The domain information trust record is disabled due to a conflict.

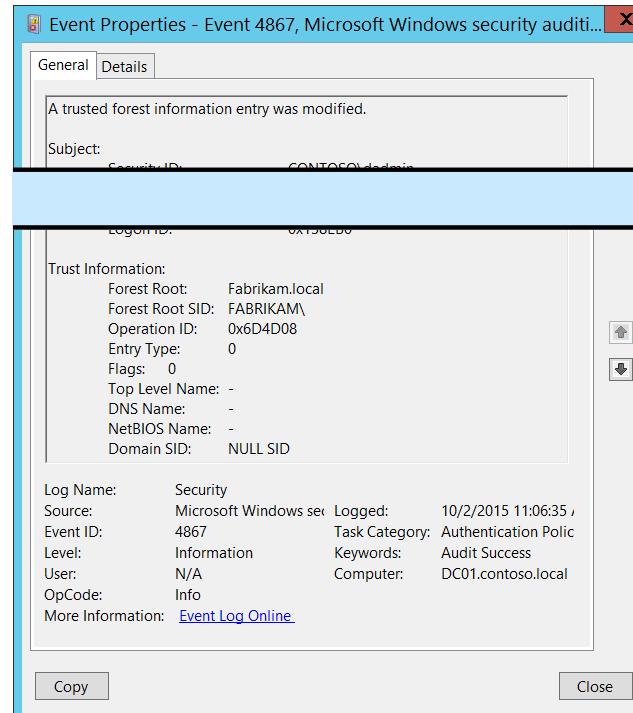
- **Top Level Name** [Type = UnicodeString]: the name of the removed trusted forest information entry.
- **DNS Name** [Type = UnicodeString]: DNS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **NetBIOS Name** [Type = UnicodeString]: NetBIOS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **Domain SID** [Type = SID]: SID of the trust partner. This parameter might not be captured in the event, and in that case appears as “NULL SID”.

Security Monitoring Recommendations:

For 4866(S): A trusted forest information entry was removed.

- Any changes related to Active Directory forest trusts (especially trust removal) must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.

4867(S): A trusted forest information entry was modified.

 Event Properties - Event 4867, Microsoft Windows security audit...

General Details

A trusted forest information entry was modified.

Subject: Security ID: CONTOSO\dc01\$

Logon ID: 0x150E00

Trust Information:

- Forest Root: Fabrikam.local
- Forest Root SID: FABRIKAM\
- Operation ID: 0x6D4D08
- Entry Type: 0
- Flags: 0
- Top Level Name: -
- DNS Name: -
- NetBIOS Name: -
- Domain SID: NULL SID

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 4867
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Logged: 10/2/2015 11:06:35 / Task Category: Authentication Policy
 Keywords: Audit Success
 Computer: DC01.contoso.local

Copy **Close**

Event Description:

This event generates the trusted forest information entry was modified.

This event is generated only on domain controllers.

This event contains new values only, it doesn't contain old values and it doesn't show you which trust attributes were modified.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4867</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13569</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T03:11:33.397715700Z" />
<EventRecordID>1049810</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4808" />
<Channel>Security</Channel>
```

```
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ForestRoot">Fabrikam.local</Data>
<Data Name="ForestRootSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
<Data Name="OperationId">0x648620</Data>
<Data Name="EntryType">2</Data>
<Data Name="Flags">0</Data>
<Data Name="TopLevelName">-</Data>
<Data Name="DnsName">Fabrikam.local</Data>
```

```

<Data Name="NetbiosName">FABRIKAM</Data>
<Data Name="DomainSid">S-1-5-21-2703072690-1374247579-2643703677</Data>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x138eb0</Data>
</EventData>
</Event>

```

Required Server Roles: Active Directory domain controller.

Minimum OS Version: Windows Server 2008.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested the “modify/change a trusted forest information entry” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “modify/change a trusted forest information entry” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Trust Information:

- **Forest Root** [Type = UnicodeString]: the name of the Active Directory forest for which trusted forest information entry was modified.
- **Forest Root SID** [Type = SID]: the SID of the Active Directory forest for which trusted forest information entry was modified. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.
- **Operation ID** [Type = HexInt64]: unique hexadecimal identifier of the operation. You can correlate this event with other events ([4865](#)(S): A trusted forest information entry was added, [4866](#)(S): A trusted forest information entry was removed) using this field.
- **Entry Type** [Type = UInt32]: the type of modified entry:

Value	Type Name	Description
-------	-----------	-------------

0	ForestTrustTopLevelName	The DNS name of the trusted forest . The structure used for this record type is equivalent to LSA_UNICODE_STRING
1	ForestTrustTopLevelNameEx	This type commonly used for name suffix exceptions. The structure used for this record type is equivalent to LSA_UNICODE_STRING .
2	ForestTrustDomainInfo	This field specifies a record containing identification and name information

- **Flags** [Type = UInt32]: The following table specifies the possible flags.

Some flag values are reused for different forest record types. See the “Meaning” column for more information.

Value	Trust Type	Meaning
0	-	No flags were set.
1	ForestTrustTopLevelNameEx ForestTrustTopLevelName	The top-level name trust record is disabled during initial creation.
	ForestTrustDomainInfo	The domain information trust record is disabled by the domain administrator.
2	ForestTrustTopLevelNameEx ForestTrustTopLevelName	The top-level name trust record is disabled by the domain administrator.
	ForestTrustDomainInfo	The domain information trust record is disabled due to a conflict.
4	ForestTrustTopLevelNameEx ForestTrustTopLevelName	The top-level name trust record is disabled due to a conflict.
	ForestTrustDomainInfo	The domain information trust record is disabled by the domain administrator.
8	ForestTrustDomainInfo	The domain information trust record is disabled due to a conflict.

- **Top Level Name** [Type = UnicodeString]: the name of the modified trusted forest information entry.
- **DNS Name** [Type = UnicodeString]: DNS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **NetBIOS Name** [Type = UnicodeString]: NetBIOS name of the trust partner. This parameter might not be captured in the event, and in that case appears as “-”.
- **Domain SID** [Type = SID]: SID of the trust partner. This parameter might not be captured in the event, and in that case appears as “NULL SID”.

Security Monitoring Recommendations:

For 4867(S): A trusted forest information entry was modified.

- Any changes in Active Directory forest trust settings must be monitored and alerts should be triggered. If this change was not planned, investigate the reason for the change.

Audit Authorization Policy Change

Audit Authorization Policy Change allows you to audit assignment and removal of user rights in user right policies, changes in security token object permission, resource attributes changes and Central Access Policy changes for file system objects.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	No	IF	No	<p>IF – With Success auditing for this subcategory, you can get information related to changes in user rights policies, or changes of resource attributes or Central Access Policy applied to file system objects.</p> <p>However, if you are using an application or system service that makes changes to system privileges through the AdjustPrivilegesToken API, we do not recommend Success auditing because of the high volume of event “4703(S): A user right was adjusted” that may be generated. As of Windows 10, event 4703 is generated by applications or services that dynamically adjust token privileges. An example of such an application is System Center Configuration Manager, which makes WMI queries at recurring intervals and quickly generates a large number of 4703 events (with the WMI activity listed as coming from svchost.exe).</p> <p>If one of your applications or services is generating a large number of 4703 events, you might find that your event-management software has filtering logic that can automatically discard the recurring events, which would make it easier to work with Success auditing for this category.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	IF	No	IF	No	<p>IF – With Success auditing for this subcategory, you can get information related to changes in user rights policies, or changes of resource attributes or Central Access Policy applied to file system objects.</p> <p>However, if you are using an application or system service that makes changes to system privileges through the AdjustPrivilegesToken API, we do not recommend Success auditing because of the high volume of event “4703(S): A user right was adjusted” that may be generated. As of Windows 10, event 4703 is generated by applications or services that dynamically adjust token privileges. An example of such an application is System Center Configuration Manager, which makes WMI queries at recurring intervals and quickly generates a large number of 4703 events (with the WMI activity listed as coming from svchost.exe).</p> <p>If one of your applications or services is generating a large number of 4703 events, you might find that your event-management software has filtering logic that can automatically discard the recurring events, which would make it easier to work with Success auditing for this category.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	IF	No	IF	No	IF – With Success auditing for this subcategory, you can get information related to changes in user rights policies, or changes of resource attributes or Central Access Policy applied to file system objects.

				<p>However, if you are using an application or system service that makes changes to system privileges through the AdjustPrivilegesToken API, we do not recommend Success auditing because of the high volume of event “4703(S): A user right was adjusted” that may be generated. As of Windows 10, event 4703 is generated by applications or services that dynamically adjust token privileges. An example of such an application is System Center Configuration Manager, which makes WMI queries at recurring intervals and quickly generates a large number of 4703 events (with the WMI activity listed as coming from svchost.exe).</p> <p>If one of your applications or services is generating a large number of 4703 events, you might find that your event-management software has filtering logic that can automatically discard the recurring events, which would make it easier to work with Success auditing for this category.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
--	--	--	--	--

Events List:

- [4703\(S\): A user right was adjusted.](#)
- [4704\(S\): A user right was assigned.](#)
- [4705\(S\): A user right was removed.](#)
- [4670\(S\): Permissions on an object were changed.](#)
- [4911\(S\): Resource attributes of the object were changed.](#)
- [4913\(S\): Central Access Policy on the object was changed.](#)

Event volume: Medium to High.

4703(\$): A user right was adjusted.

Event Properties - Event 4703, Microsoft Windows security auditing.

General Details

A user right was adjusted.

Subject:

- Security ID: SYSTEM
- Account Name: WIN-GG82ULGC9GO\$
- Account Domain: CONTOSO
- Logon Id: 0x3E7

Target Account:

Process Information:

- Process ID: 0x270
- Process Name: C:\Windows\System32\svchost.exe

Enabled Privileges:

- SeAssignPrimaryTokenPrivilege
- SeIncreaseQuotaPrivilege
- SeSecurityPrivilege
- SeTakeOwnershipPrivilege
- SeLoadDriverPrivilege
- SeSystemtimePrivilege
- SeBackupPrivilege
- SeRestorePrivilege
- SeShutdownPrivilege
- SeSystemEnvironmentPrivilege
- SeUndockPrivilege
- SeManageVolumePrivilege

Disabled Privileges:

Log Name: Security
Source: Microsoft Windows security
Event ID: 4703
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online Help](#)

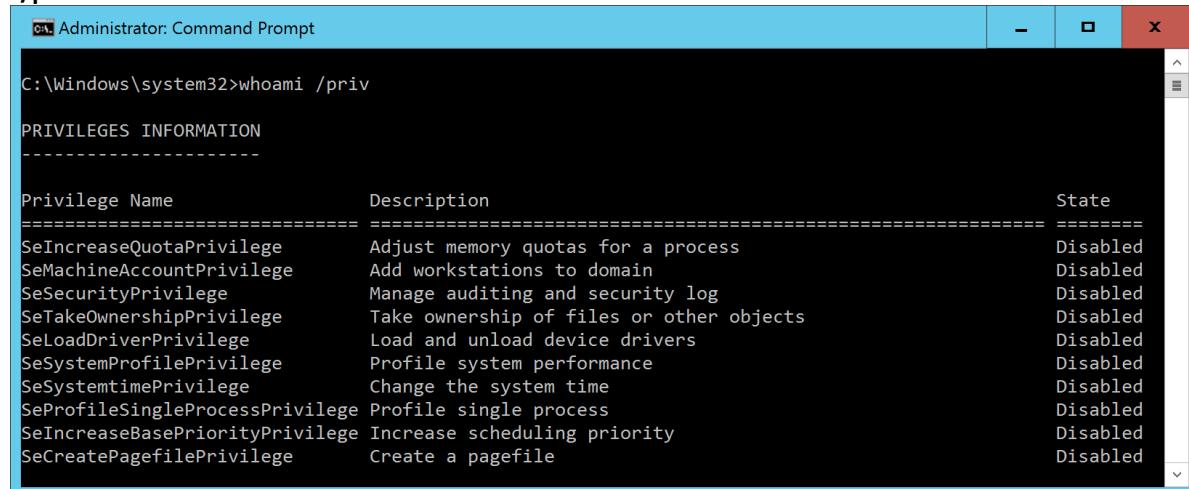
Copy **Close**

Event Description:

This event generates when [token privileges](#) were enabled or disabled for a specific account's token. As of Windows 10, event 4703 is also logged by applications or services that dynamically adjust token privileges. An example of such an application is System Center Configuration Manager, which makes WMI queries at recurring intervals and quickly generates a large number of 4703 events (with the WMI activity listed as coming from svchost.exe). If you are using an application or system service that makes changes to system privileges through the `AdjustPrivilegesToken` API, you might need to disable Success auditing for this subcategory (Audit Authorization Policy Change), or work with a very high volume of event 4703.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Token privileges provide the ability to take certain system-level actions that you only need to do at particular moments. For example, anybody can restart a computer, but the operating system doesn't enable that privilege by default. Instead, the privilege is enabled when you click **Shutdown**. You can check the current state of the user's token privileges using the **whoami /priv** command:



Privilege Name	Description	State
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeMachineAccountPrivilege	Add workstations to domain	Disabled
SeSecurityPrivilege	Manage auditing and security log	Disabled
SeTakeOwnershipPrivilege	Take ownership of files or other objects	Disabled
SeLoadDriverPrivilege	Load and unload device drivers	Disabled
SeSystemProfilePrivilege	Profile system performance	Disabled
SeSystemtimePrivilege	Change the system time	Disabled
SeProfileSingleProcessPrivilege	Profile single process	Disabled
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Disabled
SeCreatePagefilePrivilege	Create a pagefile	Disabled

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4703</EventID>
<Version>0</Version>
<Level>0</Level>
```

```
<Task>13570</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T20:49:46.365958700Z" />
<EventRecordID>5245</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="3632" />
<Channel>Security</Channel>
<Computer>WIN-GG82ULGC9GO.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">WIN-GG82ULGC9GO$</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x3e7</Data>
  <Data Name="TargetUserSid">S-1-5-18</Data>
  <Data Name="TargetUserName">WIN-GG82ULGC9GO$</Data>
  <Data Name="TargetDomainName">CONTOSO</Data>
  <Data Name="TargetLogonId">0x3e7</Data>
  <Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
  <Data Name="ProcessId">0x270</Data>
  <Data Name="EnabledPrivilegeList">SeAssignPrimaryTokenPrivilege SeIncreaseQuotaPrivilege SeSecurityPrivilege SeTakeOwnershipPrivilege SeLoadDriverPrivilege SeSystemtimePrivilege SeBackupPrivilege SeRestorePrivilege SeShutdownPrivilege SeSystemEnvironmentPrivilege SeUndockPrivilege SeManageVolumePrivilege</Data>
  <Data Name="DisabledPrivilegeList"></Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2016, Windows 10.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested the “enable” or “disable” operation for **Target Account** privileges. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it

in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “enable” or “disable” operation for **Target Account** privileges.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Target Account:

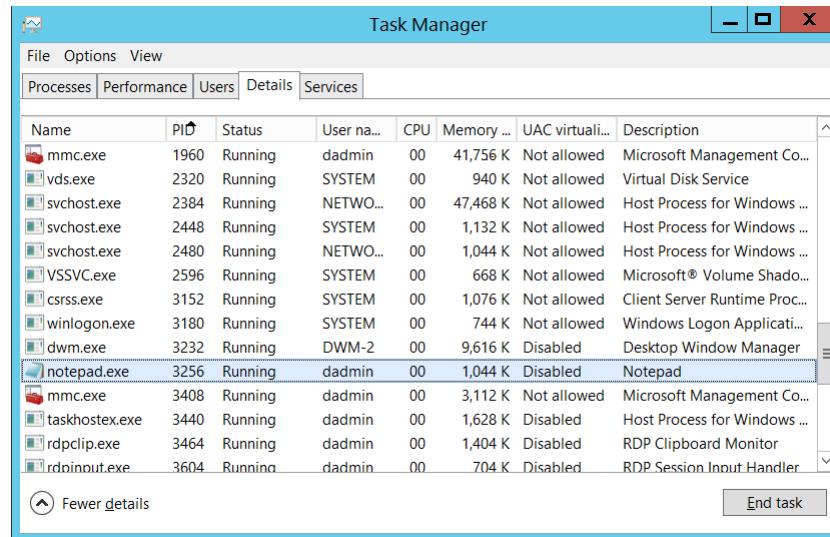
- **Security ID** [Type = SID]: SID of account for which privileges were enabled or disabled. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account for which privileges were enabled or disabled.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that enabled or disabled token privileges. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.
- **Enabled Privileges** [Type = UnicodeString]: the list of enabled user rights. This event generates only for user rights, not logon rights. Here is the list of possible user rights:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	<p>Required to assign the <i>primary token</i> of a process.</p> <p>With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.</p>
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	<p>Required to perform backup operations.</p> <p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.

		With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	Required to create a permanent object. This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs. When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.
SeDebugPrivilege	Debug programs	Required to debug and adjust the memory of a process owned by another account. With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	Required to mark user and computer accounts as trusted for delegation. With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver.

		With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	<p>Required to lock physical pages in memory.</p> <p>With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).</p>
SeMachineAccountPrivilege	Add workstations to domain	<p>With this privilege, the user can create a computer account.</p> <p>This privilege is valid only on domain controllers.</p>
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	<p>Required to gather profiling information for a single process.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.</p>
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	<p>Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and

		LocalSystem accounts on domain controllers. With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	Required to gather profiling information for the entire system. With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.
SeSystemtimePrivilege	Change the system time	Required to modify the system time. With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.
SeTakeOwnershipPrivilege	Take ownership of files or other objects	Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object. With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.
SeTcbPrivilege	Act as part of the operating system	This privilege identifies its holder as part of the trusted computer base. This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivilege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

Disabled Privileges [Type = UnicodeString]: the list of disabled user rights. See possible values in the table above.

Security Monitoring Recommendations:

For 4703(S): A user right was adjusted.

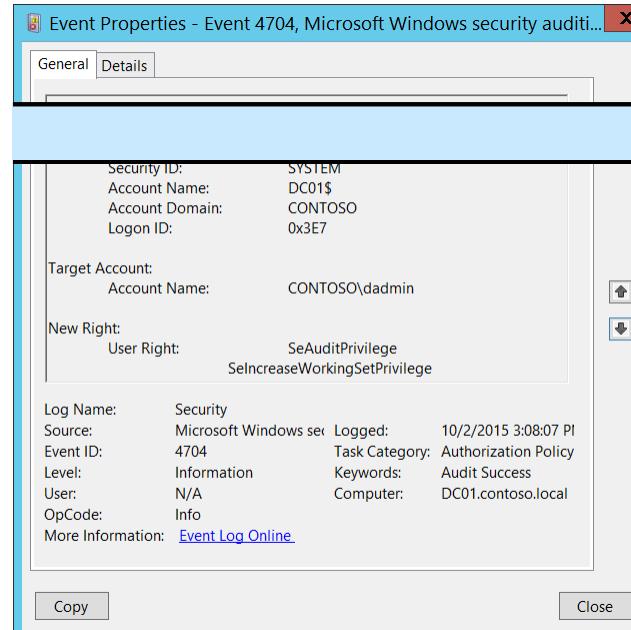
As of Windows 10, event 4703 is generated by applications or services that dynamically adjust token privileges. An example of such an application is System Center Configuration Manager, which makes WMI queries at recurring intervals and quickly generates a large number of 4703 events (with the WMI activity listed as coming from svchost.exe). If you are using an application or system service that makes changes to system privileges through the AdjustPrivilegesToken API, you might need to disable Success auditing for this subcategory (Audit Authorization Policy Change), or work with a very high volume of event 4703.

Otherwise, see the recommendations in the following table.

Type of monitoring required	Recommendation
<p>High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.</p> <p>Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.</p>	Monitor this event with the “ Subject\Security ID ” that corresponds to the high-value account or accounts.
<p>Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.</p>	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
<p>Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.</p>	Monitor this event with the “ Subject\Security ID ” or “ Target Account\Security ID ” that correspond to the accounts that should never be used.
<p>Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.</p>	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist. Also check the “ Target Account\Security ID ” and “ Enabled Privileges ” to see what was enabled.
<p>Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.</p>	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” to see whether the account type is as expected.
<p>External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).</p>	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
<p>Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should perform only limited actions, or no actions at all.</p>	<p>Monitor the target Computer: (or other target device) for actions performed by the “Subject\Security ID” that you are concerned about.</p> <p>Also check “Target Account\Security ID” to see whether the change in privileges should be made on that computer for that account.</p>
<p>User rights that should be restricted or monitored: You might have a list of user rights that you want to restrict or monitor.</p>	<p>Monitor this event and compare the “Enabled Privileges” to your list of user rights. Trigger an alert for user rights that should not be enabled, especially on high-value servers or other computers.</p> <p>For example, you might have SeDebugPrivilege on a list of user rights to be restricted.</p>
<p>Account naming conventions: Your organization might have specific naming conventions</p>	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

for account names.

4704(S): A user right was assigned.

 Event Properties - Event 4704, Microsoft Windows security audit... X

General Details

Security ID:	SYSTEM
Account Name:	DC01\$
Account Domain:	CONTOSO
Logon ID:	0x3E7
Target Account:	
Account Name:	CONTOSO\dadmin
New Right:	
User Right:	SeAuditPrivilege SeIncreaseWorkingSetPrivilege
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4704
Level:	Information
User:	N/A
OpCode:	Info
More Information: Event Log Online	

Copy Close

Event Description:

This event generates every time local user right policy is changed and user right was assigned to an account. You will see unique event for every user.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4704</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13570</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T22:08:07.136050600Z" />
<EventRecordID>1049866</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="1216" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId"›S-1-5-18</Data>
<Data Name="SubjectUserName"›DC01$</Data>
<Data Name="SubjectDomainName"›CONTOSO</Data>
<Data Name="SubjectLogonId"›0x3e7</Data>
<Data Name="TargetSid"›S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="PrivilegeList"›SeAuditPrivilege SeIncreaseWorkingSetPrivilege</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to local user right policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to local user right policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Target Account:

- **Account Name** [Type = SID]: the SID of security principal for which user rights were assigned. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

New Right:

- **User Right** [Type = UnicodeString]: the list of assigned user rights. This event generates only for [user](#) rights, not logon rights. Here is the list of possible user rights:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	Required to perform backup operations. With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system. This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held: <ul style="list-style-type: none">• READ_CONTROL

		<ul style="list-style-type: none"> • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses <code>NtCreateToken()</code> or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.</p>
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	<p>Required to increase the base priority of a process.</p> <p>With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change</p>

		the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
SeMachineAccountPrivilege	Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.
SeSecurityPrivilege	Manage auditing and security log	Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.

		With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys. A user with this privilege can also view and clear the security log.
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers. With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	Required to gather profiling information for the entire system. With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.
SeSystemtimePrivilege	Change the system time	Required to modify the system time. With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.
SeTakeOwnershipPrivilege	Take ownership of files or other objects	Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object. With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.
SeTcbPrivilege	Act as part of the operating system	This privilege identifies its holder as part of the trusted computer base. This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivilege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.
SeUnsolicitedInputPrivilege	Not applicable	Required to read unsolicited input from a <i>terminal</i> device.

Security Monitoring Recommendations:

For 4704(S): A user right was assigned.

Type of monitoring required	Recommendation
<p>Actions typically performed by the SYSTEM account: This event and certain other events should be monitored to see if they are triggered by any account other than SYSTEM.</p>	Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “ Subject\Security ID ” is not SYSTEM.
<p>High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action.</p> <p>Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.</p>	Monitor this event with the “ Subject\Security ID ” that corresponds to the high-value account or accounts.
<p>Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.</p>	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
<p>Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.</p>	Monitor this event with the “ Subject\Security ID ” or “ Target Account\ Account Name ” that correspond to the accounts that should never be used.
<p>Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.</p>	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist. Also check the “ Target Account\Account Name ” and “ New Right ” to see what was enabled.
<p>Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.</p>	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” to see whether the account type is as expected.
<p>External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).</p>	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
<p>Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should perform only limited actions, or no actions at all.</p>	<p>Monitor the target Computer: (or other target device) for actions performed by the “Subject\Security ID” that you are concerned about.</p> <p>Also check “Target Account\ Account Name” to see whether the change in rights should be made on that computer for that account.</p>
<p>User rights that should be restricted or monitored: You might have a list of user rights that you want to restrict or monitor.</p>	Monitor this event and compare the “ New Right\User Right ” to your list of user rights, to see whether the right should be assigned to “ Target Account\Account Name .” Trigger an alert for user rights that should not be enabled, especially on high-value servers or other

computers.

For example, your list of restricted rights might say that only administrative accounts should have **SeAuditPrivilege**. As another example, your list might say that no accounts should have **SeTcbPrivilege** or **SeDebugPrivilege**.

Account naming conventions: Your organization might have specific naming conventions for account names.

Monitor “**Subject\Account Name**” for names that don’t comply with naming conventions.

4705(\$): A user right was removed.

Event Properties - Event 4705, Microsoft Windows security audit...

General Details

Security ID:	SYSTEM
Account Name:	DC01\$
Account Domain:	CONTOSO
Logon ID:	0x3E7
Target Account:	CONTOSO\dadmin
Removed Right:	User Right: SeTimeZonePrivilege
Log Name:	Security
Source:	Microsoft Windows sec
Event ID:	4705
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy Close

Event Description:

This event generates every time local user right policy is changed and user right was removed from an account. You will see unique event for every user.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4705</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13570</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T22:08:07.152488600Z" />
<EventRecordID>1049867</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="1216" />
```

```

<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
```

```
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="TargetSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="PrivilegeList">SeTimeZonePrivilege</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that made a change to local user right policy. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that made a change to local user right policy.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624: An account was successfully logged on.](#)"

Target Account:

- **Account Name** [Type = SID]: the SID of security principal for which user rights were removed. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

Removed Right:

- **User Right** [Type = UnicodeString]: the list of removed user rights. This event generates only for [user rights](#), not logon rights. Here is the list of possible user rights:

Privilege Name	User Right Group Policy Name	Description
SeAssignPrimaryTokenPrivilege	Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
SeAuditPrivilege	Generate security audits	With this privilege, the user can add entries to the security log.
SeBackupPrivilege	Back up files and directories	Required to perform backup operations.

		<p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system.</p> <p>This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
SeChangeNotifyPrivilege	Bypass traverse checking	<p>Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks.</p> <p>With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.</p>
SeCreateGlobalPrivilege	Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
SeCreatePagefilePrivilege	Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
SeCreatePermanentPrivilege	Create permanent shared objects	<p>Required to create a permanent object.</p> <p>This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.</p>
SeCreateSymbolicLinkPrivilege	Create symbolic links	Required to create a symbolic link.
SeCreateTokenPrivilege	Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses <code>NtCreateToken()</code> or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
SeDebugPrivilege	Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.</p>
SeEnableDelegationPrivilege	Enable computer and user accounts to be trusted for delegation	<p>Required to mark user and computer accounts as trusted for delegation.</p> <p>With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be</p>

		delegated account control flag set.
SeImpersonatePrivilege	Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
SeIncreaseBasePriorityPrivilege	Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
SeIncreaseWorkingSetPrivilege	Increase a process working set	Required to allocate more memory for applications that run in the context of users.
SeLoadDriverPrivilege	Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
SeLockMemoryPrivilege	Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
SeMachineAccountPrivilege	Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
SeManageVolumePrivilege	Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
SeProfileSingleProcessPrivilege	Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
SeRelabelPrivilege	Modify an object label	Required to modify the mandatory integrity level of an object.
SeRemoteShutdownPrivilege	Force shutdown from a remote system	Required to shut down a system using a network request.
SeRestorePrivilege	Restore files and directories	Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held: <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY

		<ul style="list-style-type: none"> • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
SeSecurityPrivilege	Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys.</p> <p>A user with this privilege can also view and clear the security log.</p>
SeShutdownPrivilege	Shut down the system	Required to shut down a local system.
SeSyncAgentPrivilege	Synchronize directory service data	<p>This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers.</p> <p>With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.</p>
SeSystemEnvironmentPrivilege	Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
SeSystemProfilePrivilege	Profile system performance	<p>Required to gather profiling information for the entire system.</p> <p>With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.</p>
SeSystemtimePrivilege	Change the system time	<p>Required to modify the system time.</p> <p>With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.</p>
SeTakeOwnershipPrivilege	Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>
SeTcbPrivilege	Act as part of the operating system	<p>This privilege identifies its holder as part of the trusted computer base.</p> <p>This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.</p>
SeTimeZonePrivilege	Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
SeTrustedCredManAccessPrivil ege	Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
SeUndockPrivilege	Remove computer from docking station	<p>Required to undock a laptop.</p> <p>With this privilege, the user can undock a portable computer from its docking station without logging</p>

SeUnsolicitedInputPrivilege	Not applicable	on. Required to read unsolicited input from a <i>terminal</i> device.
-----------------------------	----------------	--

Security Monitoring Recommendations:

For 4705(S): A user right was removed.

Type of monitoring required	Recommendation
Actions typically performed by the SYSTEM account: This event and certain other events should be monitored to see if they are triggered by any account other than SYSTEM.	Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “ Subject\Security ID ” is not SYSTEM.
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor this event with the “ Subject\Security ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor this event with the “ Subject\Security ID ” or “ Target Account\Account Name ” that correspond to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	If this event corresponds to a “whitelist-only” action, review the “ Subject\Security ID ” for accounts that are outside the whitelist. If you have specific user rights policies, for example, a whitelist of accounts that can perform certain actions, monitor this event to confirm that it was appropriate that the “ Removed Right ” was removed from “ Target Account\Account Name .”
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.	If this event corresponds to an action you want to monitor for certain account types, review the “ Subject\Security ID ” and “ Target Account\Account Name ” to see whether the account type is as expected. For example, if some accounts have critical user rights which should never be removed, monitor this event for the “ Target Account\Account Name ” and the appropriate rights. As another example, if non-administrative accounts should never be granted certain user rights (for example, SeAuditPrivilege), you might monitor this event, because a right can

	be removed only after it was previously granted.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor this event for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should perform only limited actions, or no actions at all.	Monitor the target Computer : (or other target device) for actions performed by the “ Subject\Security ID ” that you are concerned about. Also be sure to check “ Target Account\Account Name ” to see whether user rights should be removed from that account (or whether that account should have any rights on that computer). For high-value servers or other computers, we recommend that you track this event and investigate whether the specific “ Removed Right ” should be removed from “ Target Account\Account Name ” in each case.
User rights that should be restricted: You might have a list of user rights that you want to monitor.	Monitor this event and compare the “ Removed Right ” to your list of restricted rights. Monitor this event to discover the removal of a right that should never have been granted (for example, SeTcbPrivilege), so that you can investigate further.
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

4670(S): Permissions on an object were changed.

This event also belongs in the Audit File System subcategory, and is described there. See “[4670\(S\): Permissions on an object were changed](#).”

4911(S): Resource attributes of the object were changed.

Event Properties - Event 4911, Microsoft Windows security auditing.

General **Details**

Resource attributes of the object were changed.

Account Domain:	CONTOSO
Logon ID:	0x37925
Object:	
Object Server:	Security
Object Type:	File
Object Name:	C:\Audit Files\HBI Data.txt
Handle ID:	0x49c
Process Information:	
Process ID:	0x67c
Process Name:	C:\Windows\System32\svchost.exe
Resource Attributes:	
Original Security Descriptor:	S:AI
New Security Descriptor:	S:ARAI(RA;ID;;;;WD;("Impact_MS",TI,0x10020,3000))
Log Name:	Security
Source:	Microsoft Windows security
Event ID:	4911
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy **Close**

Event Description:

This event generates when [resource attributes](#) of the file system object were changed. Resource attributes for file or folder can be changed, for example, using Windows File Explorer (object's Properties->Classification tab).

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4911</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13570</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-09T23:43:04.009319300Z" />
<EventRecordID>1183714</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="524" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x37925</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Audit Files\HBI Data.txt</Data>
<Data Name="HandleId">0x49c</Data>
<Data Name="OldSd">S:AI</Data>
<Data Name="NewSd">S:ARAI(RA;ID;;;;WD;("Impact_MS",TI,0x10020,3000))</Data>
```

```
<Data Name="ProcessId">0x67c</Data>
<Data Name="ProcessName">C:\Windows\System32\svchost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that changed the resource attributes of the file system object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that changed the resource attributes of the file system object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString]: has "**Security**" value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation. Always "**File**" for this event.

The following table contains the list of the most common **Object Types**:

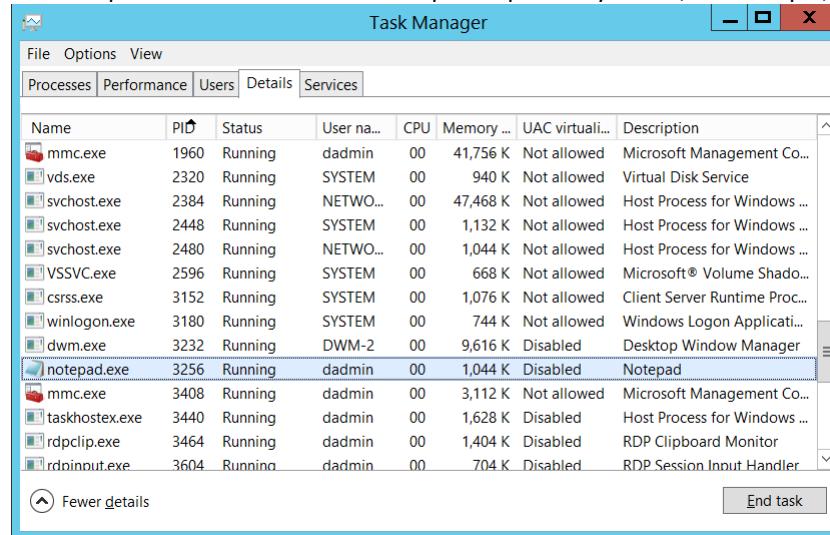
Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: full path and/or name of the object for which resource attributes were changed.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process through which the resource attributes of the file system object were changed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):

Task Manager



Name	PID	Status	User na...	CPU	Memory ...	UAC virtuali...	Description
mmc.exe	1960	Running	dadmin	00	41,756 K	Not allowed	Microsoft Management Co...
vds.exe	2320	Running	SYSTEM	00	940 K	Not allowed	Virtual Disk Service
svchost.exe	2384	Running	NETWO...	00	47,468 K	Not allowed	Host Process for Windows ...
svchost.exe	2448	Running	SYSTEM	00	1,132 K	Not allowed	Host Process for Windows ...
svchost.exe	2480	Running	NETWO...	00	1,044 K	Not allowed	Host Process for Windows ...
VSSVC.exe	2596	Running	SYSTEM	00	668 K	Not allowed	Microsoft® Volume Shado...
csrss.exe	3152	Running	SYSTEM	00	1,076 K	Not allowed	Client Server Runtime Proc...
winlogon.exe	3180	Running	SYSTEM	00	744 K	Not allowed	Windows Logon Applicati...
dwm.exe	3232	Running	DWM-2	00	9,616 K	Disabled	Desktop Window Manager
notepad.exe	3256	Running	dadmin	00	1,044 K	Disabled	Notepad
mmc.exe	3408	Running	dadmin	00	3,112 K	Not allowed	Microsoft Management Co...
taskhostex.exe	3440	Running	dadmin	00	1,628 K	Disabled	Host Process for Windows ...
rdpclip.exe	3464	Running	dadmin	00	1,404 K	Disabled	RDP Clipboard Monitor
rdoinputout.exe	3604	Running	dadmin	00	704 K	Disabled	RDP Session Input Handler

Fewer details End task

If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

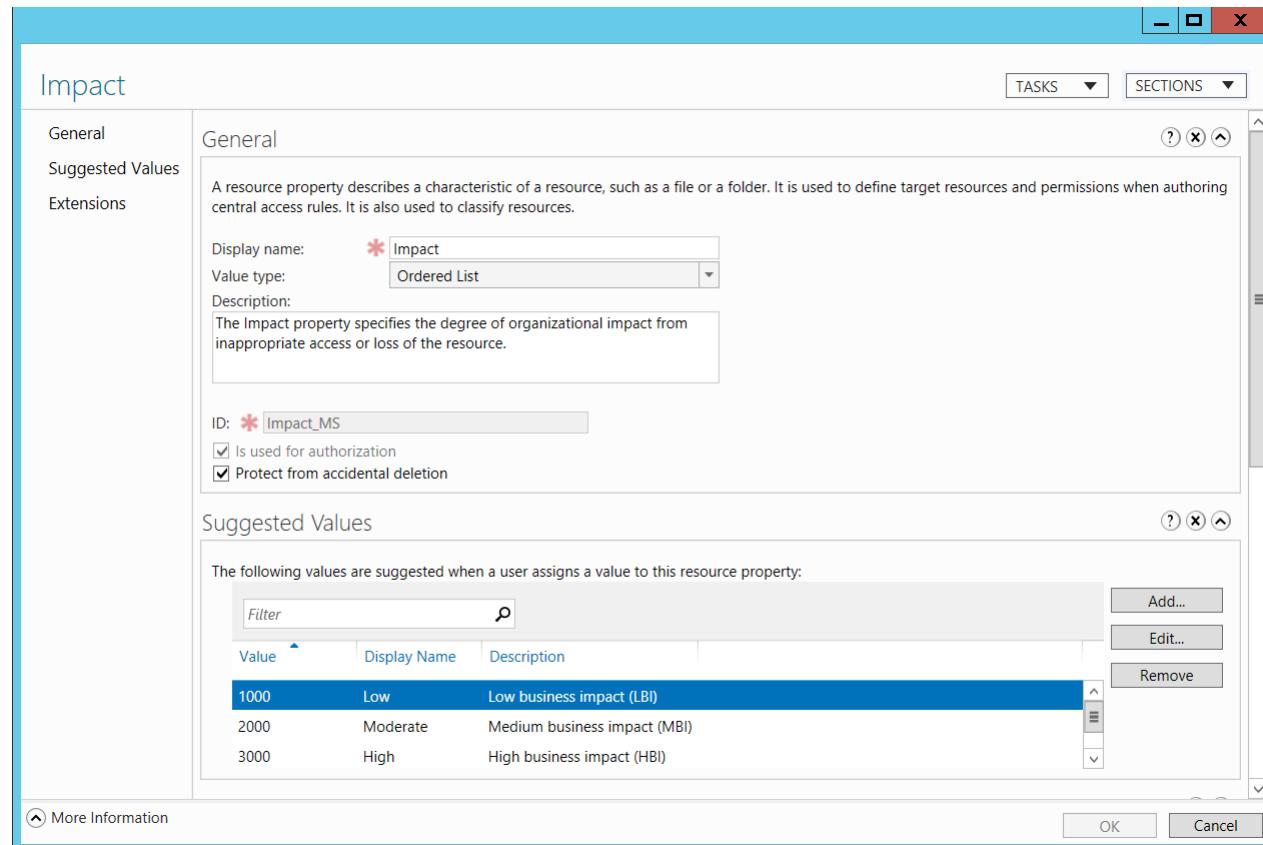
- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Resource Attributes:

- **Original Security Descriptor** [Type = UnicodeString]: the Security Descriptor Definition Language (SDDL) value for the old resource attributes.

For example: S:AI(RA;ID;;;;WD;(**Impact_MS**,Tl,0x10020,**3000**))

- Impact_MS: Resource Property ID.
- 3000: Recourse Property Value.



If no resource attributes were set to the object, then SDDL will not contain any attributes, for example “**S:AI**”.

- **New Security Descriptor** [Type = UnicodeString]: the Security Descriptor Definition Language (SDDL) value for the new resource attributes. See more information in **Resource Attributes\Original Security Descriptor** field section for this event.

The **Security Descriptor Definition Language (SDDL)** defines string elements for enumerating information contained in the security descriptor.

Example:

O:BA_G:SYD:(D;;0xf0007;;AN)(D;;0xf0007;;BG)(A;;0xf0007;;SY)(A;;0x7;;BA)_S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;WD)

- **O**: = Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user

"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- G: = Primary Group.
- D: = DACL Entries.
- S: = SACL Entries.

DACL/SACL entry format: `entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)`

Example: `D:(A;;FA;;;WD)`

- entry_type:
 - "D" - DACL
 - "S" - SACL
- inheritance_flags:
 - "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.
 - "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" Is not also set.
 - "AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.
- ace_type:
 - "A" - ACCESS ALLOWED
 - "D" - ACCESS DENIED
 - "OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).
 - "OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).
 - "AU" - SYSTEM AUDIT
 - "A" - SYSTEM ALARM
 - "OU" - OBJECT SYSTEM AUDIT

"OL" - OBJECT SYSTEM ALARM

- ace_flags:

"CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.

"OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.

"NP" - NO PROPAGATE: only immediate children inherit this ace.

"IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.

"ID" - ACE IS INHERITED

"SA" - SUCCESSFUL ACCESS AUDIT

"FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A

- inherit_object_guid: N/A

- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,

[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 4911(S): Resource attributes of the object were changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
- If you need to monitor events related to specific Windows object types (“Object Type”), for example File or Key, monitor this event for the corresponding “Object Type.” If you need to monitor all changes to specific files or folders (in this case, changes to resource attributes), monitor for the “**Object Name**” that corresponds to the file or folder.
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- You can track changes when, for example, a file was marked as High Impact, or was changed from High Impact to Medium Impact, or a resource was marked as a data type for a specific department and so on. This event can help track changes and resource attribute assignments, which you can see in “**Original Security Descriptor**” and “**New Security Descriptor**” fields.

4913(S): Central Access Policy on the object was changed.

Event Properties - Event 4913, Microsoft Windows security auditing.

General Details

Security ID: CONTOSO\dadmin
Account Name: dadmin
Account Domain: CONTOSO
Logon ID: 0x37901

Object:
Object Server: Security
Object Type: File
Object Name: C:\Audit Files\HBI Data.txt
Handle ID: 0x3d4

Process Information:
Process ID: 0x884
Process Name: C:\Windows\System32\dllhost.exe

Central Policy ID:
Original Security Descriptor: S:AI
New Security Descriptor: S:ARAI(SP;ID::S-1-17-1442530252-1178042555-1247349694-2318402534)

Log Name: Security
Source: Microsoft Windows security
Event ID: 4913
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 11/9/2015 3:40:43 PM
Task Category: Authorization Policy Change
Keywords: Audit Success
Computer: DC01.contoso.local

[Copy](#) [Close](#)

Event Description:

This event generates when a [Central Access Policy](#) on a file system object is changed. This event always generates, regardless of the object’s [SACL](#) settings.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4913</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13570</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-11-09T23:40:43.118758100Z" />
  <EventRecordID>1183666</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="524" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
```

```
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x37901</Data>
<Data Name="ObjectServer">Security</Data>
<Data Name="ObjectType">File</Data>
<Data Name="ObjectName">C:\Audit Files\HBI Data.txt</Data>
<Data Name="HandleId">0x3d4</Data>
<Data Name="OldSd">S:AI</Data>
<Data Name="NewSd">S:ARAI(SP;ID;;;;S-1-17-1442530252-1178042555-1247349694-2318402534)</Data>
<Data Name="ProcessId">0x884</Data>
<Data Name="ProcessName">C:\Windows\System32\dllhost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that changed the Central Access Policy on the object. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that changed the Central Access Policy on the object.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString]: has "**Security**" value for this event.

- **Object Type** [Type = UnicodeString]: The type of an object that was accessed during the operation. Always “File” for this event.

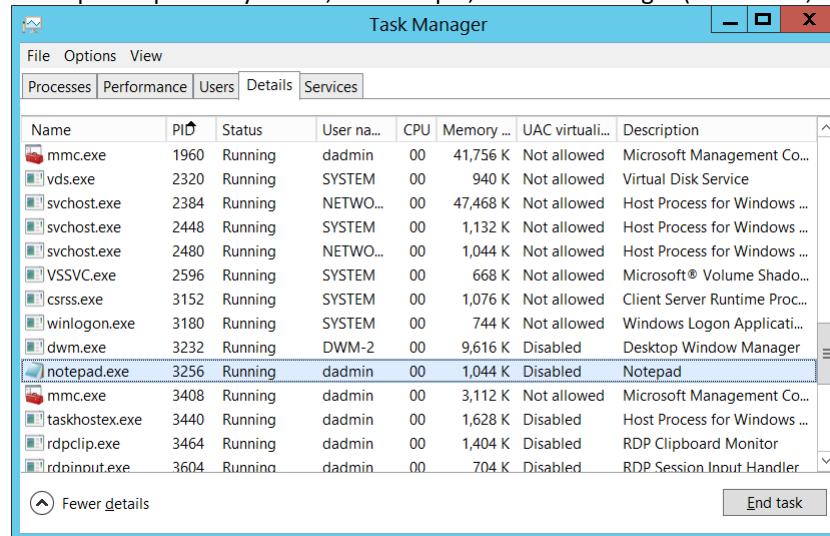
The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Adapter
Key	WaitablePort	Callback	Semaphore
Job	Port	FilterConnectionPort	ALPC Port

- **Object Name** [Type = UnicodeString]: full path and/or name of the object on which the Central Access Policy was changed.
- **Handle ID** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “[4663\(S\)](#): An attempt was made to access an object.” This parameter might not be captured in the event, and in that case appears as “0x0”.

Process:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process using which Central Access Policy was changed. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID** field.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

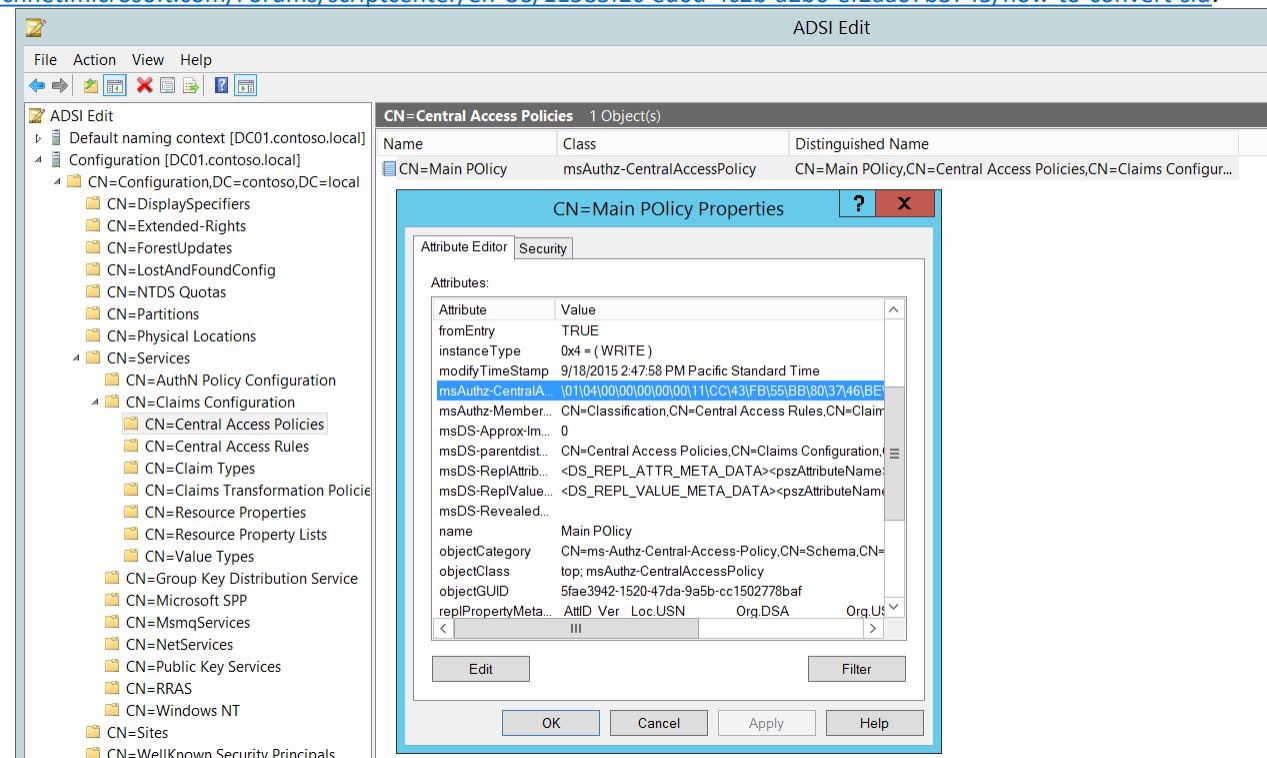
Central Policy ID:

- **Original Security Descriptor [Type = UnicodeString]:** the Security Descriptor Definition Language (SDDL) value for the old Central Policy ID (for the policy that was formerly applied to the object).

SDDL contains Central Access Policy SID, here is an example: S:ARAI(SP;ID;;;;**S-1-17-1442530252-1178042555-1247349694-2318402534**), Central Access Policy SID here is "**S-1-17-1442530252-1178042555-1247349694-2318402534**". To resolve this SID to the real Central Access Policy name you need to do the following:

1. Find Central Access Policy Active Directory object in: "CN=Central Access Policies,CN=Claims Configuration,CN=Services,CN=Configuration,DC=XXX,DC=XX" Active Directory container.
2. Open object's "**Properties**".
3. Find "**msAuthz-CentralAccessPolicyID**" attribute.
4. Convert hexadecimal value to SID (string). Here you can see more information about how to perform this action:

<https://social.technet.microsoft.com/Forums/scriptcenter/en-US/11585f2c-ed0d-4c2b-a2b6-ef2aa07b3745/how-to-convert-sid>.



If no Central Access Policies were applied to the object, then SDDL will not contain any SIDs, for example "**S:AI**".

- **New Security Descriptor [Type = UnicodeString]:** the Security Descriptor Definition Language (SDDL) value for the new Central Policy ID (for the policy that has been applied to the object). See more information in **Central Policy ID\Original Security Descriptor** field section for this event.

The Security Descriptor Definition Language (SDDL) defines string elements for enumerating information contained in the security descriptor.

Example:

O:BA;G:SYD:(D;;0xf0007;;;AN)(D;;0xf0007;;;BG)(A;;0xf0007;;;SY)(A;;0x7;;;BA)S:ARAI(AU;SAFA;DCLCRPCRSWDWO;;;WD)

- **O:** = Owner. SID of specific security principal, or reserved (pre-defined) value, for example: **BA** (BUILTIN_ADMINISTRATORS), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc.

See the list of possible values in the table below:

Value	Description	Value	Description
"AO"	Account operators	"PA"	Group Policy administrators
"RU"	Alias to allow previous Windows 2000	"IU"	Interactively logged-on user
"AN"	Anonymous logon	"LA"	Local administrator
"AU"	Authenticated users	"LG"	Local guest
"BA"	Built-in administrators	"LS"	Local service account
"BG"	Built-in guests	"SY"	Local system
"BO"	Backup operators	"NU"	Network logon user
"BU"	Built-in users	"NO"	Network configuration operators
"CA"	Certificate server administrators	"NS"	Network service account
"CG"	Creator group	"PO"	Printer operators
"CO"	Creator owner	"PS"	Personal self
"DA"	Domain administrators	"PU"	Power users
"DC"	Domain computers	"RS"	RAS servers group
"DD"	Domain controllers	"RD"	Terminal server users
"DG"	Domain guests	"RE"	Replicator
"DU"	Domain users	"RC"	Restricted code
"EA"	Enterprise administrators	"SA"	Schema administrators
"ED"	Enterprise domain controllers	"SO"	Server operators
"WD"	Everyone	"SU"	Service logon user

- **G:** = Primary Group.

- **D:** = DACL Entries.

- **S:** = SACL Entries.

DACL/SACL entry format: **entry_type:inheritance_flags(ace_type;ace_flags;rights;object_guid;inherit_object_guid;account_sid)**

Example: **D:(A;;FA;;;WD)**

- **entry_type:**

 "D" - DACL

 "S" - SACL

- **inheritance_flags:**

 "P" - SDDL_PROTECTED, Inheritance from containers that are higher in the folder hierarchy are blocked.

 "AI" - SDDL_AUTO_INHERITED, Inheritance is allowed, assuming that "P" is not also set.

"AR" - SDDL_AUTO_INHERIT_REQ, Child objects inherit permissions from this object.

- ace_type:

"A" - ACCESS ALLOWED

"D" - ACCESS DENIED

"OA" - OBJECT ACCESS ALLOWED: only applies to a subset of the object(s).

"OD" - OBJECT ACCESS DENIED: only applies to a subset of the object(s).

"AU" - SYSTEM AUDIT

"A" - SYSTEM ALARM

"OU" - OBJECT SYSTEM AUDIT

"OL" - OBJECT SYSTEM ALARM

- ace_flags:

"CI" - CONTAINER INHERIT: Child objects that are containers, such as directories, inherit the ACE as an explicit ACE.

"OI" - OBJECT INHERIT: Child objects that are not containers inherit the ACE as an explicit ACE.

"NP" - NO PROPAGATE: only immediate children inherit this ace.

"IO" - INHERITANCE ONLY: ace doesn't apply to this object, but may affect children via inheritance.

"ID" - ACE IS INHERITED

"SA" - SUCCESSFUL ACCESS AUDIT

"FA" - FAILED ACCESS AUDIT

- rights: A hexadecimal string which denotes the access mask or reserved value, for example: **FA** (File All Access), **FX** (File Execute), **FW** (File Write), etc.

Value	Description	Value	Description
Generic access rights		Directory service access rights	
"GA"	GENERIC ALL	"RC"	Read Permissions
"GR"	GENERIC READ	"SD"	Delete
"GW"	GENERIC WRITE	"WD"	Modify Permissions
"GX"	GENERIC EXECUTE	"WO"	Modify Owner
File access rights		"RP"	Read All Properties
"FA"	FILE ALL ACCESS	"WP"	Write All Properties
"FR"	FILE GENERIC READ	"CC"	Create All Child Objects
"FW"	FILE GENERIC WRITE	"DC"	Delete All Child Objects
"FX"	FILE GENERIC EXECUTE	"LC"	List Contents
Registry key access rights		"SW"	All Validated Writes
"KA"	"LO"	"LO"	List Object
"K"	KEY READ	"DT"	Delete Subtree
"KW"	KEY WRITE	"CR"	All Extended Rights
"KX"	KEY EXECUTE		

- object_guid: N/A
- inherit_object_guid: N/A
- account_sid: SID of specific security principal, or reserved value, for example: **AN** (Anonymous), **WD** (Everyone), **SY** (LOCAL_SYSTEM), etc. See the table above for more details.

For more information about SDDL syntax, see these articles: <https://msdn.microsoft.com/en-us/library/cc230374.aspx>,
[https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/aa374892(v=vs.85).aspx).

Security Monitoring Recommendations:

For 4913(S): Central Access Policy on the object was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.
- If you need to monitor events related to specific Windows object types (“Object Type”), for example File or Key, monitor this event for the corresponding “Object Type.” If you need to monitor all changes to specific files or folders (in this case, changes to the Central Access Policy), monitor for the “**Object Name**” that corresponds to the file or folder.
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If you have specific files, folders, or entire systems to which a specific Central Access Policy should be applied, you can monitor this event and compare the Central Access Policy SID in “**New Security Descriptor**” to see if it matches the expected policy.

Audit Filtering Platform Policy Change

Audit Filtering Platform Policy Change allows you to audit events generated by changes to the [Windows Filtering Platform \(WFP\)](#), such as the following:

- IPsec services status.
- Changes to IPsec policy settings.
- Changes to Windows Filtering Platform Base Filtering Engine policy settings.
- Changes to WFP providers and engine.

Windows Filtering Platform (WFP) enables independent software vendors (ISVs) to filter and modify TCP/IP packets, monitor or authorize connections, filter Internet Protocol security (IPsec)-protected traffic, and filter remote procedure calls (RPCs).

This subcategory is outside the scope of this document.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	-	-	-	-	This subcategory is outside the scope of this document.
Member Server	-	-	-	-	This subcategory is outside the scope of this document.
Workstation	-	-	-	-	This subcategory is outside the scope of this document.

4709(**S**): IPsec Services was started.

4710(**S**): IPsec Services was disabled.

4711(**S**): May contain any one of the following:

4712(**F**): IPsec Services encountered a potentially serious failure.

5040(**S**): A change has been made to IPsec settings. An Authentication Set was added.

5041(**S**): A change has been made to IPsec settings. An Authentication Set was modified.

5042(**S**): A change has been made to IPsec settings. An Authentication Set was deleted.

5043(**S**): A change has been made to IPsec settings. A Connection Security Rule was added.

5044(**S**): A change has been made to IPsec settings. A Connection Security Rule was modified.

5045(**S**): A change has been made to IPsec settings. A Connection Security Rule was deleted.

5046(**S**): A change has been made to IPsec settings. A Crypto Set was added.

5047(**S**): A change has been made to IPsec settings. A Crypto Set was modified.

5048(**S**): A change has been made to IPsec settings. A Crypto Set was deleted.

5440(**S**): The following callout was present when the Windows Filtering Platform Base Filtering Engine started.

5441(**S**): The following filter was present when the Windows Filtering Platform Base Filtering Engine started.

5442(**S**): The following provider was present when the Windows Filtering Platform Base Filtering Engine started.

5443(**S**): The following provider context was present when the Windows Filtering Platform Base Filtering Engine started.

5444(**S**): The following sub-layer was present when the Windows Filtering Platform Base Filtering Engine started.

5446(**S**): A Windows Filtering Platform callout has been changed.

5448(**S**): A Windows Filtering Platform provider has been changed.

5449(**S**): A Windows Filtering Platform provider context has been changed.

5450(**S**): A Windows Filtering Platform sub-layer has been changed.

5456(**S**): PAStore Engine applied Active Directory storage IPsec policy on the computer.

5457(**F**): PAStore Engine failed to apply Active Directory storage IPsec policy on the computer.

5458(**S**): PAStore Engine applied locally cached copy of Active Directory storage IPsec policy on the computer.

5459(**F**): PAStore Engine failed to apply locally cached copy of Active Directory storage IPsec policy on the computer.

5460(**S**): PAStore Engine applied local registry storage IPsec policy on the computer.

5461(**F**): PAStore Engine failed to apply local registry storage IPsec policy on the computer.

5462(**F**): PAStore Engine failed to apply some rules of the active IPsec policy on the computer. Use the IP Security Monitor snap-in to diagnose the problem.

5463(**S**): PAStore Engine polled for changes to the active IPsec policy and detected no changes.

5464(**S**): PAStore Engine polled for changes to the active IPsec policy, detected changes, and applied them to IPsec Services.

5465(**S**): PAStore Engine received a control for forced reloading of IPsec policy and processed the control successfully.

5466(**F**): PAStore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory cannot be reached, and will use the cached copy of the Active Directory IPsec policy instead. Any changes made to the Active Directory IPsec policy since the last poll could not be applied.

5467(**F**): PAStore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory can be reached, and found no changes to the policy. The cached copy of the Active Directory IPsec policy is no longer being used.

5468(**S**): PAStore Engine polled for changes to the Active Directory IPsec policy, determined that Active Directory can be reached, found changes to the policy, and applied those changes. The cached copy of the Active Directory IPsec policy is no longer being used.

5471(**S**): PAStore Engine loaded local storage IPsec policy on the computer.

5472(**F**): PAStore Engine failed to load local storage IPsec policy on the computer.

5473(**S**): PAStore Engine loaded directory storage IPsec policy on the computer.

5474(**F**): PAStore Engine failed to load directory storage IPsec policy on the computer.

5477(**F**): PAStore Engine failed to add quick mode filter.

Audit MPSSVC Rule-Level Policy Change

Audit MPSSVC Rule-Level Policy Change determines whether the operating system generates audit events when changes are made to policy rules for the Microsoft Protection Service (MPSSVC.exe).

The Microsoft Protection Service, which is used by Windows Firewall, is an integral part of the computer's threat protection against malware. The tracked activities include:

- Active policies when the Windows Firewall service starts.
- Changes to Windows Firewall rules.
- Changes to the Windows Firewall exception list.
- Changes to Windows Firewall settings.
- Rules ignored or not applied by the Windows Firewall service.
- Changes to Windows Firewall Group Policy settings.

Changes to firewall rules are important for understanding the security state of the computer and how well it is protected against network attacks.

Event volume: Medium.

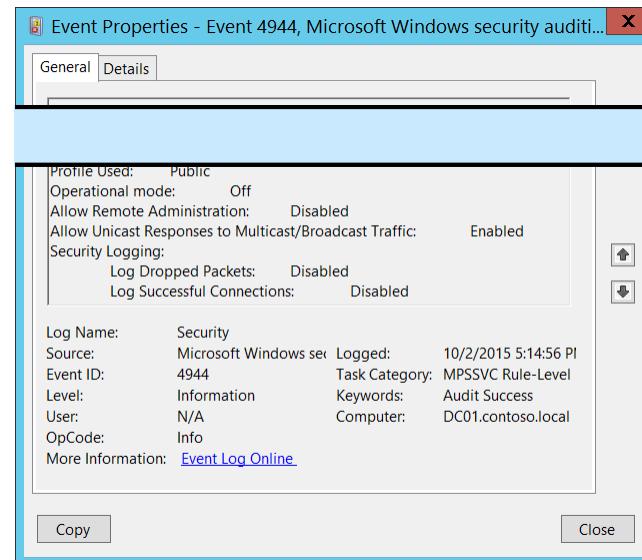
Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	Success events shows you changes in Windows Firewall rules and settings, active configuration and rules after Windows Firewall Service startup and default configuration restore actions. Failure events may help to identify configuration problems with Windows Firewall rules or settings.
Member Server	Yes	Yes	Yes	Yes	Success events shows you changes in Windows Firewall rules and settings, active configuration and rules after Windows Firewall Service startup and default configuration restore actions. Failure events may help to identify configuration problems with Windows Firewall rules or settings.
Workstation	Yes	Yes	Yes	Yes	Success events shows you changes in Windows Firewall rules and settings, active configuration and rules after Windows Firewall Service startup and default configuration restore actions. Failure events may help to identify configuration problems with Windows Firewall rules or settings.

Events List:

- [4944\(S\)](#): The following policy was active when the Windows Firewall started.
- [4945\(S\)](#): A rule was listed when the Windows Firewall started.
- [4946\(S\)](#): A change has been made to Windows Firewall exception list. A rule was added.
- [4947\(S\)](#): A change has been made to Windows Firewall exception list. A rule was modified.
- [4948\(S\)](#): A change has been made to Windows Firewall exception list. A rule was deleted.
- [4949\(S\)](#): Windows Firewall settings were restored to the default values.
- [4950\(S\)](#): A Windows Firewall setting has changed.
- [4951\(F\)](#): A rule has been ignored because its major version number was not recognized by Windows Firewall.
- [4952\(F\)](#): Parts of a rule have been ignored because its minor version number was not recognized by Windows Firewall. The other parts of the rule will be enforced.
- [4953\(F\)](#): A rule has been ignored by Windows Firewall because it could not parse the rule.
- [4954\(S\)](#): Windows Firewall Group Policy settings have changed. The new settings have been applied.

- [4956\(S\)](#): Windows Firewall has changed the active profile.
- [4957\(F\)](#): Windows Firewall did not apply the following rule:
- [4958\(F\)](#): Windows Firewall did not apply the following rule because the rule referred to items not configured on this computer:

4944(S): The following policy was active when the Windows Firewall started.

 Event Properties - Event 4944, Microsoft Windows security auditi... X

General		Details	
Profile Used:	Public	Operational mode:	Off
Allow Remote Administration:	Disabled	Allow Unicast Responses to Multicast/Broadcast Traffic:	Enabled
Security Logging:	Log Dropped Packets: Disabled Log Successful Connections: Disabled		
Log Name:	Security	Source:	Microsoft Windows sev
Event ID:	4944	Logged:	10/2/2015 5:14:56 PI
Level:	Information	Task Category:	MPSSVC Rule-Level
User:	N/A	Keywords:	Audit Success
OpCode:	Info	Computer:	DC01.contoso.local
More Information: Event Log Online			
Copy		Close	

Event Description:

This event generates every time Windows Firewall service starts.

This event shows Windows Firewall settings that were in effect when the Windows Firewall service started.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

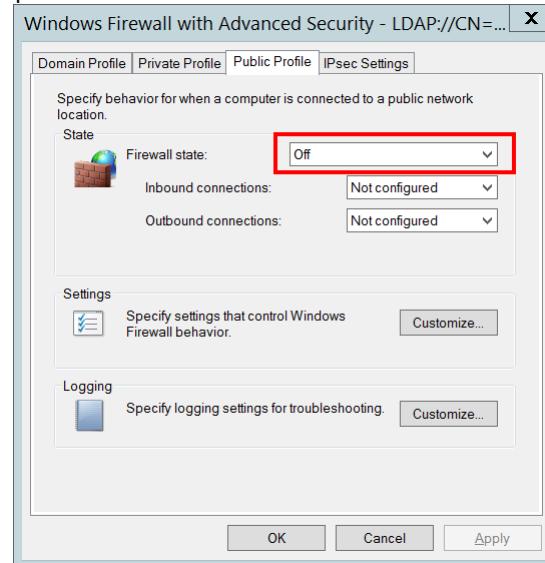
```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4944</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-03T00:14:56.644728300Z" />
<EventRecordID>1050808</EventRecordID>
```

```
<Correlation />
<Execution ProcessID="500" ThreadID="2216" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="GroupPolicyApplied">No</Data>
<Data Name="Profile">Public</Data>
<Data Name="OperationMode">Off</Data>
<Data Name="RemoteAdminEnabled">Disabled</Data>
<Data Name="MulticastFlowsEnabled">Enabled</Data>
<Data Name="LogDroppedPacketsEnabled">Disabled</Data>
<Data Name="LogSuccessfulConnectionsEnabled">Disabled</Data>
</EventData>
```

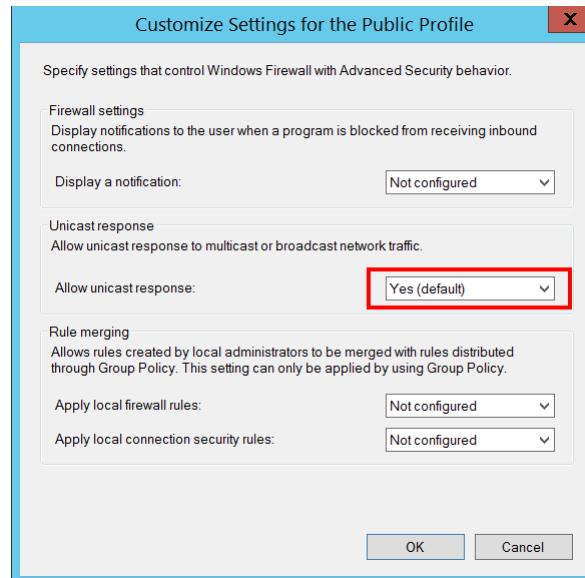
</Event>

Required Server Roles: None.**Minimum OS Version:** Windows Server 2008, Windows Vista.**Event Versions:** 0.**Field Descriptions:****Group Policy Applied** [Type = UnicodeString]: it always has “No” value for this event. This field should show information about: was Group Policy applied for Windows Firewall when it starts or not.**Profile Used** [Type = UnicodeString]: shows the active profile name for the moment Windows Firewall service starts. It always has value “**Public**” for this event, because when this event generates, the active profile is not switched to “**Domain**” or “**Private**”. Typically you will see “[4956\(S\): Windows Firewall has changed the active profile](#)” after this event, which will tell you the real active profile.**Operational mode** [Type = UnicodeString]:

- **On** – if “**Firewall state:**” setting was set to “On” for “Public” profile.
- **Off** - if “**Firewall state:**” setting was set to “Off” for “Public” profile.

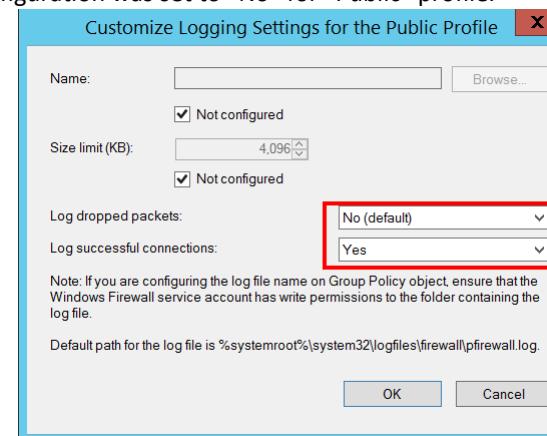
**Allow Remote Administration** [Type = UnicodeString]: looks like this setting is connected to “[Windows Firewall: Allow remote administration exception](#)” Group Policy setting, but it is always Disabled, no matter which option is set for “[Windows Firewall: Allow remote administration exception](#)” Group Policy.**Allow Unicast Responses to Multicast/Broadcast Traffic** [Type = UnicodeString]:

- **Enabled** - if “**Allow unicast response:**” Settings configuration was set to “Yes” for “Public” profile.
- **Disabled** - if “**Allow unicast response:**” Settings configuration was set to “No” for “Public” profile.



Security Logging:

- **Log Dropped Packets** [Type = UnicodeString]:
 - **Enabled** – if “**Log dropped packets:**” Logging configuration was set to “Yes” for “Public” profile.
 - **Disabled** - if “**Log dropped packets:**” Logging configuration was set to “No” for “Public” profile.
- **Log Successful Connections** [Type = UnicodeString]:
 - **Enabled** - if “**Log successful connections:**” Logging configuration was set to “Yes” for “Public” profile.
 - **Disabled** - if “**Log dropped packets:**” Logging configuration was set to “No” for “Public” profile.

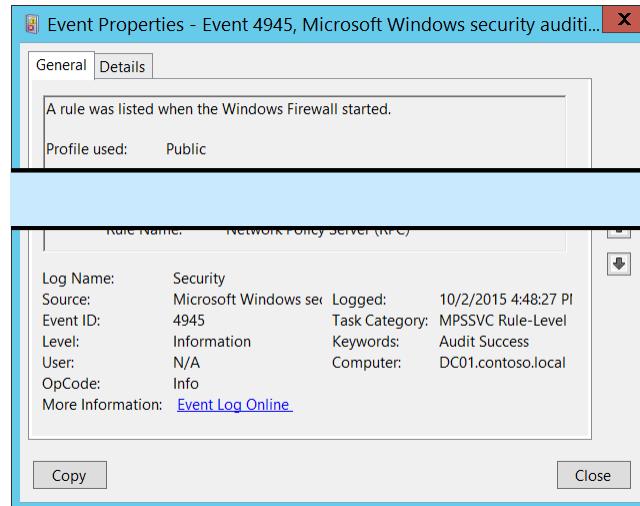


Security Monitoring Recommendations:

For 4944(S): The following policy was active when the Windows Firewall started.

- If you have a standard or baseline for Windows Firewall settings defined for **Public** profile (which can be the same as for Domain, for example), monitor this event and check whether the settings reported by the event are still the same as were defined in your standard or baseline.
- Unfortunately this event shows configuration only for **Public** profile, but you can still compare all the settings with your organization's Windows Firewall baseline for Public profile on different computers and trigger an alert if the configuration is not the same.

4945(S): A rule was listed when the Windows Firewall started.



Event Description:
 This event generates every time Windows Firewall service starts.
 This event shows the inbound and/or outbound rule which was listed when the Windows Firewall started and applied for "Public" profile.
 This event generates per rule.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4945</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>

<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T23:48:27.535295100Z" />
<EventRecordID>1049946</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="4744" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ProfileUsed">Public</Data>
<Data Name="RuleId">NPS-NPSSvc-In-RPC</Data>
<Data Name="RuleName">Network Policy Server (RPC)</Data>

```

</EventData>
</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

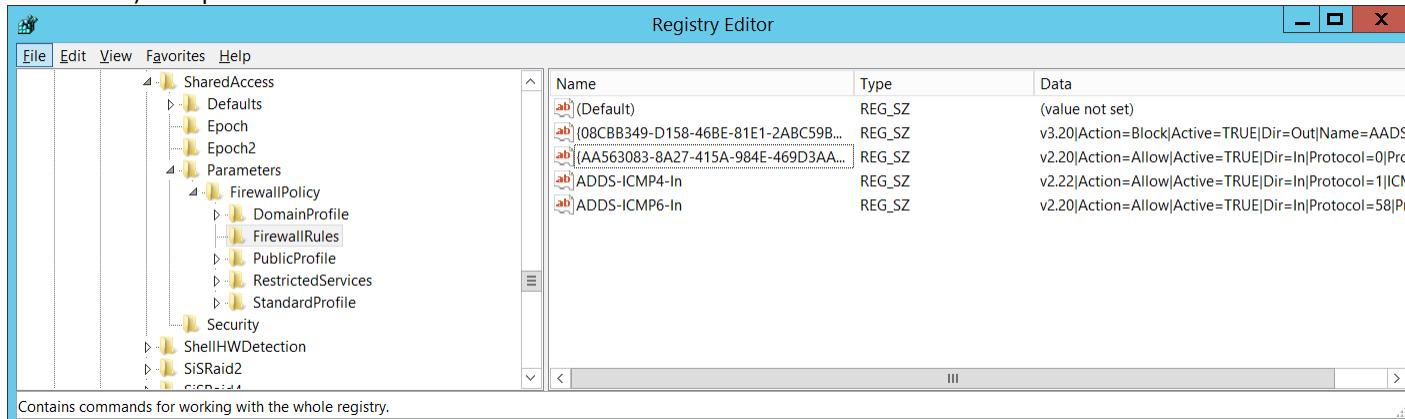
Profile used [Type = UnicodeString]: the name of the profile that the rule belongs to. It always has value “**Public**”, because this event shows rules only for “Public” profile.

Rule:

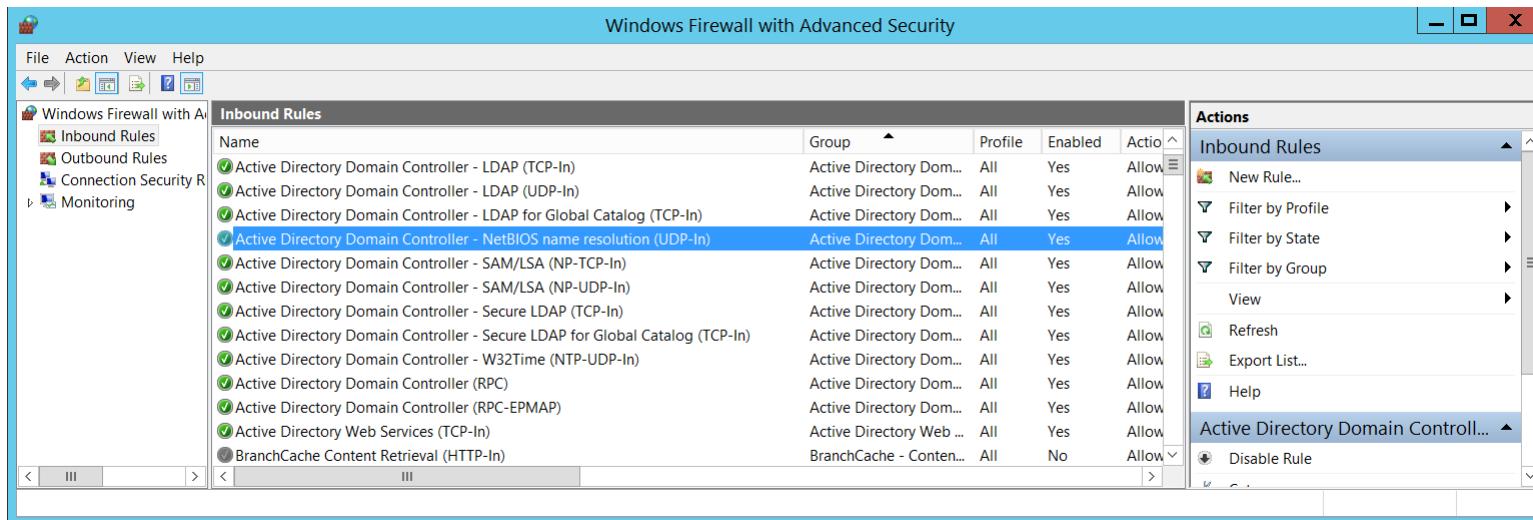
- **Rule ID** [Type = UnicodeString]: the unique firewall rule identifier.

To see the unique ID of the rule you need to navigate to

“**HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules**” registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Rule Name** [Type = UnicodeString]: the name of the rule which was listed when the Windows Firewall started. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (**wf.msc**), check “Name” column:



Security Monitoring Recommendations:

For 4945(S): A rule was listed when the Windows Firewall started.

- Typically this event has an informational purpose.
- Unfortunately this event shows rules only for **Public** profile, but you still can compare this list with your organization's Windows Firewall baseline for Public profile rules on different computers, and trigger an alert if the configuration is not the same.

4946(G): A change has been made to Windows Firewall exception list. A rule was added.

Event Description:

This event generates when new rule was locally added to Windows Firewall.

This event doesn't generate when new rule was added via Group Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event Properties - Event 4946, Microsoft Windows security audit... X

General	Details
---------	---------

A change was made to the Windows Firewall exception list. A rule was added.

Added Rule:
 Rule ID: {F2649D59-1355-4E3C-B886-CDD08B683199}
 Rule Name: Allow All Rule

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 4946
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Logged: 10/3/2015 1:05:42 PM
 Task Category: MPSSVC Rule-Level
 Keywords: Audit Success
 Computer: DC01.contoso.local

Copy Close

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4946</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-03T20:05:42.078367200Z" />
<EventRecordID>1050893</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="528" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="ProfileChanged">All</Data>
  <Data Name="RuleId">{F2649D59-1355-4E3C-B886-CDD08B683199}</Data>
  <Data Name="RuleName">Allow All Rule</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Profile Changed [Type = UnicodeString]: the list of profiles to which new rule was applied. Examples:

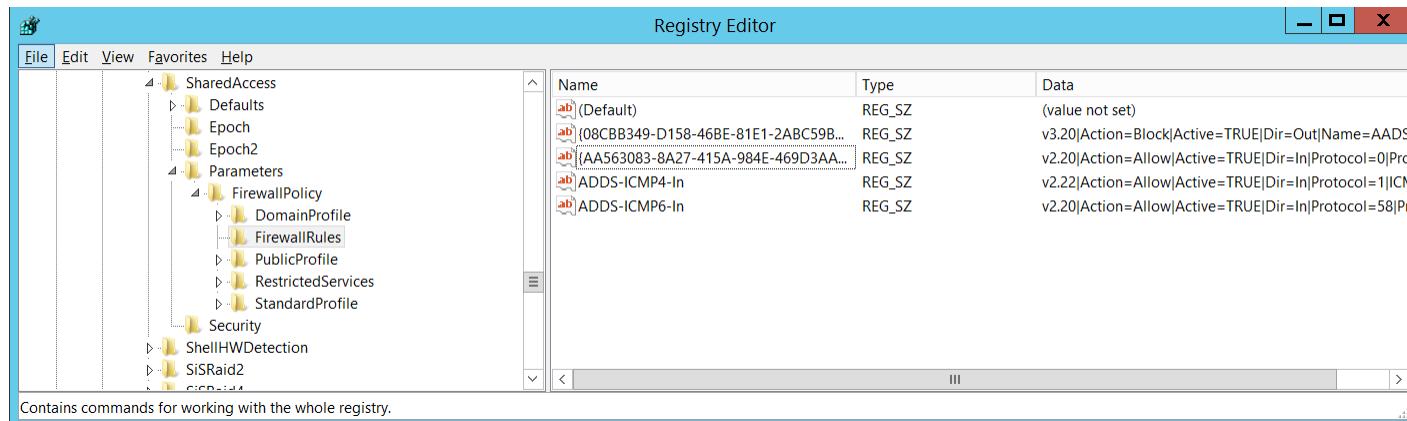
- All
- Domain,Public
- Domain,Private
- Private,Public
- Public
- Domain
- Private

Added Rule:

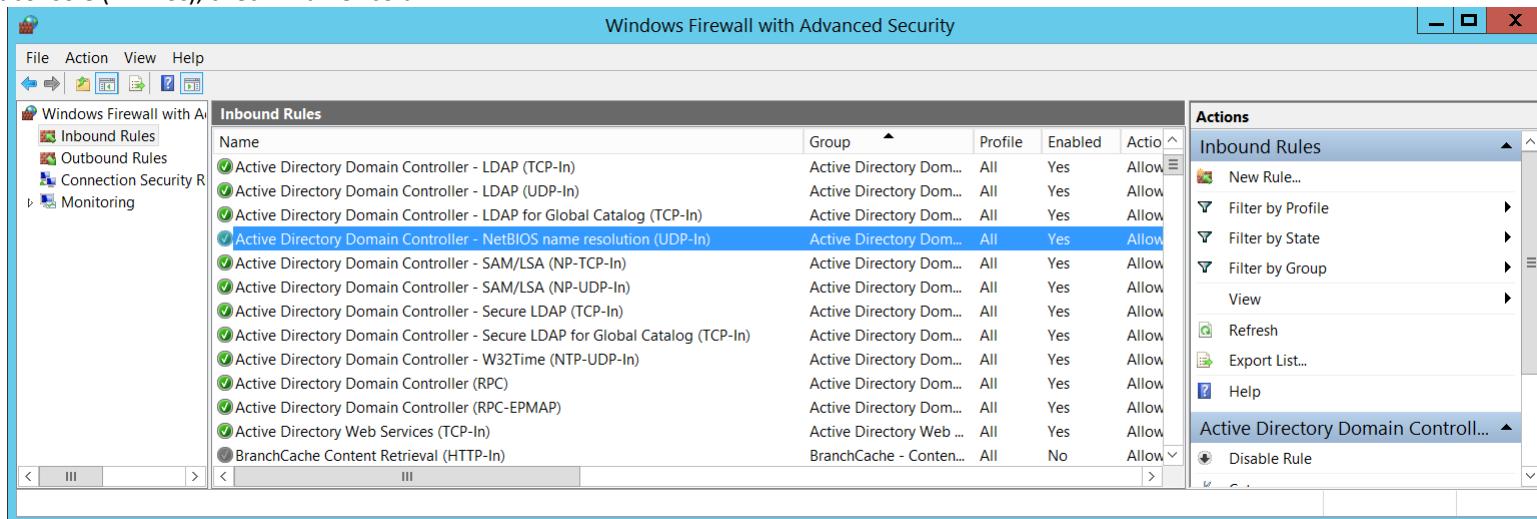
- **Rule ID** [Type = UnicodeString]: the unique new firewall rule identifier.

To see the unique ID of the rule you need to navigate to

"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules" registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Rule Name** [Type = UnicodeString]: the name of the rule which was added. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (`wf.msc`), check “Name” column:

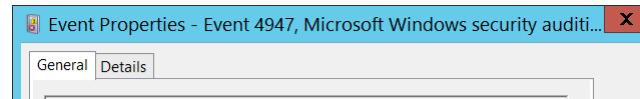


Security Monitoring Recommendations:

For 4946(S): A change has been made to Windows Firewall exception list. A rule was added.

- This event can be helpful in case you want to monitor all creations of new Firewall rules which were done locally.

4947(S): A change has been made to Windows Firewall exception list. A rule was modified.



Event Description:

This event generates when Windows Firewall rule was modified.

This event doesn't generate when Firewall rule was modified via Group Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Profile Changed: All
Modified Rule: Rule ID: {F2649D59-1355-4E3C-B886-CDD08B683199} Rule Name: Allow All Rule
Log Name: Security Source: Microsoft-Windows-Security-Auditing Event ID: 4947 Level: Information User: N/A OpCode: Info More Information: Event Log Online
Logged: 10/3/2015 1:27:22 PM Task Category: MPSSVC Rule-Level Keywords: Audit Success Computer: DC01.contoso.local
Copy
Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4947</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

<TimeCreated SystemTime="2015-10-03T20:27:22.485152000Z" />

<EventRecordID>1050908</EventRecordID>

<Correlation />

<Execution ProcessID="500" ThreadID="3796" />

<Channel>Security</Channel>

<Computer>DC01.contoso.local</Computer>

<Security />

</System>

- <EventData>

<Data Name="ProfileChanged">All</Data>

<Data Name="RuleId">{F2649D59-1355-4E3C-B886-CDD08B683199}</Data>

<Data Name="RuleName">Allow All Rule</Data>

</EventData>

</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Profile Changed [Type = UnicodeString]: the list of profiles to which changed rule is applied. Examples:

- All
- Domain,Public

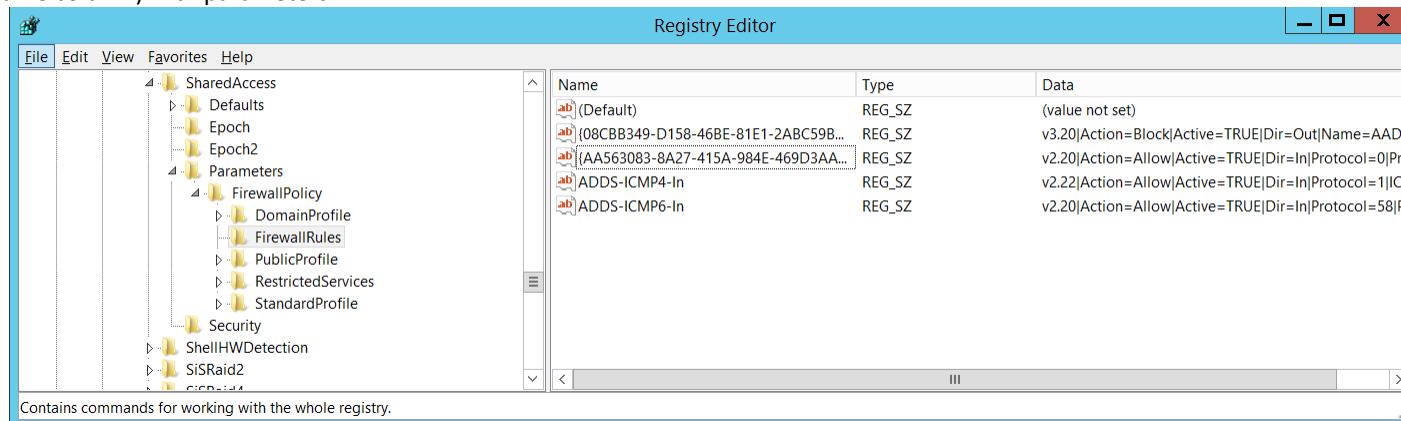
- Domain,Private
- Private,Public
- Public
- Domain
- Private

Modified Rule:

- **Rule ID** [Type = UnicodeString]: the unique identifier for modified firewall rule.

To see the unique ID of the rule you need to navigate to

"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules" registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Rule Name** [Type = UnicodeString]: the name of the rule which was modified. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (**wf.msc**), check "Name" column:

Windows Firewall with Advanced Security

Inbound Rules

Name	Group	Profile	Enabled	Action
Active Directory Domain Controller - LDAP (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - LDAP (UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - LDAP for Global Catalog (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - NetBIOS name resolution (UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - SAM/LSA (NP-TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - SAM/LSA (NP-UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - Secure LDAP (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - Secure LDAP for Global Catalog (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - W32Time (NTP-UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller (RPC)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller (RPC-EPMAP)	Active Directory Dom...	All	Yes	Allow
Active Directory Web Services (TCP-In)	Active Directory Web ...	All	Yes	Allow
BranchCache Content Retrieval (HTTP-In)	BranchCache - Conten...	All	No	Allow

Actions

- New Rule...
- Filter by Profile
- Filter by State
- Filter by Group
- View
- Refresh
- Export List...
- Help
- Active Directory Domain Controll...
- Disable Rule

Security Monitoring Recommendations:

For 4947(S): A change has been made to Windows Firewall exception list. A rule was modified.

- This event can be helpful in case you want to monitor all Firewall rules modifications which were done locally.

4948(S): A change has been made to Windows Firewall exception list. A rule was deleted.

Event Properties - Event 4948, Microsoft Windows security audit...

Event Description:
This event generates when Windows Firewall rule was deleted.
This event doesn't generate when the rule was deleted via Group Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Profile Changed: All

Deleted Rule:
Rule ID: {F2649D59-1355-4E3C-B886-CDD08B683199}
Rule Name: Allow All Rule

Log Name: Security
Source: Microsoft Windows security
Event ID: 4948
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/3/2015 2:19:15 PM
Task Category: MPSSVC Rule-Level
Keywords: Audit Success
Computer: DC01.contoso.local

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4948</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>

```

<TimeCreated SystemTime="2015-10-03T21:19:15.646187500Z" />

```
<EventRecordID>1050934</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="528" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ProfileChanged">All</Data>
<Data Name="RuleId">{F2649D59-1355-4E3C-B886-CDD08B683199}</Data>
<Data Name="RuleName">Allow All Rule</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Profile Changed [Type = UnicodeString]: the list of profiles to which deleted rule was applied. Examples:

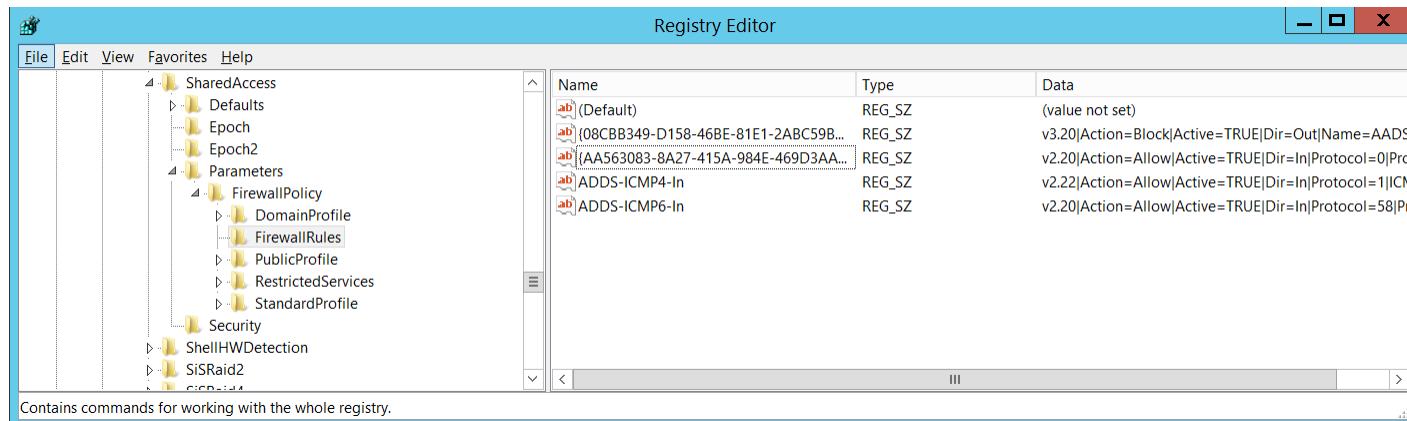
- All
- Domain,Public
- Domain,Private
- Private,Public
- Public
- Domain
- Private

Deleted Rule:

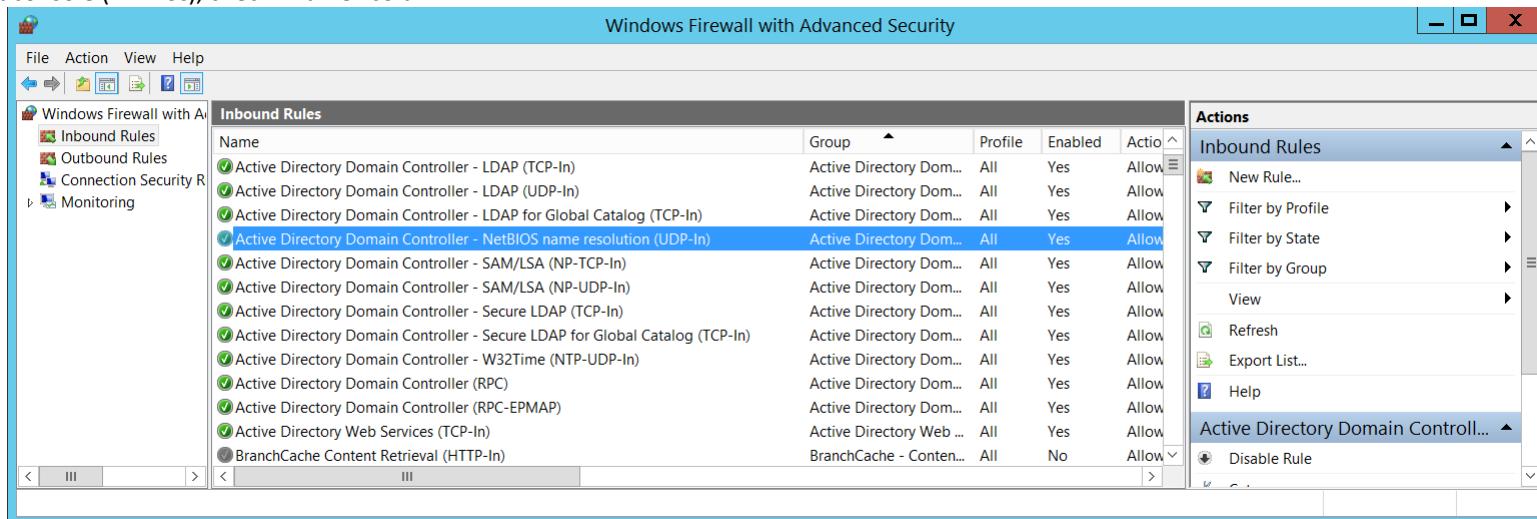
- **Rule ID** [Type = UnicodeString]: the unique identifier for deleted firewall rule.

To see the unique ID of the rule you need to navigate to

"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules" registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Rule Name** [Type = UnicodeString]: the name of the rule which was deleted. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (`wf.msc`), check “Name” column:



Security Monitoring Recommendations:

For 4948(S): A change has been made to Windows Firewall exception list. A rule was deleted.

- This event can be helpful in case you want to monitor all deletions of Firewall rules which were done locally.

4949(S): Windows Firewall settings were restored to the default values.

Event Properties - Event 4949, Microsoft Windows security auditi... X

General Details

Log Name: Security
Source: Microsoft Windows sev
Event ID: 4949
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/2/2015 4:38:28 PM
Task Category: MPSSVC Rule-Level
Keywords: Audit Success
Computer: DC01.contoso.local

Copy Close

Event Description:

This event generates when Windows Firewall settings were locally restored to the default configuration.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4949</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T23:38:28.804003300Z" />
<EventRecordID>1049926</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="3768" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

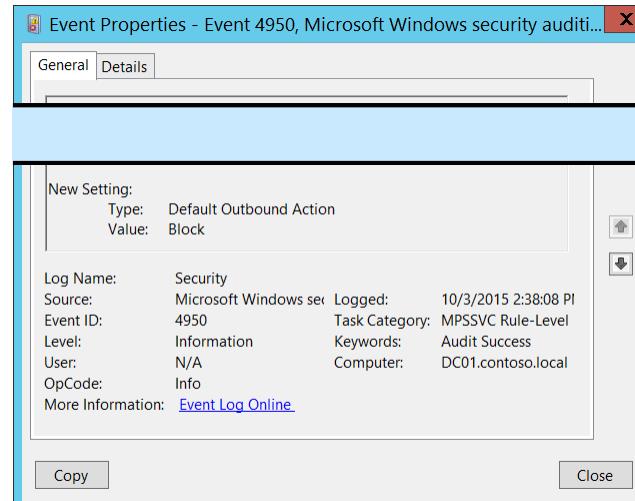
Event Versions: 0.

Security Monitoring Recommendations:

For 4949(S): Windows Firewall settings were restored to the default values.

- You shouldn't see this event during normal Windows Firewall operations, because it should be intentionally done by user or software. This event should be always monitored and an alert should be triggered, especially on critical computers or devices.
- This event can be helpful in case you want to monitor all changes of Firewall rules which were done locally, especially restores to default configuration.

4950(S): A Windows Firewall setting has changed.



Event Description:

This event generates when Windows Firewall local setting was changed.
This event doesn't generate when Windows Firewall setting was changed via Group Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4950</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

<TimeCreated SystemTime="2015-10-03T21:38:08.086908400Z" />

<EventRecordID>1050944</EventRecordID>

<Correlation />

<Execution ProcessID="500" ThreadID="924" />

<Channel>Security</Channel>

<Computer>DC01.contoso.local</Computer>

<Security />

</System>

- <EventData>

<Data Name="ProfileChanged">Domain</Data>

<Data Name="SettingType">Default Outbound Action</Data>

<Data Name="SettingValue">Block</Data>

</EventData>

</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

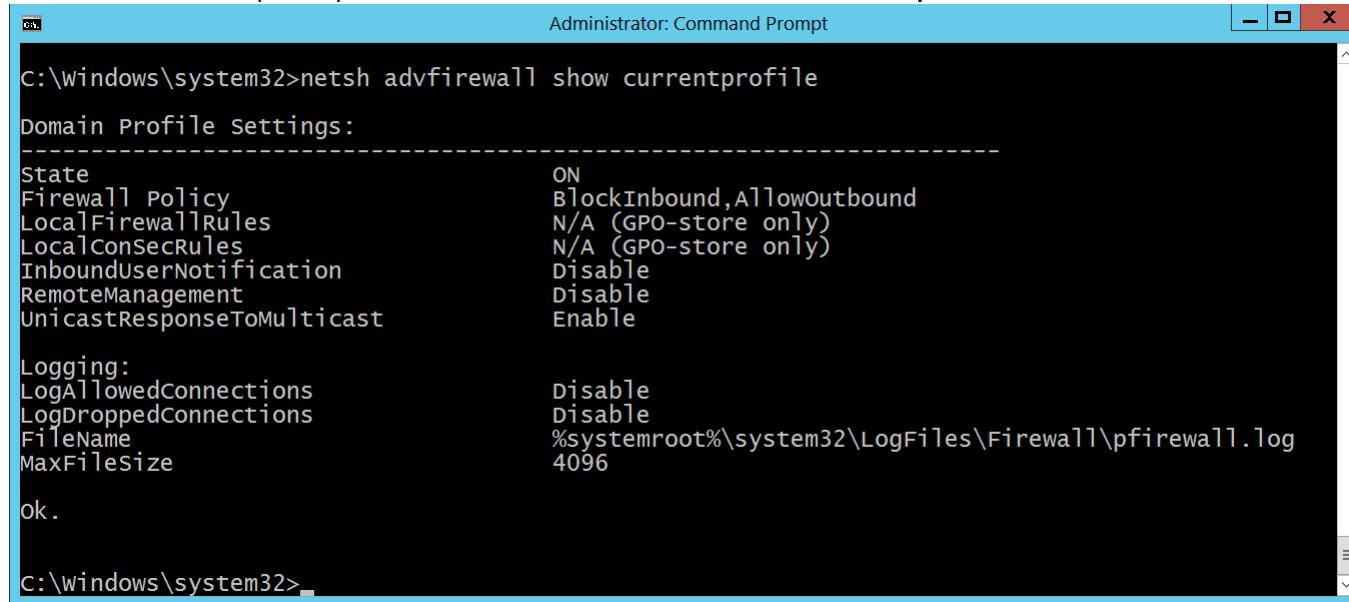
Changed Profile [Type = UnicodeString]: the name of profile in which setting was changed. Possible values are:

- Public
- Domain

- Private

New Setting:

- Type [Type = UnicodeString]: the name of the setting which was modified. You can use “**netsh advfirewall**” command to see or set Windows Firewall settings, for example, to see settings for current\active Windows Firewall profile you need to execute “**netsh advfirewall show currentprofile**” command:



```
Administrator: Command Prompt
C:\windows\system32>netsh advfirewall show currentprofile
Domain Profile settings:
-----
State ON
Firewall Policy BlockInbound,AllowOutbound
LocalFirewallRules N/A (GPO-store only)
LocalConSecRules N/A (GPO-store only)
InboundUserNotification Disable
RemoteManagement Disable
UnicastResponseToMulticast Enable

Logging:
LogAllowedConnections Disable
LogDroppedConnections Disable
FileName %systemroot%\system32\LogFiles\Firewall\pfirewall.log
MaxFilesize 4096

ok.

C:\windows\system32>
```



- Value [Type = UnicodeString]: new value of modified setting.

Security Monitoring Recommendations:

For 4950(S): A Windows Firewall setting has changed.

- If you have a standard or baseline for Windows Firewall settings defined, monitor this event and check whether the settings reported by the event are still the same as were defined in your standard or baseline.
- This event can be helpful in case you want to monitor all changes in Windows Firewall settings which were done locally.

4951(F): A rule has been ignored because its major version number was not recognized by Windows Firewall.

Event Description:

When you create or edit a Windows Firewall rule, the settings that you can include depend upon the version of Windows you use when creating the rule. As new settings are added to later versions of Windows or to service packs for existing versions of Windows, the version number of the rules processing engine is updated, and that version number is stamped into rules that

are created by using that version of Windows. For example, Windows Vista produces firewall rules that are stamped with version "v2.0". Future versions of Windows might use "v2.1", or "v3.0" to indicate, respectively, minor or major changes and additions.

If you create a firewall rule on a newer version of Windows that references firewall settings that are not available on earlier versions of Windows, and then try to deploy that rule to computers running the earlier version of Windows, the firewall engine produces this error to indicate that it cannot process the rule.

The only solution is to remove the incompatible rule, and then deploy a compatible rule.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4951</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-07T21:49:06.951537900Z" />
<EventRecordID>1052309</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="556" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="Profile">All</Data>
<Data Name="RuleId">{08CBB349-D158-46BE-81E1-2ABC59BDD523}</Data>
<Data Name="RuleName">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Profile [Type = UnicodeString]: the name of the profile of the ignored rule. Possible values are:

- All

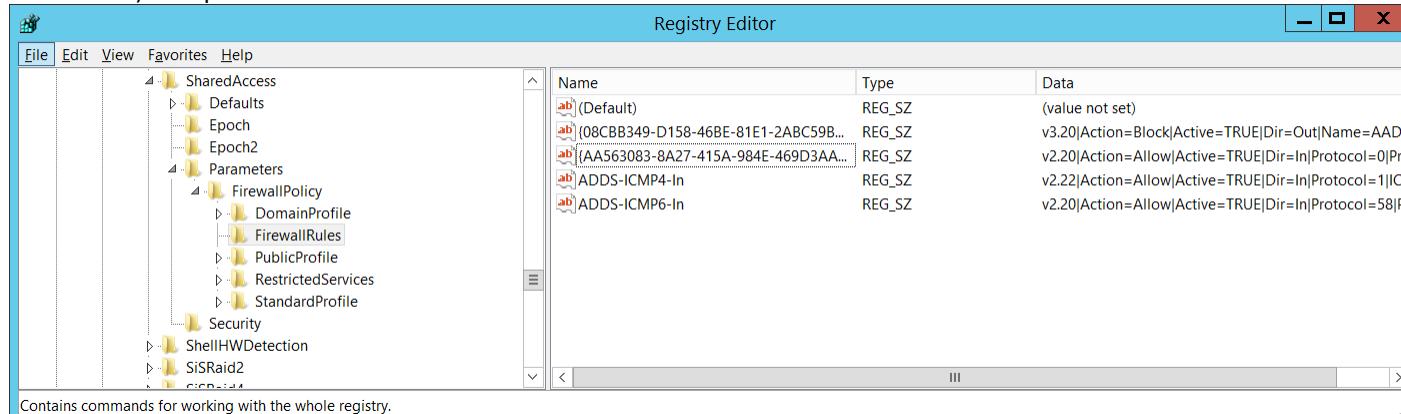
- Domain,Public
- Domain,Private
- Private,Public
- Public
- Domain
- Private

Ignored Rule:

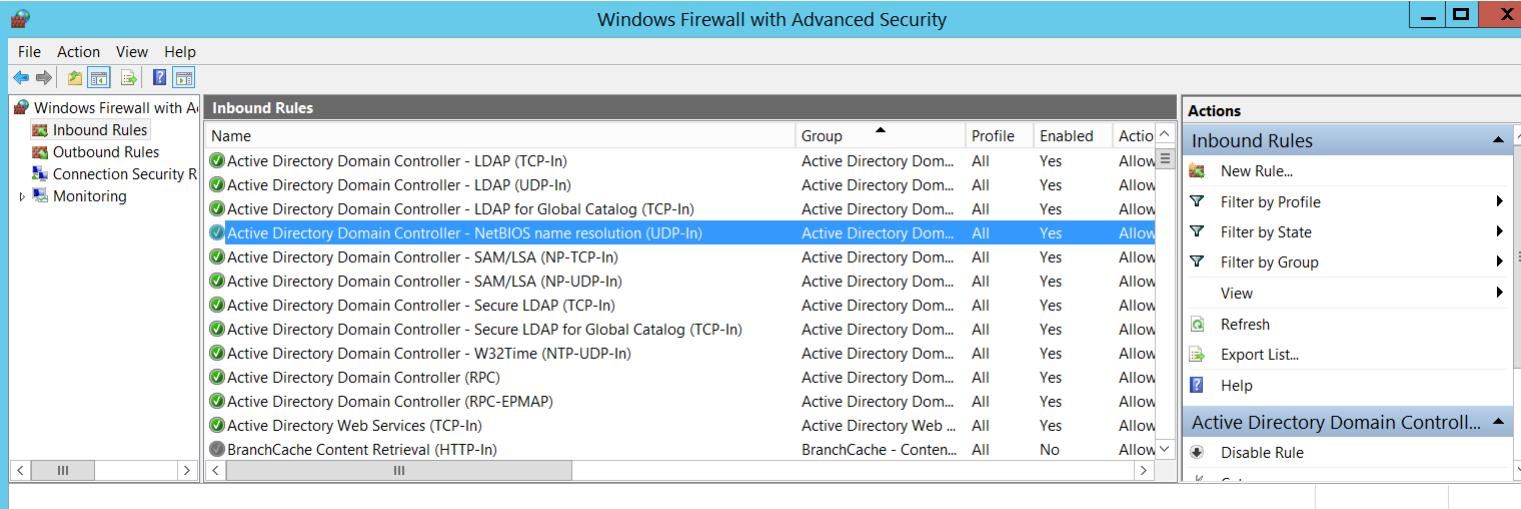
- **ID** [Type = UnicodeString]: the unique identifier for ignored firewall rule.

To see the unique ID of the rule you need to navigate to

"HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules" registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Name** [Type = UnicodeString]: the name of the rule which was ignored. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (**wf.msc**), check "Name" column:



The screenshot shows the Windows Firewall with Advanced Security interface. The left navigation pane includes options like Windows Firewall with Advanced Security, Inbound Rules, Outbound Rules, Connection Security Rules, and Monitoring. The main area displays a table titled 'Inbound Rules' with columns: Name, Group, Profile, Enabled, and Action. The 'Action' column shows 'Allow'. A context menu is open over the selected row ('Active Directory Domain Controller - NetBIOS name resolution (UDP-In)'). The menu is titled 'Actions' and lists: New Rule..., Filter by Profile, Filter by State, Filter by Group, View, Refresh, Export List..., Help, and a separator line followed by 'Disable Rule'. The 'Active Directory Domain Controller...' item is highlighted.

Name	Group	Profile	Enabled	Action
Active Directory Domain Controller - LDAP (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - LDAP (UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - LDAP for Global Catalog (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - NetBIOS name resolution (UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - SAM/LSA (NP-TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - SAM/LSA (NP-UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - Secure LDAP (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - Secure LDAP for Global Catalog (TCP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller - W32Time (NTP-UDP-In)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller (RPC)	Active Directory Dom...	All	Yes	Allow
Active Directory Domain Controller (RPC-EPMAP)	Active Directory Dom...	All	Yes	Allow
Active Directory Web Services (TCP-In)	Active Directory Web ...	All	Yes	Allow
BranchCache Content Retrieval (HTTP-In)	BranchCache - Conten...	All	No	Allow

Security Monitoring Recommendations:

For 4951(F): A rule has been ignored because its major version number was not recognized by Windows Firewall.

- This event can be a sign of software issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

4952(F): Parts of a rule have been ignored because its minor version number was not recognized by Windows Firewall. The other parts of the rule will be enforced.

When you create or edit a Windows Firewall rule, the settings that you can include depend upon the version of Windows you use when creating the rule. As new settings are added to later versions of Windows or to service packs for existing versions of Windows, the version number of the rules processing engine is updated, and that version number is stamped into rules that are created by using that version of Windows. For example, Windows Vista produces firewall rules that are stamped with version "v2.0". Future versions of Windows might use "v2.1", or "v3.0" to indicate, respectively, minor or major changes and additions.

If you create a firewall rule on a newer version of Windows that references firewall settings that are not available on earlier versions of Windows, and then try to deploy that rule to computers running the earlier version of Windows, the firewall engine produces this error to indicate that it cannot process the rule.

The only solution is to remove the incompatible rule, and then deploy a compatible rule.

There is no example of this event in this document.

Event Schema:

Parts of a rule have been ignored because its minor version number was not recognized by Windows Firewall. The other parts of the rule will be enforced.

%t

Profile:%t%1

Partially Ignored Rule:

%tID:%t%2

%tName:%t%3

Required Server Roles: None.

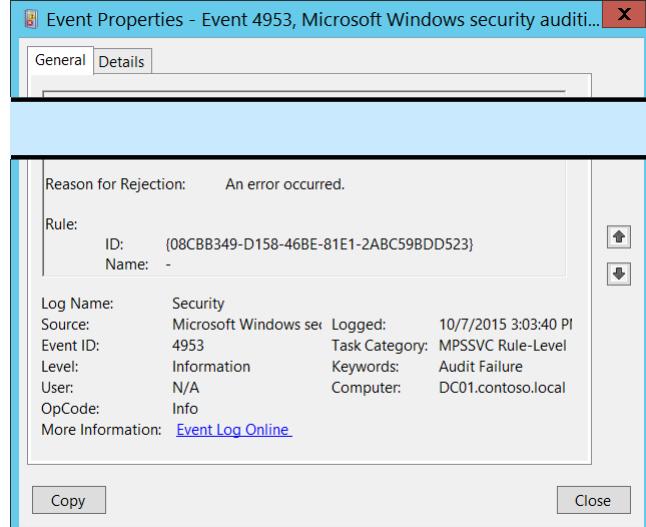
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

4953(F): Windows Firewall ignored a rule because it could not be parsed.



Event Properties - Event 4953, Microsoft Windows security audit...

General Details

Reason for Rejection: An error occurred.

Rule:
ID: {08CBB349-D158-46BE-81E1-2ABC59BDD523}
Name: -

Log Name: Security
Source: Microsoft Windows security
Event ID: 4953
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/7/2015 3:03:40 PM
Task Category: MPSSVC Rule-Level
Keywords: Audit Failure
Computer: DC01.contoso.local

Copy Close

Event Description:

This event generates if Windows Firewall was not able to parse Windows Firewall rule for some reason. It can happen if Windows Firewall rule registry entry was corrupted.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4953</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13571</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-07T22:03:40.261507200Z" />

```

```

<EventRecordID>1052340</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="5088" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="Profile">All</Data>
<Data Name="ReasonForRejection">An error occurred.</Data>
<Data Name="RuleId">{08CBB349-D158-46BE-81E1-2ABC59BDD523}</Data>

```

```
<Data Name="RuleName">-</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Profile [Type = UnicodeString]: the name of the profile of the ignored rule. Possible values are:

- All
- Domain,Public
- Domain,Private
- Private,Public
- Public
- Domain
- Private

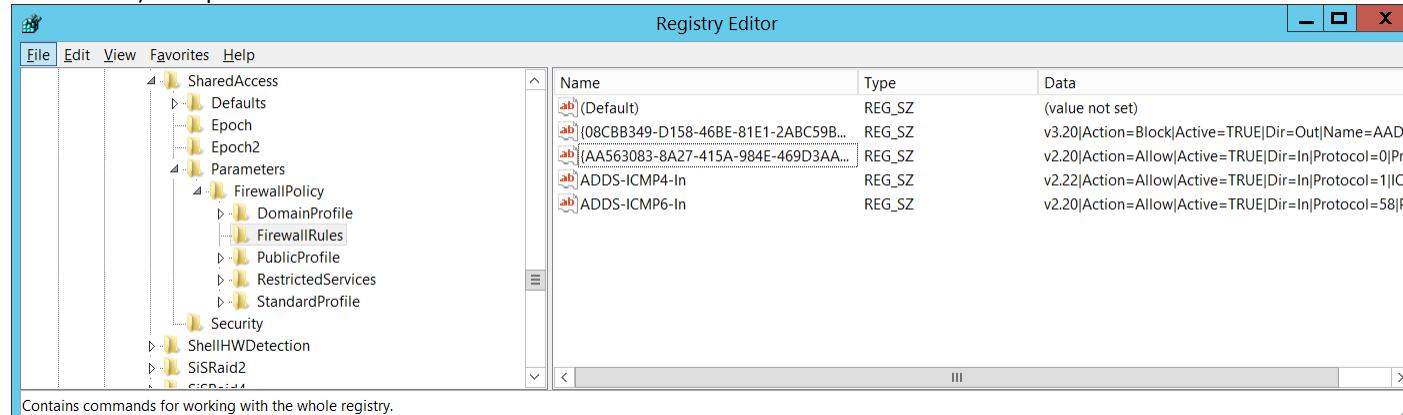
Reason for Rejection [Type = UnicodeString]: the reason, why the rule was ignored.

Rule:

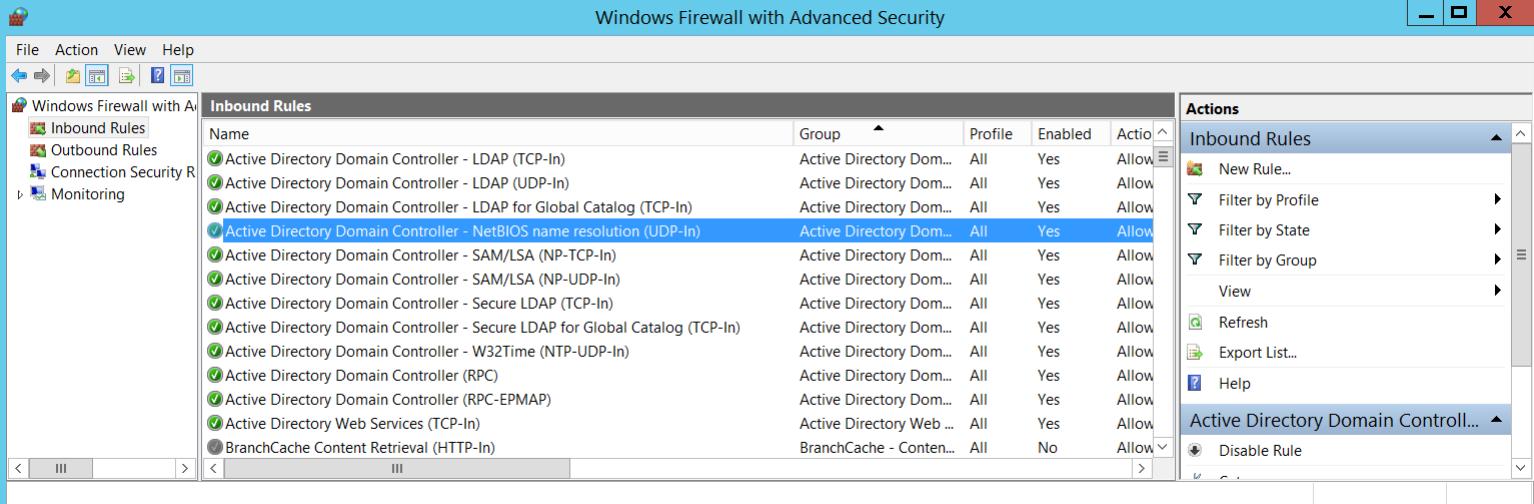
- **ID** [Type = UnicodeString]: the unique identifier for ignored firewall rule.

To see the unique ID of the rule you need to navigate to

"**HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules**" registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Name** [Type = UnicodeString]: the name of the rule which was ignored. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (**wf.msc**), check "Name" column:



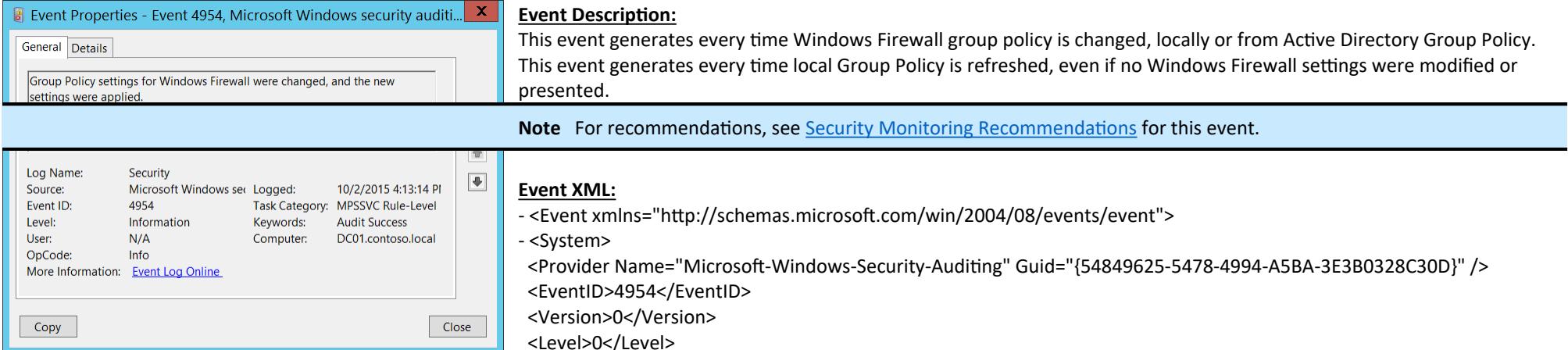
The screenshot shows the Windows Firewall with Advanced Security interface. The left navigation pane includes options like Windows Firewall with Advanced Security, Inbound Rules, Outbound Rules, Connection Security Rules, and Monitoring. The main area displays a table titled 'Inbound Rules' with columns for Name, Group, Profile, Enabled, and Action. The table lists various Active Directory-related rules, such as Active Directory Domain Controller - LDAP (TCP-In), Active Directory Domain Controller - LDAP (UDP-In), and Active Directory Domain Controller - Secure LDAP (TCP-In). The right side features a sidebar titled 'Actions' with options like New Rule..., Filter by Profile, Filter by State, Filter by Group, View, Refresh, Export List..., Help, and Disable Rule.

Security Monitoring Recommendations:

For 4953(F): Windows Firewall ignored a rule because it could not be parsed.

- This event can be a sign of software issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

4954(S): Windows Firewall Group Policy settings have changed. The new settings have been applied.



Event Description:
This event generates every time Windows Firewall group policy is changed, locally or from Active Directory Group Policy. This event generates every time local Group Policy is refreshed, even if no Windows Firewall settings were modified or presented.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name: Security	Source: Microsoft-Windows-security-audit	Logged: 10/2/2015 4:13:14 PM
Event ID: 4954	Task Category: MPSSVC Rule-Level	
Level: Information	Keywords: Audit Success	
User: N/A	Computer: DC01.contoso.local	
OpCode: Info		
More Information: Event Log Online		

Event XML:

```

<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
    <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
    <EventID>4954</EventID>
    <Version>0</Version>
    <Level>0</Level>
  </System>
  <Task>13571</Task>
  <Opcode>0</Opcode>

```

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-02T23:13:14.527924800Z" />
<EventRecordID>1049893</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2284" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 4954(S): Windows Firewall Group Policy settings have changed. The new settings have been applied.

- Unfortunately this event generates every time local Group Policy is refreshed and does not indicate that settings really were modified. Typically this event can be ignored.

4956(S): Windows Firewall has changed the active profile.

 Event Properties - Event 4956, Microsoft Windows security audit... X

<input checked="" type="checkbox"/> General	<input type="checkbox"/> Details
---	----------------------------------

Event Description:
This event generates when Windows Firewall has changed the active profile.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

<p>New Active Profile: Domain</p> <p>Log Name: Security Source: Microsoft Windows sec... Logged: 10/2/2015 5:14:56 PM Event ID: 4956 Task Category: MPSSVC Rule-Level Level: Information Keywords: Audit Success User: N/A Computer: DC01.contoso.local OpCode: Info More Information: Event Log Online</p> <p style="text-align: center;">Copy Close</p>	<p>Event XML:</p> <pre>- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>4956</EventID> <Version>0</Version> <Level>0</Level> <Task>13571</Task> <Opcode>0</Opcode></pre>
---	--

```
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-03T00:14:56.676017600Z" />
<EventRecordID>1050811</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2216" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="ActiveProfile">Domain</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

New Active Profile [Type = UnicodeString]: the name of the new active profile. Possible values are:

- Domain
- Public
- Private

Security Monitoring Recommendations:

For 4956(S): Windows Firewall has changed the active profile.

- Typically this event has an informational purpose.
- For domain joined machines you could monitor for all events where **New Active Profile** doesn't equal "Domain". This indicates that the computer was connected to another non-domain network.

Event Properties - Event 4957, Microsoft Windows security audit... X

General		Details	
ID:	CoreNet-Teredo-In	Name:	Core Networking - Teredo (UDP-In)
Error Information:			
Reason: Local Port resolved to an empty set.			
Log Name:	Security	Source:	Microsoft Windows sec
Event ID:	4957	Task Category:	MPSSVC Rule-Level
Level:	Information	Keywords:	Audit Failure
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info		
More Information: Event Log Online			
Copy		Close	

4957(F): Windows Firewall did not apply the following rule.

Event Description:

This event generates when Windows Firewall starts or applies new rule, and the rule cannot be applied for some reason.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4957</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13571</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8010000000000000</Keywords>
```

```

<TimeCreated SystemTime="2015-10-02T23:13:14.496678500Z" />
<EventRecordID>1049892</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2284" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="RuleId">CoreNet-Teredo-In</Data>
  <Data Name="RuleName">Core Networking - Teredo (UDP-In)</Data>
  <Data Name="RuleAttr">Local Port</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

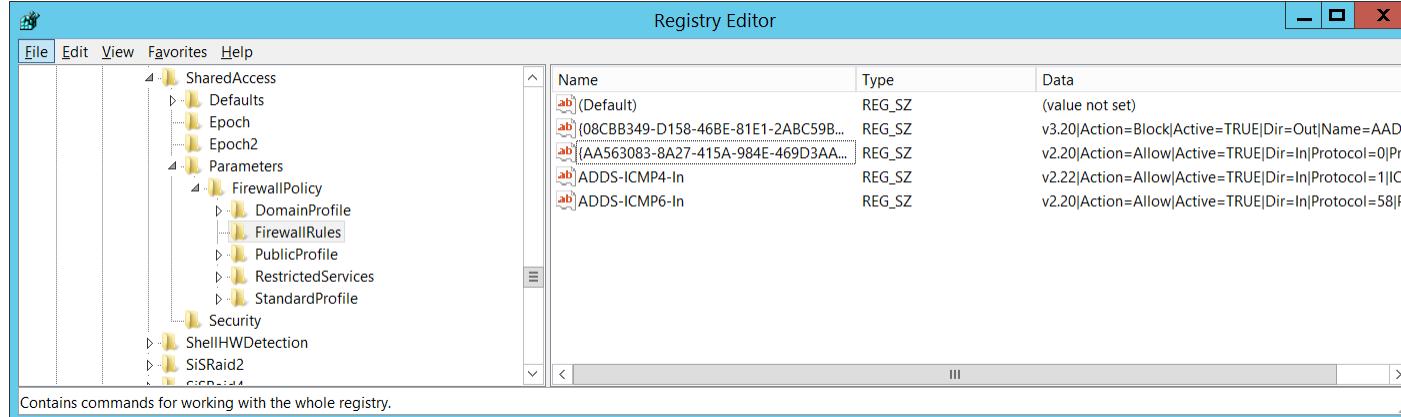
Field Descriptions:

Rule Information:

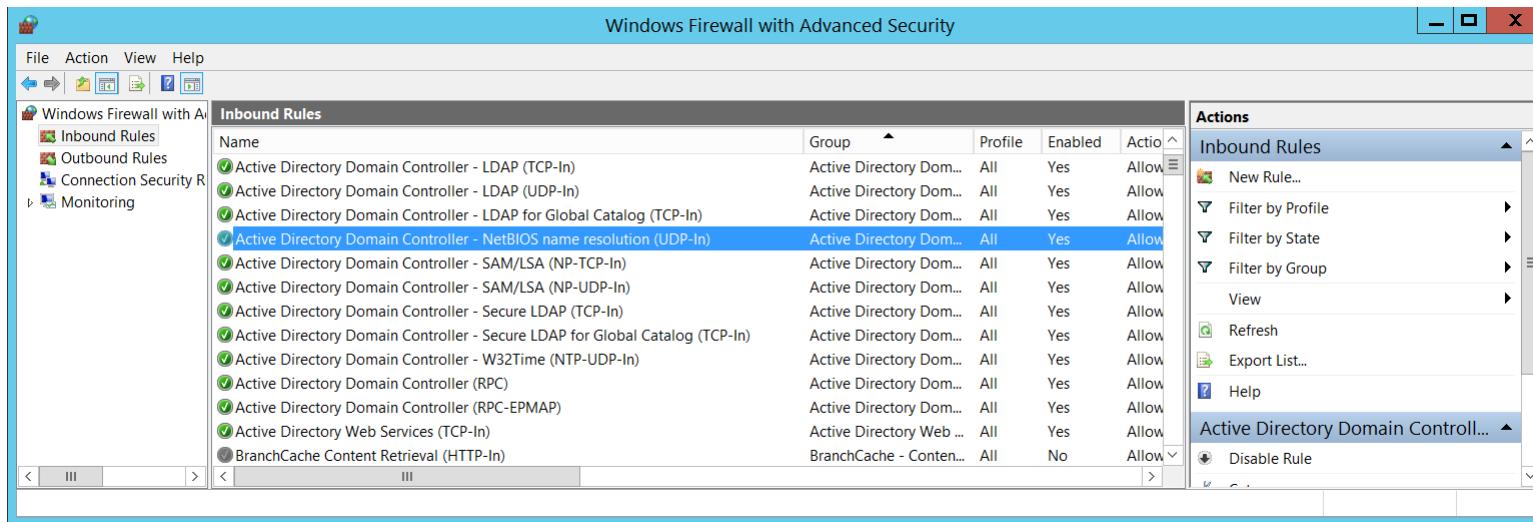
- **ID** [Type = UnicodeString]: the unique identifier for not applied firewall rule.

To see the unique ID of the rule you need to navigate to

“**HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicy\FirewallRules**” registry key and you will see the list of Windows Firewall rule IDs (Name column) with parameters:



- **Name** [Type = UnicodeString]: the name of the rule which was not applied. You can see the name of Windows Firewall rule using Windows Firewall with Advanced Security management console (**wf.msc**), check “Name” column:



Error Information:

- **Reason** [Type = UnicodeString]: the reason why the rule was not applied.

Security Monitoring Recommendations:

For 4957(F): Windows Firewall did not apply the following rule.

- This event can be a sign of software issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

4958(F): Windows Firewall did not apply the following rule because the rule referred to items not configured on this computer.

Windows Firewall with Advanced Security processed a rule that contains parameters that cannot be resolved on the local computer. The rule is therefore not enforceable on the computer and so is excluded from the runtime state of the firewall. This is not necessarily an error. Examine the rule for applicability on the computers to which it was applied.

There is no example of this event in this document.

Event Schema:

Windows Firewall did not apply the following rule because the rule referred to items not configured on this computer:

Rule Information:

%tID:%t%1

%tName:%t%2

Error Information:

%tError:%t%3

%tReason:%t%4

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

Audit Other Policy Change Events

Audit Other Policy Change Events contains events about EFS Data Recovery Agent policy changes, changes in Windows Filtering Platform filter, status on Security policy settings updates for local Group Policy settings, Central Access Policy changes, and detailed troubleshooting events for Cryptographic Next Generation (CNG) operations.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	IF	Yes	IF	Yes	<p>IF - We do not recommend Success auditing because of event "5447: A Windows Filtering Platform filter has been changed"—this event generates many times during group policy updates and typically is used for troubleshooting purposes for Windows Filtering Platform filters. But you would still need to enable Success auditing for this subcategory if, for example, you must monitor changes in Boot Configuration Data or Central Access Policies.</p> <p>We recommend Failure auditing, to detect errors in applied Security settings which came from Group Policy, and failure events related to Cryptographic Next Generation (CNG) functions.</p>
Member Server	IF	Yes	IF	Yes	<p>IF - We do not recommend Success auditing because of event "5447: A Windows Filtering Platform filter has been changed"—this event generates many times during group policy updates and typically is used for troubleshooting purposes for Windows Filtering Platform filters. But you would still need to enable Success auditing for this subcategory if, for example, you must monitor changes in Boot Configuration Data or Central Access Policies.</p> <p>We recommend Failure auditing, to detect errors in applied Security settings which came from Group Policy, and failure events related to Cryptographic Next Generation (CNG) functions.</p>
Workstation	IF	Yes	IF	Yes	<p>IF - We do not recommend Success auditing because of event "5447: A Windows Filtering Platform filter has been changed"—this event generates many times during group policy updates and typically is used for troubleshooting purposes for Windows Filtering Platform filters. But you would still need to enable Success auditing for this subcategory if, for example, you must monitor changes in Boot Configuration Data or Central Access Policies.</p> <p>We recommend Failure auditing, to detect errors in applied Security settings which came from Group Policy, and failure events related to Cryptographic Next Generation (CNG) functions.</p>

Events List:

- [4714\(S\)](#): Encrypted data recovery policy was changed.
- [4819\(S\)](#): Central Access Policies on the machine have been changed.
- [4826\(S\)](#): Boot Configuration Data loaded.
- [4909\(-\)](#): The local policy settings for the TBS were changed.
- [4910\(-\)](#): The group policy settings for the TBS were changed.
- [5063\(S, F\)](#): A cryptographic provider operation was attempted.
- [5064\(S, F\)](#): A cryptographic context operation was attempted.
- [5065\(S, F\)](#): A cryptographic context modification was attempted.

- [5066](#)(S, F): A cryptographic function operation was attempted.
- [5067](#)(S, F): A cryptographic function modification was attempted.
- [5068](#)(S, F): A cryptographic function provider operation was attempted.
- [5069](#)(S, F): A cryptographic function property operation was attempted.
- [5070](#)(S, F): A cryptographic function property modification was attempted.
- [5447](#)(S): A Windows Filtering Platform filter has been changed.
- [6144](#)(S): Security policy in the group policy objects has been applied successfully.
- [6145](#)(F): One or more errors occurred while processing security policy in the group policy objects.

4714(S): Encrypted data recovery policy was changed.

 Event Properties - Event 4714, Microsoft Windows security audit... X

General		Details
Data Recovery Agent group policy for Encrypting File System (EFS) has changed. The new changes have been applied.		

Copy Close

Event Description:

This event generates when a Data Recovery Agent group policy for Encrypting File System ([EFS](#)) has changed.

This event generates when a Data Recovery Agent certificate or [Data Recovery Agent policy](#) was changed for the computer or device.

In the background, this event generates when the <\\HKLM\\Software\\Policies\\Microsoft\\SystemCertificates\\EFS\\EfsBlob> registry value is changed during a Group Policy update.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-a5ba-3e3b0328c30d}" />
<EventID>4714</EventID>
```

```

<Version>0</Version>
<Level>0</Level>
<Task>13573</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-08T05:27:40.740602500Z" />
<EventRecordID>1080883</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="4856" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <ProcessingErrorData>
```

```
<ErrorCode>13</ErrorCode>
<DataItemName>SubjectUserId</DataItemName>
<EventPayload />
</ProcessingErrorData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

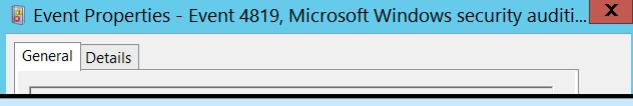
Event Versions: 0.

Security Monitoring Recommendations:

For 4714(S): Encrypted data recovery policy was changed.

- We recommend monitoring this event and if the change was not planned, investigate the reason for the change.

4819(S): Central Access Policies on the machine have been changed.

 Event Properties - Event 4819, Microsoft Windows security audit... X

General		Details
Security ID:	SYSTEM	
Account Name:	DC01\$	
Account Domain:	CONTOSO	
Logon ID:	0x3E7	
Object:	Object Server:	LSA
	Object Type:	Central Access Policies
CAPs Added:	Main Policy	
CAPs Deleted:		
CAPs Modified:		
CAPs As-Is:		
Log Name:	Security	
Source:	Microsoft Windows security	
Event ID:	4819	Logged: 11/9/2015 5:00:34 PM
Level:	Information	Task Category: Other Policy Change
User:	N/A	Keywords: Audit Success
OpCode:	Info	Computer: DC01.contoso.local
More Information: Event Log Online		

Copy Close

Event Description:
This event generates when [Central Access Policy](#) on the machine have been changed.
For example, it generates when a new [Central Access Policy](#) was applied to the machine via Group Policy.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  <EventID>4819</EventID>
  <Version>0</Version>
  <Level>0</Level>
  <Task>13573</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8020000000000000</Keywords>
  <TimeCreated SystemTime="2015-11-10T01:00:34.352877700Z" />
  <EventRecordID>1187659</EventRecordID>
  <Correlation />
  <Execution ProcessID="516" ThreadID="3500" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
</System>
```

- <EventData>

```

<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="ObjectServer">LSA</Data>
<Data Name="ObjectType">Central Access Policies</Data>
<Data Name="AddedCAPs">Main Policy</Data>
<Data Name="DeletedCAPs" />
<Data Name="ModifiedCAPs" />
<Data Name="AsIsCAPs" />
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that changed the Central Access Policies on the machine. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that changed the Central Access Policies on the machine.
- **Account Domain** [Type = UnicodeString]: domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Object:

- **Object Server** [Type = UnicodeString]: has “**LSA**” value for this event.
- **Object Type** [Type = UnicodeString]: The type of an object to which this event applies. Always “**Central Access Policies**” for this event.

The following table contains the list of the most common **Object Types**:

Directory	Event	Timer	Device
-----------	-------	-------	--------

Mutant	Type	File	Token
Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	Central Access Policies
Key	WaitablePort	Callback	
Job	Port	FilterConnectionPort	
ALPC Port	Semaphore	Adapter	

CAPs Added [Type = UnicodeString]: the list of added Central Access Policies. Empty if no Central Access Policies were added.

CAPs Deleted [Type = UnicodeString]: the list of deleted Central Access Policies. Empty if no Central Access Policies were deleted.

CAPs Modified [Type = UnicodeString]: the list of modified Central Access Policies. Empty if no Central Access Policies were modified.

CAPs As-Is [Type = UnicodeString]: the list of non-modified Central Access Policies.

Event Properties - Event 4826, Microsoft Windows security auditing.
X

General	
Boot Configuration Data	
Subject: Security ID: SYSTEM Account Name: - Account Domain: - Logon ID: 0x3E7	
General Settings: Load Options: - Advanced Options: No Default Configuration Access Policy: Default System Event Logging: No Kernel Debugging: No VSM Launch Type: Off	
Signature Settings: Test Signing: No Flight Signing: No Disable Integrity Checks: No	
HyperVisor Settings:	

Security Monitoring Recommendations:
For 4819(S): Central Access Policies on the machine have been changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- This event can help you to track modifications, additions and deletions of Central Access Policies if it is required by your security monitoring policy.
-

4826(S): Boot Configuration Data loaded.

Event Description:
This event generates every time system starts and load current [Boot Configuration Data](#) (BCD) settings.
This event is always logged regardless of the "Audit Other Policy Change Events" sub-category setting.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4826</EventID>
<Version>0</Version>
<Level>0</Level>
```

Log Name: Security
Source: Microsoft Windows security Logged: 11/12/2015 4:59:57 PM
Event ID: 4826 Task Category: Other Policy Change Events
Level: Information Keywords: Audit Success
User: N/A Computer: WIN10-1
OpCode: Info

More Information: [Event Log Online Help](#)

[Copy](#) [Close](#)

```
<Task>13573</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-13T00:59:57.553201100Z" />
<EventRecordID>751</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="164" />
<Channel>Security</Channel>
<Computer>WIN10-1</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-18</Data>
  <Data Name="SubjectUserName">-</Data>
  <Data Name="SubjectDomainName">-</Data>
  <Data Name="SubjectLogonId">0x3e7</Data>
  <Data Name="LoadOptions">-</Data>
  <Data Name="AdvancedOptions">%%1843</Data>
  <Data Name="ConfigAccessPolicy">%%1846</Data>
  <Data Name="RemoteEventLogging">%%1843</Data>
  <Data Name="KernelDebug">%%1843</Data>
  <Data Name="VsmLaunchType">%%1848</Data>
  <Data Name="TestSigning">%%1843</Data>
  <Data Name="FlightSigning">%%1843</Data>
  <Data Name="DisableIntegrityChecks">%%1843</Data>
  <Data Name="HypervisorLoadOptions">-</Data>
  <Data Name="HypervisorLaunchType">%%1848</Data>
  <Data Name="HypervisorDebug">%%1843</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2012, Windows 8.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that reported this event. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event. Always "S-1-5-18" for this event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that reported this event. Always “-” for this event.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Always “-” for this event.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

General Settings:

- **Load Options** [Type = UnicodeString]: there is no information about this field in this document.
- **Advanced Options** [Type = UnicodeString]: shows whether Windows is configured for system boot to the legacy menu (F8 menu) on the next boot (**Yes** or **No**). You can enable advanced boot using “bcdedit /set onetimeadvancedoptions yes” command.
- **Configuration Access Policy** [Type = UnicodeString]: there is no information about this field in this document.
- **System Event Logging** [Type = UnicodeString]: there is no information about this field in this document.
- **Kernel Debugging** [Type = UnicodeString]: shows whether Windows [kernel debugging](#) is enabled or not (**Yes** or **No**). You can enable kernel debugging using “bcdedit /debug on” command.
- **VSM Launch Type** [Type = UnicodeString]: there is no information about this field in this document.

Signature Settings:

- **Test Signing** [Type = UnicodeString]: shows whether Windows [test signing](#) is enabled or not (**Yes** or **No**). You can disable test signing using “bcdedit /set testsigning off” command.

This parameter controls whether Windows 8.1, Windows 8, Windows 7, Windows Server 2008, or Windows Vista will load any type of test-signed kernel-mode code. This option is not set by default, which means test-signed kernel-mode drivers on 64-bit versions of Windows 8.1, Windows 8, Windows 7, Windows Server 2008, and Windows Vista will not load by default. After you run the BCDEdit command, restart the computer so that the change takes effect. For more information, see [Introduction to Test-Signing](#).
- **Flight Signing** [Type = UnicodeString]: shows whether Windows flight signing (which allows flight-signed code signing certificates) is enabled or not (**Yes** or **No**). You can disable flight signing using “bcdedit /set flightsigning off” command.
- **Disable Integrity Checks** [Type = UnicodeString]: shows whether Windows integrity check is disabled or not (**Yes** or **No**). You can disable integrity checks using “bcdedit /set nointegritychecks on” command.

HyperVisor Settings:

- **HyperVisor Load Options** [Type = UnicodeString]: shows hypervisor **loadoptions**. See more information here:
[https://msdn.microsoft.com/en-us/library/windows/hardware/ff542202\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/ff542202(v=vs.85).aspx).
- **HyperVisor Launch Type** [Type = UnicodeString]: shows the hypervisor launch options (**Off** or **Auto**). If you are setting up a debugger to debug Hyper-V on a target computer, set this option to **Auto** on the target computer. For more information, see [Attaching to a Target Computer Running Hyper-V](#). Information about [Hyper-V](#) technology is available on Microsoft TechNet web site.
- **HyperVisor Debugging** [Type = UnicodeString]: shows whether the hypervisor debugger is enabled or not (**Yes** or **No**). For information about hypervisor debugging, see [Attaching to a Target Computer Running Hyper-V](#).

Security Monitoring Recommendations:

For 4826(S): Boot Configuration Data loaded.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- If you have a standard or baseline for Boot Configuration Data settings defined, monitor this event and check whether the settings reported by the event are still the same as were defined in your standard or baseline.

4909(-): The local policy settings for the TBS were changed.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system.

4910(-): The group policy settings for the TBS were changed.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system.

5063(**S, F**): A cryptographic provider operation was attempted.

This event generates in BCryptUnregisterProvider() and BCryptRegisterProvider() functions. These are Cryptographic Next Generation (CNG) functions.

This event generates when cryptographic provider was registered or unregistered.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic provider operation was attempted.

Subject:

*Security ID:%1
 Account Name:%2
 Account Domain:%3
 Logon ID:%4*

Cryptographic Provider:

*Name:%5
 Module:%6
 Operation:%7*

Return Code:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5064(**S, F**): A cryptographic context operation was attempted.

This event generates in [BCryptCreateContext\(\)](#) and [BCryptDeleteContext\(\)](#) functions. These are Cryptographic Next Generation (CNG) functions.

This event generates when cryptographic context was created or deleted.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic context operation was attempted.

Subject:

*Security ID:%1
 Account Name:%2
 Account Domain:%3
 Logon ID:%4*

Configuration Parameters:

*Scope:%5
 Context:%6*

Operation:%7

Return Code:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5065(S, F): A cryptographic context modification was attempted.

This event generates in [BCryptConfigureContext\(\)](#) function. This is a Cryptographic Next Generation (CNG) function.

This event generates when configuration information was changed for existing CNG context.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic context modification was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6*

Change Information:

*Old Value:%7
New Value:%8*

Return Code:%9

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5066(S, F): A cryptographic function operation was attempted.

This event generates in [BCryptAddContextFunction\(\)](#) and [BCryptRemoveContextFunction\(\)](#) functions. These are Cryptographic Next Generation (CNG) functions.

This event generates when cryptographic function was added or removed from the list of functions that are supported by an existing CNG context.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic function operation was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6
Interface:%7
Function:%8
Position:%9*

Operation:%10

Return Code:%11

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5067(S, F): A cryptographic function modification was attempted.

This event generates in [BCryptConfigureContextFunction\(\)](#) function. This is a Cryptographic Next Generation (CNG) function.

This event generates when configuration information for the cryptographic function of an existing CNG context was changed.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic function modification was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6
Interface:%7
Function:%8*

Change Information:

*Old Value:%9
New Value:%10*

Return Code:%11

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5068(S, F): A cryptographic function provider operation was attempted.

This event generates in BCryptAddContextFunctionProvider() and BCryptRemoveContextFunctionProvider() functions. These are Cryptographic Next Generation (CNG) functions.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic function provider operation was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6
Interface:%7
Function:%8
Provider:%9
Position:%10*

Operation:%11

Return Code:%12

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5069(S, F): A cryptographic function property operation was attempted.

This event generates in [BCryptSetContextFunctionProperty\(\)](#) function. This is a Cryptographic Next Generation (CNG) function.

This event generates when named property for a cryptographic function in an existing CNG context was added or removed.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic function property operation was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6
Interface:%7
Function:%8
Property:%9*

Operation:%10

Value:%11

Return Code:%12

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

5070(S, F): A cryptographic function property modification was attempted.

This event generates in [BCryptSetContextFunctionProperty\(\)](#) function. This is a Cryptographic Next Generation (CNG) function.

This event generates when named property for a cryptographic function in an existing CNG context was updated.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic function property modification was attempted.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Configuration Parameters:

*Scope:%5
Context:%6
Interface:%7
Function:%8
Property:%9*

Change Information:

*Old Value:%10
New Value:%11*

Return Code:%12

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related cryptographic functions. If you need to monitor or troubleshoot actions related to specific cryptographic functions, review this event to see if it provides the information you need.

Event Properties - Event 5447, Microsoft Windows security auditing.

General Details

A Windows Filtering Platform filter has been changed.
Subject:

Process Information:
Process ID: 284

Provider Information:
ID: {decc16ca-3f33-4346-be1e-8fb4ae0f3d62}
Name: Microsoft Corporation

Change Information:
Change Type: Delete

Filter Information:
ID: {91334e6d-ffab-40f1-8c43-5554965c228d}
Name: Port Scanning Prevention Filter
Type: Not persistent
Run-Time ID: 100100

Layer Information:
ID: {ac4a9833-f69d-4648-b261-6dc84835ef39}
Name: Inbound Transport v4 Discard Layer
Run-Time ID: 13

Callout Information:
ID: {eda08606-2494-4d78-89bc-67837c03b969}
Name: WFP Built-in Silent Drop Transport v4 Discard Layer Callout

Additional Information:
Weight: 13835058055315718144
Conditions:
Condition ID: {632ce23b-5167-435c-86d7-e903684aa80c}
Match value: No flags set
Condition value: 0x00000003

Filter Action: Callout

Log Name: Security
Source: Microsoft Windows security
Event ID: 5447
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/7/2015 4:51:12 PM
Task Category: Other Policy Change Events
Keywords: Audit Success
Computer: DC01.contoso.local

5447(S): A Windows Filtering Platform filter has been changed.

Event Description:
This event generates every time a [Windows Filtering Platform](#) filter has been changed. It typically generates during Group Policy update procedures.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5447</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13573</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-07T23:51:12.191198900Z" />
<EventRecordID>1060216</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="3784" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ProcessId">284</Data>
<Data Name="UserId">S-1-5-19</Data>
<Data Name="UserName">NT AUTHORITY\LOCAL SERVICE</Data>
<Data Name="ProviderKey">{DECC16CA-3F33-4346-BE1E-8FB4AE0F3D62}</Data>
<Data Name="ProviderName">Microsoft Corporation</Data>
<Data Name="ChangeType">%16385</Data>
<Data Name="FilterKey">{91334E6D-FFAB-40F1-8C43-5554965C228D}</Data>
<Data Name="FilterName">Port Scanning Prevention Filter</Data>

```

Copy **Close**

```

<Data Name="FilterType">%&16388</Data>
<Data Name="FilterId">100100</Data>
<Data Name="LayerKey">{AC4A9833-F69D-4648-B261-6DC84835EF39}</Data>
<Data Name="LayerName">Inbound Transport v4 Discard Layer</Data>
<Data Name="LayerId">13</Data>
<Data Name="Weight">13835058055315718144</Data>
<Data Name="Conditions">Condition ID: {632ce23b-5167-435c-86d7-e903684aa80c} Match value: No flags set Condition value: 0x00000003</Data>
<Data Name="Action">%&16391</Data>
<Data Name="CalloutKey">{EDA08606-2494-4D78-89BC-67837C03B969}</Data>
<Data Name="CalloutName">WFP Built-in Silent Drop Transport v4 Discard Layer Callout</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

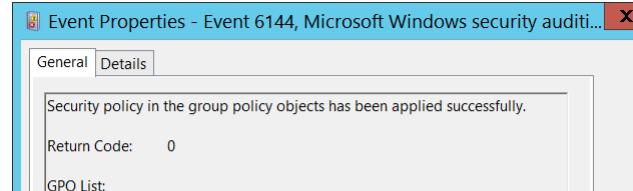
Event Versions: 0.

Security Monitoring Recommendations:

For 5447(S): A Windows Filtering Platform filter has been changed.

- This event mainly used for Windows Filtering Platform troubleshooting and typically has little to no security relevance.

6144(S): Security policy in the group policy objects has been applied successfully.



Event Description:

This event generates every time settings from the “Security Settings” section in the group policy object are applied successfully to a computer, without any errors. This event generates on the target computer itself. It is a routine event which shows you the list of Group Policy Objects that include “Security Settings” policies, and that were applied to the computer. This event generates every time Group Policy is applied to the computer.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6144</EventID>
<Version>0</Version>
<Level>0</Level>

```

<Task>13573</Task>
<Opcode>0</Opcode>

```

<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-07T22:59:32.280498500Z" />
<EventRecordID>1055041</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="712" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ErrorCode">0</Data>
<Data Name="GPOList">{8AB9311A-E5FB-4A5A-8FB7-027D1B877D6D} DC Main Policy</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

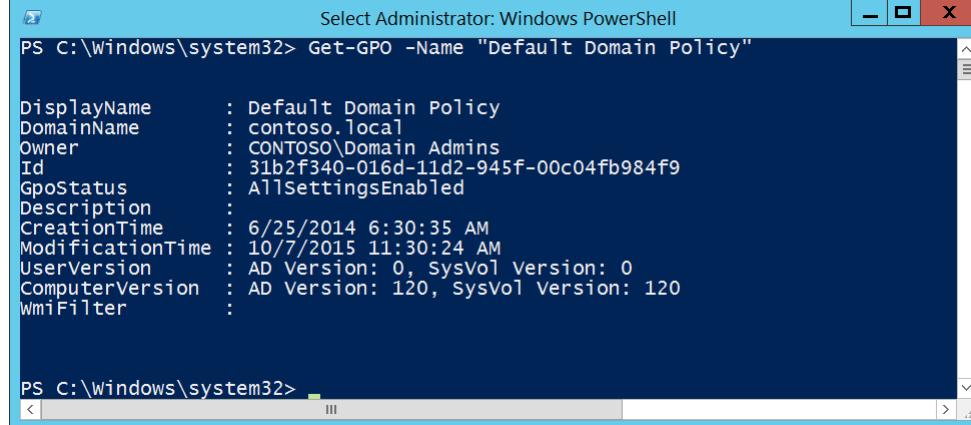
Event Versions: 0.

Field Descriptions:

Return Code [Type = UInt32]: always has “0” value for this event.

GPO List [Type = UnicodeString]: the list of Group Policy Objects that include “Security Settings” policies, and that were applied to the computer. The format of the list item is:
“GROUP_POLICY_GUID GROUP_POLICY_NAME”.

You can find specific GROUP_POLICY_GUID using **Get-GPO** PowerShell cmdlet with “–Name GROUP_POLICY_NAME” parameter. Row “Id” is the GUID of the Group Policy:



Property	Value
DisplayName	: Default Domain Policy
DomainName	: contoso.local
Owner	: CONTOSO\Domain Admins
Id	: 31b2f340-016d-11d2-945f-00c04fb984f9
GpoStatus	: AllSettingsEnabled
Description	:
CreationTime	: 6/25/2014 6:30:35 AM
ModificationTime	: 10/7/2015 11:30:24 AM
UserVersion	: AD Version: 0, SysVol Version: 0
ComputerVersion	: AD Version: 120, SysVol Version: 120
WmiFilter	:

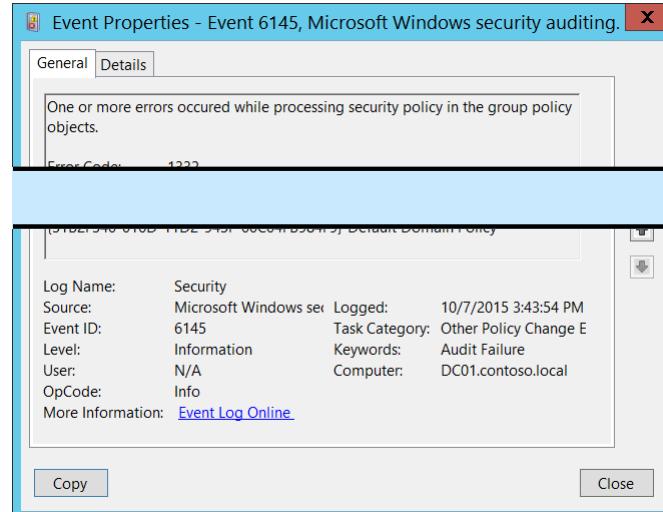
GUID is an acronym for ‘Globally Unique Identifier’. It is a 128-bit integer number used to identify resources, activities or instances.

Security Monitoring Recommendations:

For 6144(S): Security policy in the group policy objects has been applied successfully.

- If you have a pre-defined list of Group Policy Objects which contain Security Settings and must be applied to specific computers, then you can compare the list from this event with your list and in case of any difference trigger an alert.
- This event is mostly an informational event.

6145(F): One or more errors occurred while processing security policy in the group policy objects.



Event Description:

This event generates every time settings from the “Security Settings” section in the group policy object are applied to a computer with one or more errors. This event generates on the target computer itself.

This event generates, for example, if the [SID](#) of a security principal which was included in one of the Group Policy settings cannot be resolved or translated to the real account name.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>6145</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13573</Task>
```

```
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-07T22:43:54.183603800Z" />
<EventRecordID>1052680</EventRecordID>
<Correlation />
<Execution ProcessID="524" ThreadID="3476" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ErrorCode">1332</Data>
<Data Name="GPOList">{6AC1786C-016F-11D2-945F-00C04fB984F9} Default Domain Controllers Policy {31B2F340-016D-11D2-945F-00C04FB984F9} Default Domain Policy</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

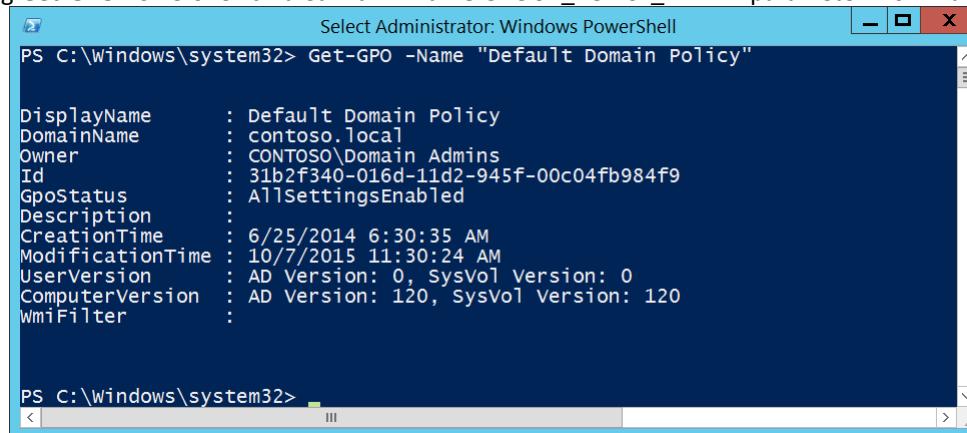
Event Versions: 0.

Field Descriptions:

Error Code [Type = UInt32]: specific error code which shows the error which happened during Group Policy processing. You can find the meaning of specific error code here: [https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ms681381(v=vs.85).aspx). For example, error code 1332 means that “no mapping between account names and security IDs was done”.

GPO List [Type = UnicodeString]: the list of Group Policy Objects that include “Security Settings” policies, and that were applied with errors to the computer. The format of the list item is: “GROUP_POLICY_GUID GROUP_POLICY_NAME”.

You can find specific GROUP_POLICY_GUID using **Get-GPO** PowerShell cmdlet with “**-Name GROUP_POLICY_NAME**” parameter. Row “Id” is the GUID of the Group Policy:



```
PS C:\Windows\system32> Get-GPO -Name "Default Domain Policy"

DisplayName      : Default Domain Policy
DomainName       : contoso.local
Owner            : CONTOSO\Domain Admins
Id               : 31b2f340-016d-11d2-945f-00c04fb984f9
GpoStatus        : AllSettingsEnabled
Description       :
CreationTime     : 6/25/2014 6:30:35 AM
ModificationTime  : 10/7/2015 11:30:24 AM
UserVersion      : AD Version: 0, SysVol Version: 0
ComputerVersion   : AD Version: 120, SysVol Version: 120
WmiFilter        :
```

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

Security Monitoring Recommendations:

For 6145(F): One or more errors occurred while processing security policy in the group policy objects.

- This event indicates that Group Policy Objects which were applied to the computer or device had some errors during processing. If you see this event, we recommend checking settings in the GPOs from **GPO List** and resolving the cause of the errors.
- If you have a pre-defined list of Group Policy Objects that contain Security Settings and that must be applied to specific computers, check this event to see if errors occurred when the Security Settings were applied. If so, you can review the error codes and investigate the cause of the failure.
- Typically this event has an informational purpose and the reason is configuration errors in Group Policy's security settings.
- This event might be used for Group Policy troubleshooting purposes.

Privilege Use

Audit Non Sensitive Privilege Use

Audit Non Sensitive Privilege Use contains events that show usage of non-sensitive privileges. This is the list of non-sensitive privileges:

- Access Credential Manager as a trusted caller
- Add workstations to domain
- Adjust memory quotas for a process
- Bypass traverse checking
- Change the system time
- Change the time zone
- Create a page file
- Create global objects
- Create permanent shared objects
- Create symbolic links
- Force shutdown from a remote system
- Increase a process working set
- Increase scheduling priority
- Lock pages in memory
- Modify an object label
- Perform volume maintenance tasks
- Profile single process
- Profile system performance
- Remove computer from docking station
- Shut down the system
- Synchronize directory service data

This subcategory also contains informational events from filesystem Transaction Manager.

If you configure this policy setting, an audit event is generated when a non-sensitive privilege is called. Success audits record successful attempts, and failure audits record unsuccessful attempts.

Event volume: Very High.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	IF	No	IF	We do not recommend Success auditing because the volume of events is very high and typically they are not as important as events from Audit Sensitive Privilege Use subcategory. IF – You can enable Failure auditing if you need information about failed attempts to use non-sensitive privileges, for example, <code>SeShutdownPrivilege</code> or <code>SeRemoteShutdownPrivilege</code> .
Member Server	No	IF	No	IF	We do not recommend Success auditing because the volume of events is very high and typically they are not as

					important as events from Audit Sensitive Privilege Use subcategory. IF – You can enable Failure auditing if you need information about failed attempts to use non-sensitive privileges, for example, SeShutdownPrivilege or SeRemoteShutdownPrivilege .
Workstation	No	IF	No	IF	We do not recommend Success auditing because the volume of events is very high and typically they are not as important as events from Audit Sensitive Privilege Use subcategory. IF – You can enable Failure auditing if you need information about failed attempts to use non-sensitive privileges, for example, SeShutdownPrivilege or SeRemoteShutdownPrivilege .

Events List:

- [4673](#)(S, F): A privileged service was called.
- [4674](#)(S, F): An operation was attempted on a privileged object.
- [4985](#)(S): The state of a transaction has changed.

4673(S, F): A privileged service was called.

This event also belongs in the **Audit Sensitive Privilege Use** subcategory, and is described there. See "[4673](#)(S, F): A privileged service was called."

4674(S, F): An operation was attempted on a privileged object.

This event also belongs in the **Audit Sensitive Privilege Use** subcategory, and is described there. See "[4674](#)(S, F): An operation was attempted on a privileged object."

4985(S): The state of a transaction has changed.

This event is also generated in the **Audit File System** subcategory, and is described there. See "[4985](#)(S): The state of a transaction has changed."

Audit Other Privilege Use Events

This auditing subcategory should not have any events in it, but for some reason Success auditing will enable generation of event 4985(S): The state of a transaction has changed.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	No	No	No	No	This auditing subcategory doesn't have any informative events inside.
Member Server	No	No	No	No	This auditing subcategory doesn't have any informative events inside.
Workstation	No	No	No	No	This auditing subcategory doesn't have any informative events inside.

Events List:

- 4985(S): The state of a transaction has changed.

4985(S): The state of a transaction has changed.

This event is also generated in the **Audit File System** subcategory, and is described there. See "[4985\(S\): The state of a transaction has changed](#)."

Audit Sensitive Privilege Use

Audit Sensitive Privilege Use contains events that show the usage of sensitive privileges. This is the list of sensitive privileges:

- Act as part of the operating system
- Back up files and directories
- Restore files and directories
- Create a token object
- Debug programs
- Enable computer and user accounts to be trusted for delegation
- Generate security audits
- Impersonate a client after authentication
- Load and unload device drivers
- Manage auditing and security log
- Modify firmware environment values
- Replace a process-level token
- Take ownership of files or other objects

The use of two privileges, “Back up files and directories” and “Restore files and directories,” generate events only if the “[Audit: Audit the use of Backup and Restore privilege](#)” Group Policy setting is enabled on the computer or device. We do not recommend enabling this Group Policy setting because of the high number of events recorded.

This subcategory also contains informational events from the file system Transaction Manager.

If you configure this policy setting, an audit event is generated when sensitive privilege requests are made. Success audits record successful attempts, and failure audits record unsuccessful attempts.

Event volume: High.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	We recommend tracking Success and Failure for this subcategory of events, especially if the sensitive privileges were used by a user account.
Member Server	Yes	Yes	Yes	Yes	We recommend tracking Success and Failure for this subcategory of events, especially if the sensitive privileges were used by a user account.
Workstation	Yes	Yes	Yes	Yes	We recommend tracking Success and Failure for this subcategory of events, especially if the sensitive privileges were used by a user account.

Events List:

- [4673](#)(S, F): A privileged service was called.
- [4674](#)(S, F): An operation was attempted on a privileged object.
- [4985](#)(S): The state of a transaction has changed.

[4673](#)(S, F): A privileged service was called.

Event Properties - Event 4673, Microsoft Windows security audit... X

General	Details
A privileged service was called.	
Subject:	Security ID: S-1-5-18
Logon ID: 0x3e7	
Service: Server: NT Local Security Authority / Authentication Service Service Name: LsaRegisterLogonProcess()	
Process: Process ID: 0x1f0 Process Name: C:\Windows\System32\lsass.exe	
Service Request Information: Privileges: SeTcbPrivilege	
Log Name: Security Source: Microsoft Windows security Event ID: 4673 Level: Information User: N/A OpCode: Info More Information: Event Log Online	
<input type="button" value="Copy"/> <input type="button" value="Close"/>	

Event Description:

This event generates when an attempt was made to perform privileged system service operations.

This event generates, for example, when **SeSystemtimePrivilege**, **SeCreateGlobalPrivilege**, or **SeTcbPrivilege** privilege was used.

Failure event generates when service call attempt fails.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4673</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13056</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T00:37:36.434836600Z" />
<EventRecordID>1099777</EventRecordID>
<Correlation />
<Execution ProcessID="496" ThreadID="504" />
<Channel>Security</Channel>

```

```

<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="ObjectServer">NT Local Security Authority / Authentication Service</Data>
<Data Name="Service">LsaRegisterLogonProcess()</Data>
<Data Name="PrivilegeList">SeTcbPrivilege</Data>
<Data Name="ProcessId">0x1f0</Data>
<Data Name="ProcessName">C:\Windows\System32\lsass.exe</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested privileged operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

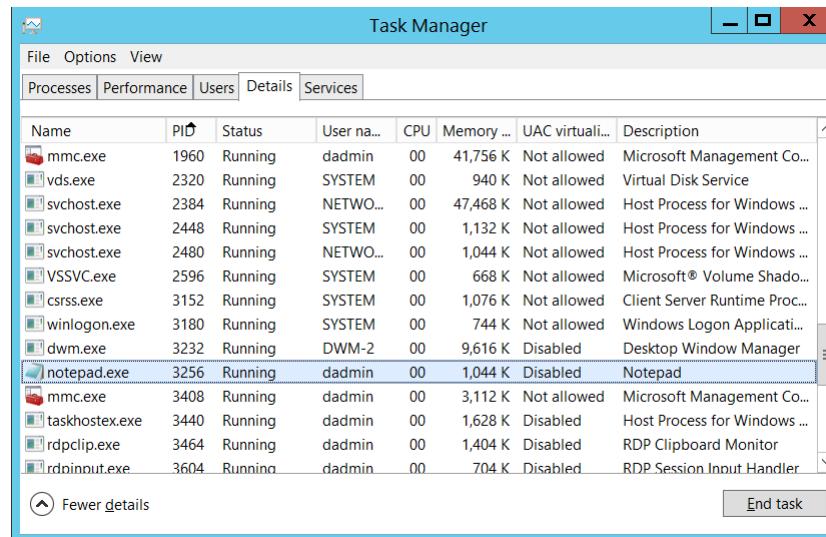
- **Account Name** [Type = UnicodeString]: the name of the account that requested privileged operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Service:

- **Server** [Type = UnicodeString]: contains the name of the Windows subsystem calling the routine. Subsystems examples are:
 - Security
 - Security Account Manager
 - NT Local Security Authority / Authentication Service
 - SC Manager
 - Win32 SystemShutdown module
 - LSA
- **Service Name** [Type = UnicodeString] [Optional]: supplies a name of the privileged subsystem service or function. For example, "RESET RUNTIME LOCAL SECURITY" might be specified by a **Local Security Authority** service used to update the local security policy database or **LsaRegisterLogonProcess()** might be specified by a **NT Local Security Authority / Authentication Service** used to register new logon process.

Process:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted to call the privileged service. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Service Request Information:

- **Privileges** [Type = UnicodeString]: the list of user privileges which were requested. The possible privileges depend on the subcategory, either **Audit Non Sensitive Privilege Use** or **Audit Sensitive Privilege Use**, as shown in the following two tables:

Subcategory of event	Privilege Name: User Right Group Policy Name	Description
Audit Non Sensitive Privilege Use	SeChangeNotifyPrivilege: Bypass traverse checking	Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks. With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.
Audit Non Sensitive Privilege Use	SeCreateGlobalPrivilege: Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
Audit Non Sensitive Privilege Use	SeCreatePagefilePrivilege: Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
Audit Non Sensitive Privilege Use	SeCreatePermanentPrivilege: Create permanent shared objects	Required to create a permanent object. This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.

Audit Non Sensitive Privilege Use	SeCreateSymbolicLinkPrivilege: Create symbolic links	Required to create a symbolic link.
Audit Non Sensitive Privilege Use	SeIncreaseBasePriorityPrivilege: Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
Audit Non Sensitive Privilege Use	SeIncreaseQuotaPrivilege: Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
Audit Non Sensitive Privilege Use	SeIncreaseWorkingSetPrivilege: Increase a process working set	Required to allocate more memory for applications that run in the context of users.
Audit Non Sensitive Privilege Use	SeLockMemoryPrivilege: Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
Audit Non Sensitive Privilege Use	SeMachineAccountPrivilege: Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
Audit Non Sensitive Privilege Use	SeManageVolumePrivilege: Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.
Audit Non Sensitive Privilege Use	SeProfileSingleProcessPrivilege: Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
Audit Non Sensitive Privilege Use	SeRelabelPrivilege: Modify an object label	Required to modify the mandatory integrity level of an object.
Audit Non Sensitive Privilege Use	SeRemoteShutdownPrivilege: Force shutdown from a remote system	Required to shut down a system using a network request.
Audit Non Sensitive Privilege Use	SeShutdownPrivilege: Shut down the system	Required to shut down a local system.
Audit Non Sensitive Privilege Use	SeSyncAgentPrivilege: Synchronize directory service data	This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers. With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.
Audit Non Sensitive Privilege Use	SeSystemProfilePrivilege: Profile system performance	Required to gather profiling information for the entire system. With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.

Audit Non Sensitive Privilege Use	SeSystemtimePrivilege: Change the system time	Required to modify the system time. With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.
Audit Non Sensitive Privilege Use	SeTimeZonePrivilege: Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
Audit Non Sensitive Privilege Use	SeTrustedCredManAccessPrivilege: Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
Audit Non Sensitive Privilege Use	SeUndockPrivilege: Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.

Subcategory of event	Privilege Name: User Right Group Policy Name	Description
Audit Sensitive Privilege Use	SeAssignPrimaryTokenPrivilege: Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
Audit Sensitive Privilege Use	SeAuditPrivilege: Generate security audits	With this privilege, the user can add entries to the security log.
Audit Sensitive Privilege Use	SeCreateTokenPrivilege: Create a token object	Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs. When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.
Audit Sensitive Privilege Use	SeDebugPrivilege: Debug programs	Required to debug and adjust the memory of a process owned by another account. With this privilege, the user can attach a debugger to any process or to the kernel. Developers who are debugging their own applications do not need this user right. Developers who are debugging new system components need this user right. This user right provides complete access to sensitive and critical operating system components.
Audit Sensitive Privilege Use	SeImpersonatePrivilege: Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
Audit Sensitive Privilege Use	SeLoadDriverPrivilege: Load and unload device drivers	Required to load or unload a device driver. With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.
Audit Sensitive Privilege Use	SeLockMemoryPrivilege: Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system

		performance by decreasing the amount of available random access memory (RAM).
Audit Sensitive Privilege Use	SeSystemEnvironmentPrivilege: Modify firmware environment values	Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.
Audit Sensitive Privilege Use	SeTcbPrivilege: Act as part of the operating system	This privilege identifies its holder as part of the trusted computer base. This user right allows a process to impersonate any user without authentication. The process can therefore gain access to the same local resources as that user.
Audit Sensitive Privilege Use	SeEnableDelegationPrivilege: Enable computer and user accounts to be trusted for delegation	Required to mark user and computer accounts as trusted for delegation. With this privilege, the user can set the Trusted for Delegation setting on a user or computer object. The user or object that is granted this privilege must have write access to the account control flags on the user or computer object. A server process running on a computer (or under a user context) that is trusted for delegation can access resources on another computer using the delegated credentials of a client, as long as the account of the client does not have the Account cannot be delegated account control flag set.

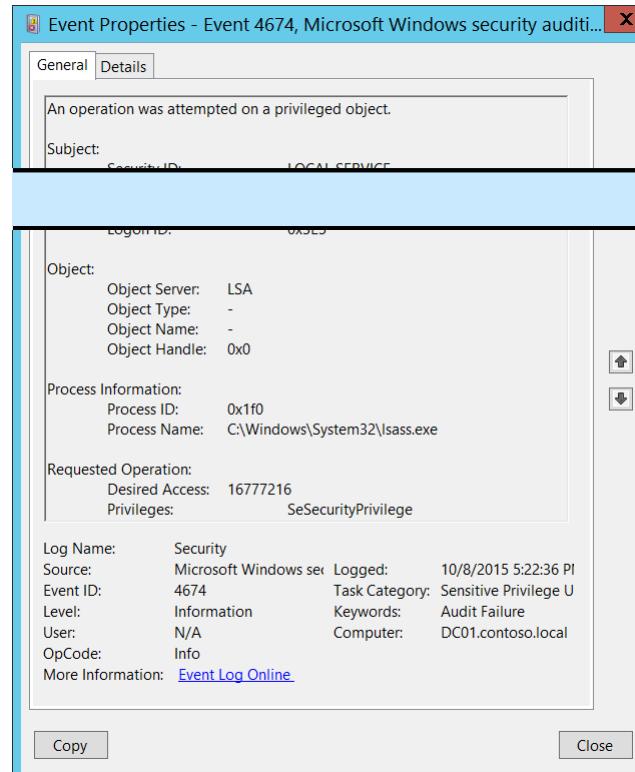
Security Monitoring Recommendations:

For 4673(S, F): A privileged service was called.

Appendix A: Security monitoring recommendations for many audit events

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Monitor for this event where “**Subject\Security ID**” is not one of these well-known security principals: LOCAL SYSTEM, NETWORK SERVICE, LOCAL SERVICE, and where “**Subject\Security ID**” is not an administrative account that is expected to have the listed **Privileges**. Especially monitor Failure events.
- If you need to monitor events related to specific Windows subsystems (“**Service\Server**”), for example **NT Local Security Authority / Authentication Service** or **Security Account Manager**, monitor this event for the corresponding “**Service\Server**.”
- If you need to monitor events related to specific Windows security services or functions (“**Service\Service Name**”), for example **LsaRegisterLogonProcess()**, monitor this event for the corresponding “**Service\Service Name**.”
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- For a specific “**Subject\Security ID**,” if there is a defined list of allowed privileges, monitor for “**Privileges**” that it should not be able to use.
- If you have a list of specific user rights which should never be used, or used only by a few accounts (for example, **SeDebugPrivilege**), trigger an alert for those “**Privileges**.”
- If you have a list of specific user rights for which every use must be reported or monitored (for example, **SeRemoteShutdownPrivilege**), trigger an alert for those “**Privileges**.”

4674(S, F): An operation was attempted on a privileged object.

 Event Properties - Event 4674, Microsoft Windows security audit...

General Details

An operation was attempted on a privileged object.

Subject: Security ID: LOCAL SERVICE

Logon ID: 0x3e5

Object:

- Object Server: LSA
- Object Type: -
- Object Name: -
- Object Handle: 0x0

Process Information:

- Process ID: 0x1f0
- Process Name: C:\Windows\System32\lsass.exe

Requested Operation:

- Desired Access: 16777216
- Privileges: SeSecurityPrivilege

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 4674
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Logged: 10/8/2015 5:22:36 PM
 Task Category: Sensitive Privilege U
 Keywords: Audit Failure
 Computer: DC01.contoso.local

Copy **Close**

Event Description:

This event generates when an attempt is made to perform privileged operations on a protected subsystem object after the object is already opened.

This event generates, for example, when SeShutdownPrivilege, SeRemoteShutdownPrivilege, or SeSecurityPrivilege is used. Failure event generates when operation attempt fails.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4674</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>13056</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T00:22:36.237816000Z" />
<EventRecordID>1099680</EventRecordID>
<Correlation />
<Execution ProcessID="496" ThreadID="504" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
```

```
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId" value="S-1-5-19" />
<Data Name="SubjectUserName" value="LOCAL SERVICE" />
<Data Name="SubjectDomainName" value="NT AUTHORITY" />
<Data Name="SubjectLogonId" value="0x3e5" />
<Data Name="ObjectServer" value="LSA" />
<Data Name="ObjectType" value="-" />
<Data Name="ObjectName" value="-" />
<Data Name="HandleId" value="0x0" />
<Data Name="AccessMask" value="16777216" />
<Data Name="PrivilegeList" value="SeSecurityPrivilege" />
```

```
<Data Name="ProcessId">0x1f0</Data>
<Data Name="ProcessName">C:\Windows\System32\lsass.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested privileged operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested privileged operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Object:

- **Object Server** [Type = UnicodeString] [Optional]: Contains the name of the Windows subsystem calling the routine. Subsystems examples are:
 - Security
 - Security Account Manager
 - NT Local Security Authority / Authentication Service
 - SC Manager
 - Win32 SystemShutdown module
 - LSA
- **Object Type** [Type = UnicodeString] [Optional]: The type of an object that was accessed during the operation.

The following table contains the list of the most common **Object Types**:

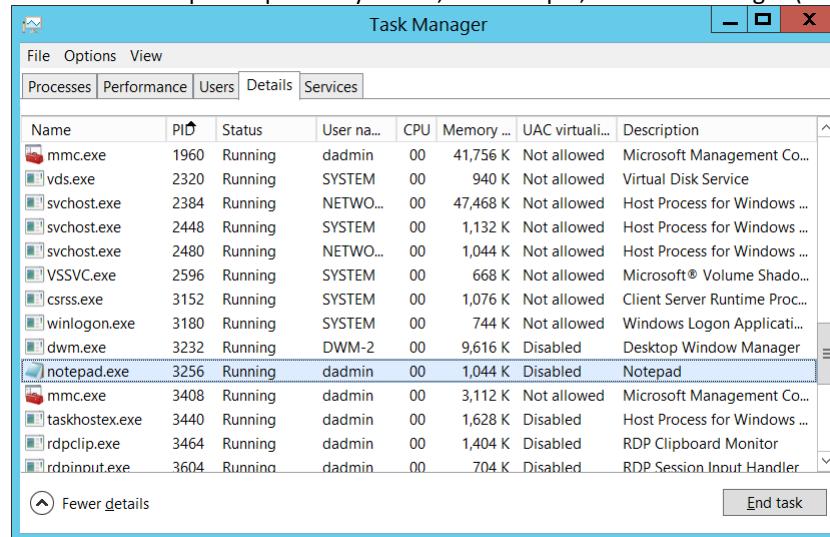
Directory	Event	Timer	Device
Mutant	Type	File	Token

Thread	Section	WindowStation	DebugObject
FilterCommunicationPort	EventPair	Driver	IoCompletion
Controller	SymbolicLink	WmiGuid	Process
Profile	Desktop	KeyedEvent	SC_MANAGER OBJECT
Key	WaitablePort	Callback	
Job	Port	FilterConnectionPort	
ALPC Port	Semaphore	Adapter	

- **Object Name** [Type = UnicodeString] [Optional]: the name of the object that was accessed during the operation.
- **Object Handle** [Type = Pointer]: hexadecimal value of a handle to **Object Name**. This field can help you correlate this event with other events that might contain the same Handle ID, for example, “4656: A handle to an object was requested” event in appropriate/other subcategory. This parameter might not be captured in the event, and in that case appears as “0x0”.

Process Information:

- **Process ID** [Type = Pointer]: hexadecimal Process ID of the process that attempted the operation on the privileged object. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688: A new process has been created](#)” **Process Information\New Process ID**.

- **Process Name** [Type = UnicodeString]: full path and the name of the executable for the process.

Requested Operation:

- **Desired Access** [Type = UnicodeString]: The desired access mask. This mask depends on **Object Server** and **Object Type** parameters values. The value of this parameter is in decimal format. There is no detailed information about this parameter in this document. If **Desired Access** is not presented, then this parameter will have “0” value.

- **Privileges** [Type = UnicodeString]: the list of user privileges which were requested. The possible privileges depend on the subcategory, either **Audit Non Sensitive Privilege Use** or **Audit Sensitive Privilege Use**, as shown in the following two tables:

Subcategory of event	Privilege Name: User Right Group Policy Name	Description
Audit Non Sensitive Privilege Use	SeChangeNotifyPrivilege: Bypass traverse checking	Required to receive notifications of changes to files or directories. This privilege also causes the system to skip all traversal access checks. With this privilege, the user can traverse directory trees even though the user may not have permissions on the traversed directory. This privilege does not allow the user to list the contents of a directory, only to traverse directories.
Audit Non Sensitive Privilege Use	SeCreateGlobalPrivilege: Create global objects	Required to create named file mapping objects in the global namespace during Terminal Services sessions.
Audit Non Sensitive Privilege Use	SeCreatePagefilePrivilege: Create a pagefile	With this privilege, the user can create and change the size of a pagefile.
Audit Non Sensitive Privilege Use	SeCreatePermanentPrivilege: Create permanent shared objects	Required to create a permanent object. This privilege is useful to kernel-mode components that extend the object namespace. Components that are running in kernel mode already have this privilege inherently; it is not necessary to assign them the privilege.
Audit Non Sensitive Privilege Use	SeCreateSymbolicLinkPrivilege: Create symbolic links	Required to create a symbolic link.
Audit Non Sensitive Privilege Use	SeIncreaseBasePriorityPrivilege: Increase scheduling priority	Required to increase the base priority of a process. With this privilege, the user can use a process with Write property access to another process to increase the execution priority assigned to the other process. A user with this privilege can change the scheduling priority of a process through the Task Manager user interface.
Audit Non Sensitive Privilege Use	SeIncreaseQuotaPrivilege: Adjust memory quotas for a process	Required to increase the quota assigned to a process. With this privilege, the user can change the maximum memory that can be consumed by a process.
Audit Non Sensitive Privilege Use	SeIncreaseWorkingSetPrivilege: Increase a process working set	Required to allocate more memory for applications that run in the context of users.
Audit Non Sensitive Privilege Use	SeLockMemoryPrivilege: Lock pages in memory	Required to lock physical pages in memory. With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).
Audit Non Sensitive Privilege Use	SeMachineAccountPrivilege: Add workstations to domain	With this privilege, the user can create a computer account. This privilege is valid only on domain controllers.
Audit Non Sensitive Privilege Use	SeManageVolumePrivilege: Perform volume maintenance tasks	Required to run maintenance tasks on a volume, such as remote defragmentation.

Audit Non Sensitive Privilege Use	SeProfileSingleProcessPrivilege: Profile single process	Required to gather profiling information for a single process. With this privilege, the user can use performance monitoring tools to monitor the performance of non-system processes.
Audit Non Sensitive Privilege Use	SeRelabelPrivilege: Modify an object label	Required to modify the mandatory integrity level of an object.
Audit Non Sensitive Privilege Use	SeRemoteShutdownPrivilege: Force shutdown from a remote system	Required to shut down a system using a network request.
Audit Non Sensitive Privilege Use	SeShutdownPrivilege: Shut down the system	Required to shut down a local system.
Audit Non Sensitive Privilege Use	SeSyncAgentPrivilege: Synchronize directory service data	This privilege enables the holder to read all objects and properties in the directory, regardless of the protection on the objects and properties. By default, it is assigned to the Administrator and LocalSystem accounts on domain controllers. With this privilege, the user can synchronize all directory service data. This is also known as Active Directory synchronization.
Audit Non Sensitive Privilege Use	SeSystemProfilePrivilege: Profile system performance	Required to gather profiling information for the entire system. With this privilege, the user can use performance monitoring tools to monitor the performance of system processes.
Audit Non Sensitive Privilege Use	SeSystemtimePrivilege: Change the system time	Required to modify the system time. With this privilege, the user can change the time and date on the internal clock of the computer. Users that are assigned this user right can affect the appearance of event logs. If the system time is changed, events that are logged will reflect this new time, not the actual time that the events occurred.
Audit Non Sensitive Privilege Use	SeTimeZonePrivilege: Change the time zone	Required to adjust the time zone associated with the computer's internal clock.
Audit Non Sensitive Privilege Use	SeTrustedCredManAccessPrivilege: Access Credential Manager as a trusted caller	Required to access Credential Manager as a trusted caller.
Audit Non Sensitive Privilege Use	SeUndockPrivilege: Remove computer from docking station	Required to undock a laptop. With this privilege, the user can undock a portable computer from its docking station without logging on.

Subcategory of event	Privilege Name: User Right Group Policy Name	Description
Audit Sensitive Privilege Use	SeAssignPrimaryTokenPrivilege: Replace a process-level token	Required to assign the <i>primary token</i> of a process. With this privilege, the user can initiate a process to replace the default token associated with a started subprocess.
Audit Sensitive Privilege Use	SeAuditPrivilege: Generate security audits	With this privilege, the user can add entries to the security log.

Audit Sensitive Privilege Use	SeBackupPrivilege: Back up files and directories	<p>Required to perform backup operations.</p> <p>With this privilege, the user can bypass file and directory, registry, and other persistent object permissions for the purposes of backing up the system. This privilege causes the system to grant all read access control to any file, regardless of the <i>access control list</i> (ACL) specified for the file. Any access request other than read is still evaluated with the ACL.</p> <p>The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • READ_CONTROL • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_READ • FILE_TRAVERSE
Audit Sensitive Privilege Use	SeCreateTokenPrivilege: Create a token object	<p>Allows a process to create a token which it can then use to get access to any local resources when the process uses NtCreateToken() or other token-creation APIs.</p> <p>When a process requires this privilege, we recommend using the LocalSystem account (which already includes the privilege), rather than creating a separate user account and assigning this privilege to it.</p>
Audit Sensitive Privilege Use	SeDebugPrivilege: Debug programs	<p>Required to debug and adjust the memory of a process owned by another account.</p> <p>With this privilege, the user can attach a debugger to any process or to the kernel.</p> <p>Developers who are debugging their own applications do not need this user right.</p> <p>Developers who are debugging new system components need this user right.</p> <p>This user right provides complete access to sensitive and critical operating system components.</p>
Audit Sensitive Privilege Use	SeImpersonatePrivilege: Impersonate a client after authentication	With this privilege, the user can impersonate other accounts.
Audit Sensitive Privilege Use	SeLoadDriverPrivilege: Load and unload device drivers	<p>Required to load or unload a device driver.</p> <p>With this privilege, the user can dynamically load and unload device drivers or other code in to kernel mode. This user right does not apply to Plug and Play device drivers.</p>
Audit Sensitive Privilege Use	SeLockMemoryPrivilege: Lock pages in memory	<p>Required to lock physical pages in memory.</p> <p>With this privilege, the user can use a process to keep data in physical memory, which prevents the system from paging the data to virtual memory on disk. Exercising this privilege could significantly affect system performance by decreasing the amount of available random access memory (RAM).</p>
Audit Sensitive Privilege Use	SeRestorePrivilege: Restore files and directories	<p>Required to perform restore operations. This privilege causes the system to grant all write access control to any file, regardless of the ACL specified for the file. Any access request other than write is still evaluated with the ACL. Additionally, this privilege enables you to set any valid user or group SID as the owner of a file. The following access rights are granted if this privilege is held:</p> <ul style="list-style-type: none"> • WRITE_DAC • WRITE_OWNER

		<ul style="list-style-type: none"> • ACCESS_SYSTEM_SECURITY • FILE_GENERIC_WRITE • FILE_ADD_FILE • FILE_ADD_SUBDIRECTORY • DELETE <p>With this privilege, the user can bypass file, directory, registry, and other persistent objects permissions when restoring backed up files and directories and determines which users can set any valid security principal as the owner of an object.</p>
Audit Sensitive Privilege Use	SeSecurityPrivilege: Manage auditing and security log	<p>Required to perform a number of security-related functions, such as controlling and viewing audit events in security event log.</p> <p>With this privilege, the user can specify object access auditing options for individual resources, such as files, Active Directory objects, and registry keys. A user with this privilege can also view and clear the security log.</p>
Audit Sensitive Privilege Use	SeSystemEnvironmentPrivilege: Modify firmware environment values	<p>Required to modify the nonvolatile RAM of systems that use this type of memory to store configuration information.</p>
Audit Sensitive Privilege Use	SeTakeOwnershipPrivilege: Take ownership of files or other objects	<p>Required to take ownership of an object without being granted discretionary access. This privilege allows the owner value to be set only to those values that the holder may legitimately assign as the owner of an object.</p> <p>With this privilege, the user can take ownership of any securable object in the system, including Active Directory objects, files and folders, printers, registry keys, processes, and threads.</p>

Security Monitoring Recommendations:

For 4674(S, F): An operation was attempted on a privileged object.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Monitor for this event where “**Subject\Security ID**” is not one of these well-known security principals: LOCAL SYSTEM, NETWORK SERVICE, LOCAL SERVICE, and where “**Subject\Security ID**” is not an administrative account that is expected to have the listed **Privileges**. Especially monitor Failure events.
- If you need to monitor events related to specific Windows subsystems (“**Object Server**”), for example LSA or **Security Account Manager**, monitor this event for the corresponding “**Object Server**.”
- If you need to monitor events related to specific Windows object types (“**Object Type**”), for example **File** or **Key**, monitor this event for the corresponding “**Object Type**.”
-
-
- If you have a pre-defined “Process Name” for the process reported in this event, monitor all events with “Process Name” not equal to your defined value.
- If you know that specific “**Subject\Security ID**” should only be able to use the privileges in a pre-defined list, monitor for events in which “**Subject\Security ID**” used “**Privileges**” that are not on that list.

- If you have a list of specific user rights which should never be used, or used only by a few accounts (for example, SeDebugPrivilege), trigger an alert for those “**Privileges**.”
- If you have a list of specific user rights for which every use must be reported or monitored (for example, SeRemoteShutdownPrivilege), trigger an alert for those “**Privileges**.”

4985(S): The state of a transaction has changed.

For some reason event “4985(S): The state of a transaction has changed.” from [Audit File System](#) subcategory generates also in this subcategory. See description of this event in [Audit File System](#) subcategory.

System

Audit IPsec Driver

Audit IPsec Driver allows you to audit events generated by IPsec driver such as the following:

- Startup and shutdown of the IPsec services.
- Network packets dropped due to integrity check failure.
- Network packets dropped due to replay check failure.
- Network packets dropped due to being in plaintext.
- Network packets received with incorrect Security Parameter Index (SPI). This may indicate that either the network card is not working correctly or the driver needs to be updated.
- Inability to process IPsec filters.

A high rate of packet drops by the IPsec filter driver may indicate attempts to gain access to the network by unauthorized systems.

Failure to process IPsec filters poses a potential security risk because some network interfaces may not get the protection that is provided by the IPsec filter.

This subcategory is outside the scope of this document.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at IPsec Driver level.
Member Server	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at IPsec Driver level.
Workstation	-	-	-	-	There is no recommendation for this subcategory in this document, unless you know exactly what you need to monitor at IPsec Driver level.

4960(**S**): IPsec dropped an inbound packet that failed an integrity check. If this problem persists, it could indicate a network issue or that packets are being modified in transit to this computer. Verify that the packets sent from the remote computer are the same as those received by this computer. This error might also indicate interoperability problems with other IPsec implementations.

4961(**S**): IPsec dropped an inbound packet that failed a replay check. If this problem persists, it could indicate a replay attack against this computer.

4962(**S**): IPsec dropped an inbound packet that failed a replay check. The inbound packet had too low a sequence number to ensure it was not a replay.

4963(**S**): IPsec dropped an inbound clear text packet that should have been secured. This is usually due to the remote computer changing its IPsec policy without informing this computer. This could also be a spoofing attack attempt.

4965(**S**): IPsec received a packet from a remote computer with an incorrect Security Parameter Index (SPI). This is usually caused by malfunctioning hardware that is corrupting packets. If these errors persist, verify that the packets sent from the remote computer are the same as those received by this computer. This error may also indicate interoperability problems with other IPsec implementations. In that case, if connectivity is not impeded, then these events can be ignored.

5478(**S**): IPsec Services has started successfully.

5479(): IPsec Services has been shut down successfully. The shutdown of IPsec Services can put the computer at greater risk of network attack or expose the computer to potential security risks.

5480(**F**): IPsec Services failed to get the complete list of network interfaces on the computer. This poses a potential security risk because some of the network interfaces may not get the protection provided by the applied IPsec filters. Use the IP Security Monitor snap-in to diagnose the problem.

5483(**F**): IPsec Services failed to initialize RPC server. IPsec Services could not be started.

5484(**F**): IPsec Services has experienced a critical failure and has been shut down. The shutdown of IPsec Services can put the computer at greater risk of network attack or expose the computer to potential security risks.

5485(**F**): IPsec Services failed to process some IPsec filters on a plug-and-play event for network interfaces. This poses a potential security risk because some of the network interfaces may not get the protection provided by the applied IPsec filters. Use the IP Security Monitor snap-in to diagnose the problem.

Audit Other System Events

Audit Other System Events contains Windows Firewall Service and Windows Firewall driver start and stop events, failure events for these services and Windows Firewall Service policy processing failures.

Audit Other System Events determines whether the operating system audits various system events.

The system events in this category include:

- Startup and shutdown of the Windows Firewall service and driver.
- Security policy processing by the Windows Firewall service.
- Cryptography key file and migration operations.
- BranchCache events.

Event volume: Low.

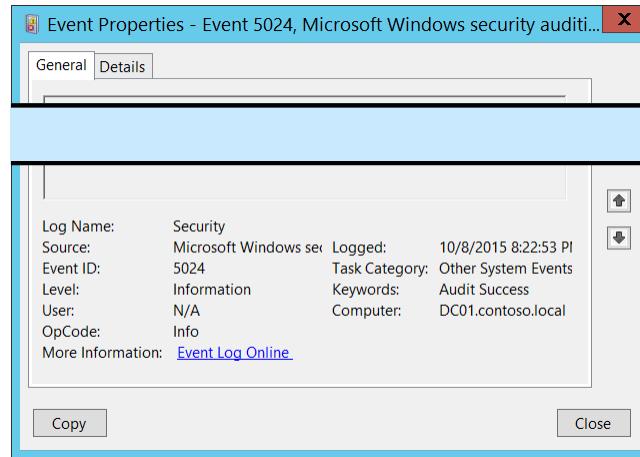
Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	We recommend enabling Success and Failure auditing because you will be able to get Windows Firewall Service and Windows Firewall Driver status events.
Member Server	Yes	Yes	Yes	Yes	We recommend enabling Success and Failure auditing because you will be able to get Windows Firewall Service and Windows Firewall Driver status events.
Workstation	Yes	Yes	Yes	Yes	We recommend enabling Success and Failure auditing because you will be able to get Windows Firewall Service and Windows Firewall Driver status events.

Events List:

- [5024\(S\)](#): The Windows Firewall Service has started successfully.
- [5025\(S\)](#): The Windows Firewall Service has been stopped.
- [5027\(F\)](#): The Windows Firewall Service was unable to retrieve the security policy from the local storage. The service will continue enforcing the current policy.
- [5028\(F\)](#): The Windows Firewall Service was unable to parse the new security policy. The service will continue with currently enforced policy.
- [5029\(F\)](#): The Windows Firewall Service failed to initialize the driver. The service will continue to enforce the current policy.
- [5030\(F\)](#): The Windows Firewall Service failed to start.
- [5032\(F\)](#): Windows Firewall was unable to notify the user that it blocked an application from accepting incoming connections on the network.
- [5033\(S\)](#): The Windows Firewall Driver has started successfully.
- [5034\(S\)](#): The Windows Firewall Driver was stopped.
- [5035\(F\)](#): The Windows Firewall Driver failed to start.
- [5037\(F\)](#): The Windows Firewall Driver detected critical runtime error. Terminating.
- [5058\(S, F\)](#): Key file operation.
- [5059\(S, F\)](#): Key migration operation.
- [6400\(-\)](#): BranchCache: Received an incorrectly formatted response while discovering availability of content.
- [6401\(-\)](#): BranchCache: Received invalid data from a peer. Data discarded.
- [6402\(-\)](#): BranchCache: The message to the hosted cache offering it data is incorrectly formatted.
- [6403\(-\)](#): BranchCache: The hosted cache sent an incorrectly formatted response to the client.
- [6404\(-\)](#): BranchCache: Hosted cache could not be authenticated using the provisioned SSL certificate.
- [6405\(-\)](#): BranchCache: %2 instance(s) of event id %1 occurred.

- [6406\(-\)](#): %1 registered to Windows Firewall to control filtering for the following: %2
- [6407\(-\)](#): 1%
- [6408\(-\)](#): Registered product %1 failed and Windows Firewall is now controlling the filtering for %2
- [6409\(-\)](#): BranchCache: A service connection point object could not be parsed.

5024(S): The Windows Firewall Service has started successfully.

 Event Properties - Event 5024, Microsoft Windows security audit... X

General	Details
-------------------------	-------------------------

Event Description:
This event generates when Windows Firewall (MpsSvc) service has started successfully.
This event is typically logged during operating system startup process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name: Security
 Source: Microsoft Windows ser Logged: 10/8/2015 8:22:53 PM
 Event ID: 5024 Task Category: Other System Events
 Level: Information Keywords: Audit Success
 User: N/A Computer: DC01.contoso.local
 OpCode: Info
 More Information: [Event Log Online](#)

[Copy](#) [Close](#)

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5024</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12292</Task>

<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T03:22:53.842816300Z" />
<EventRecordID>1101613</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="528" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

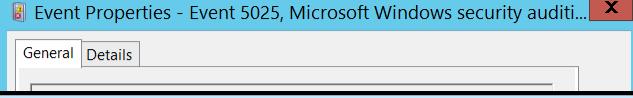
Security Monitoring Recommendations:

For 5024(S): The Windows Firewall Service has started successfully.

- Typically this event has an informational purpose. It's logged during operating system startup process.

- You should not see this event after system startup, so we recommend that you monitor it when it occurs outside the system startup process.

5025(S): The Windows Firewall Service has been stopped.

 Event Properties - Event 5025, Microsoft Windows security audit... X

[General](#) [Details](#)

Event Description:
This event generates when Windows Firewall (MpsSvc) service has been stopped.
This event is typically logged during operating system shutdown process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name:	Security
Source:	Microsoft Windows se
Event ID:	5025
Level:	Information
User:	N/A
OpCode:	Info
More Information:	Event Log Online

Copy Close

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5025</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12292</Task>
```

```
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T03:22:23.742965400Z" />
<EventRecordID>1101606</EventRecordID>
<Correlation />
<Execution ProcessID="508" ThreadID="3780" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 5025(S): The Windows Firewall Service has been stopped.

- Typically this event has an informational purpose. It's logged during operating system shutdown process.
- You should not see this event after system startup, so we recommend that you monitor it when it occurs outside the system startup process.

5027(F): The Windows Firewall Service was unable to retrieve the security policy from the local storage. The service will continue enforcing the current policy.



Event Description:

This error indicates one of two situations, low memory resources or Windows Firewall group policy registry corruption. Typically if this event occurs it indicates that Windows Firewall service was not able to start.

It typically occurs with “[5028\(S\)](#): The Windows Firewall Service was unable to parse the new security policy. The service will continue with currently enforced policy.”

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5027</EventID>
<Version>0</Version>
```

```
<Level>0</Level>
<Task>12292</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-13T23:10:05.318922900Z" />
<EventRecordID>1101848</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2000" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="ErrorCode">2147942413</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

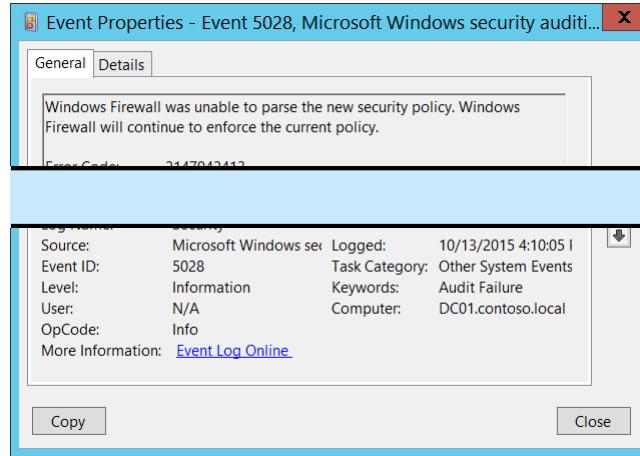
Error Code [Type = UInt32]: unique error code. For information about error codes meanings for this event use <https://technet.microsoft.com/> or other informational resources.

Security Monitoring Recommendations:

For 5027(F): The Windows Firewall Service was unable to retrieve the security policy from the local storage. The service will continue enforcing the current policy.

- This event can be a sign of software or operating system issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

5028(F): The Windows Firewall Service was unable to parse the new security policy. The service will continue with currently enforced policy.



Event Description:
This error indicates one of two situations, low memory resources or Windows Firewall group policy registry corruption. Typically if this event occurs it indicates that Windows Firewall service was not able to start. It typically occurs with “[5027\(S\)](#): The Windows Firewall Service was unable to retrieve the security policy from the local storage. The service will continue enforcing the current policy.”

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Source: Microsoft Windows se...	Logged: 10/13/2015 4:10:05 I
Event ID: 5028	Task Category: Other System Events
Level: Information	Keywords: Audit Failure
User: N/A	Computer: DC01.contoso.local
OpCode: Info	
More Information: Event Log Online	

Copy Close

```

<Level>0</Level>
<Task>12292</Task>
<Opcode>0</Opcode>
<Keywords>0x8010000000000000</Keywords>
<TimeCreated SystemTime="2015-10-13T23:10:05.318922900Z" />
<EventRecordID>1101849</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="2000" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="ErrorCode">2147942413</Data>
</EventData>
</Event>

```

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  - <System>
    <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
    <EventID>5028</EventID>
    <Version>0</Version>
  </System>
  <Level>0</Level>
  <Task>12292</Task>
  <Opcode>0</Opcode>
  <Keywords>0x8010000000000000</Keywords>
  <TimeCreated SystemTime="2015-10-13T23:10:05.318922900Z" />
  <EventRecordID>1101849</EventRecordID>
  <Correlation />
  <Execution ProcessID="500" ThreadID="2000" />
  <Channel>Security</Channel>
  <Computer>DC01.contoso.local</Computer>
  <Security />
  </System>
- <EventData>
  <Data Name="ErrorCode">2147942413</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.**Field Descriptions:**

Error Code [Type = UInt32]: unique error code. For information about error codes meanings for this event use <https://technet.microsoft.com/> or other informational resources.

Security Monitoring Recommendations:

For 5028(F): The Windows Firewall Service was unable to parse the new security policy. The service will continue with currently enforced policy.

- This event can be a sign of software or operating system issues, Windows Firewall registry errors or corruption, or Group Policy setting misconfigurations. We recommend monitoring this event and investigating the reason for the condition. Typically this event indicates configuration issues, not security issues.

5029(F): The Windows Firewall Service failed to initialize the driver. The service will continue to enforce the current policy.

Windows logs an error if either the Windows Firewall service or its driver fails to start, or if they unexpectedly terminate. The error message indicates the cause of the service failure by including an error code in the text of the message.

There is no example of this event in this document.

Event Schema:

The Windows Firewall service failed to initialize the driver. Windows Firewall will continue to enforce the current policy.

Error Code:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software or operating system issues, or a sign of malicious activity that corrupted Windows Firewall Driver. We recommend monitoring this event and investigating the reason for the condition.

5030(F): The Windows Firewall Service failed to start.

Windows logs this event if the Windows Firewall service fails to start, or if it unexpectedly terminates. The error message indicates the cause of the service failure by including an error code in the text of the message.

This event doesn't generate during Windows Firewall service failures if Windows Firewall policy is incorrect\corrupted or one of the service dependencies was not started.

There is no example of this event in this document.

Event Schema:

The Windows Firewall service failed to start.

Error Code:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software or operating system issues, or a sign of malicious activity that corrupted Windows Firewall Driver. We recommend monitoring this event and investigating the reason for the condition.

5032(F): Windows Firewall was unable to notify the user that it blocked an application from accepting incoming connections on the network.

Windows Firewall with Advanced Security can be configured to notify the user when an application is blocked by the firewall, and ask if the application should continue to be blocked in the future.

This event generates if Windows Firewall was unable to notify the user that it blocked an application from accepting incoming connections on the network.

There is no example of this event in this document.

Event Schema:

Windows Firewall was unable to notify the user that it blocked an application from accepting incoming connections on the network.

Error Code: %1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

5033(S): The Windows Firewall Driver has started successfully.



Event Description:
This event generates when Windows Firewall driver (Windows Firewall Authorization Driver service) has started successfully. This event is typically logged during operating system startup process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5033</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12292</Task>
```

<Opcode>0</Opcode>

<Keywords>0x8020000000000000</Keywords>

```

<TimeCreated SystemTime="2015-10-09T03:22:53.526024800Z" />
<EventRecordID>1101612</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="148" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 5033(S): The Windows Firewall Driver has started successfully.

- Typically this event has an informational purpose. It's logged during operating system startup process.
- You should not see this event after system startup, so we recommend that you monitor it when it occurs outside the system startup process.

5034(S): The Windows Firewall Driver was stopped.

 Event Properties - Event 5034, Microsoft Windows security auditi... X

General	Details
---------	---------

Event Description:
This event generates when Windows Firewall driver (Windows Firewall Authorization Driver service) was stopped.
This event is NOT logged during the operating system shutdown process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name: Security Source: Microsoft Windows se... Logged: 10/13/2015 4:40:55 I Event ID: 5034 Task Category: Other System Events Level: Information Keywords: Audit Success User: N/A Computer: DC01.contoso.local OpCode: Info More Information: Event Log Online	Event XML: <pre> - <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event"> - <System> <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" /> <EventID>5034</EventID> <Version>0</Version> <Level>0</Level> <Task>12292</Task> <Opcode>0</Opcode> <Keywords>0x8020000000000000</Keywords> <TimeCreated SystemTime="2015-10-13T23:40:55.482270000Z" /> <EventRecordID>1101856</EventRecordID> <Correlation /></pre>
---	--

```
<Execution ProcessID="4" ThreadID="140" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 5034(S): The Windows Firewall Driver was stopped.

- This event is NOT logged during the operating system shutdown process.
- You should not see this event during normal operating system operations, so we recommend that when it occurs, you investigate why the Windows Firewall driver was stopped.

5035(F): The Windows Firewall Driver failed to start.

Windows logs this event if Windows Firewall driver fails to start, or if it unexpectedly terminates. The error message indicates the cause of the failure by including an error code in the text of the message.

There is no example of this event in this document.

Event Schema:

The Windows Firewall Driver failed to start.

Error Code:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software or operating system issues, or a sign of malicious activity that corrupted Windows Firewall Driver. We recommend monitoring this event and investigating the reason for the condition.

5037(F): The Windows Firewall Driver detected critical runtime error. Terminating.

Windows logs this event if Windows Firewall driver fails to start, or if it unexpectedly terminates. The error message indicates the cause of the failure by including an error code in the text of the message.

There is no example of this event in this document.

Event Schema:

The Windows Firewall Driver detected a critical runtime error, terminating.

Error Code:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- This event can be a sign of software or operating system issues, or a sign of malicious activity that corrupted Windows Firewall Driver. We recommend monitoring this event and investigating the reason for the condition.

5058(S, F): Key file operation.

Event Properties - Event 5058, Microsoft Windows security auditing.

General Details

Key file operation.

Subject:

Security ID:	CONTOSO\admind
Account Name:	admind
Account Domain:	CONTOSO
Logon ID:	0x38E2D

Cryptographic Parameters:

Provider Name:	Microsoft Software Key Storage Provider
----------------	---

File Path:C:\Users\admind\AppData\Roaming\Microsoft\Crypto\Keys\c0a496c6786f0d25e8624fee96e4e580_7a1bf91d-ebdd-449c-825d-c97f2f47cd01
 Operation: Write persisted key to file.
 Return Code: 0x0

Log Name: Security
 Source: Microsoft Windows security
 Event ID: 5058
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Copy **Close**

Event Description:

This event generates when an operation (read, write, delete, and so on) was performed on a file that contains a KSP key by using a [Key Storage Provider](#) (KSP). This event generates only if one of the following KSPs were used:

- Microsoft Software Key Storage Provider
- Microsoft Smart Card Key Storage Provider

You can see these events, for example, during certificate renewal or export operations using KSP.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event
  xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  - <System>
    <Provider Name="Microsoft-Windows-Security-Auditing">
      Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
    <EventID>5058</EventID>
    <Version>0</Version>
    <Level>0</Level>
  
```

```

<Task>12292</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-14T19:32:07.888796600Z" />
<EventRecordID>1048275</EventRecordID>

```

```
<Correlation />
<Execution ProcessID="520" ThreadID="2312" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38e2d</Data>
<Data Name="ProviderName">Microsoft Software Key Storage Provider</Data>
<Data Name="AlgorithmName">ECDH_P521</Data>
<Data Name="KeyName">le-SuperAdmin-5e350d8e-ae46-458c-bac0-d8f3279c944e</Data>
<Data Name="KeyType">%%2500</Data>
<Data Name="KeyFilePath">C:\Users\dadmin\AppData\Roaming\Microsoft\Crypto\Keys\c0a496c6786f0d25e8624fee96e4e580_7a1bf91d-ebdd-449c-825d-c97f2f47cd01</Data>
<Data Name="Operation">%%2459</Data>
<Data Name="ReturnCode">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested key file operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested key file operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".

- For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Cryptographic Parameters:

- **Provider Name** [Type = UnicodeString]: the name of KSP through which the operation was performed. Can have one of the following values:
 - Microsoft Software Key Storage Provider
 - Microsoft Smart Card Key Storage Provider
- **Algorithm Name** [Type = UnicodeString]: the name of cryptographic algorithm through which the key was used or accessed. For "Read persisted key from file" operation, this typically has "**UNKNOWN**" value. Can also have one of the following values:
 - RSA – algorithm created by Ron Rivest, Adi Shamir, and Leonard Adleman.
 - DSA – Digital Signature Algorithm.
 - DH – Diffie-Hellman.
 - ECDH_P521 – Elliptic Curve Diffie-Hellman algorithm with 512-bit key length.
 - ECDH_P384 – Elliptic Curve Diffie-Hellman algorithm with 384-bit key length.
 - ECDH_P256 – Elliptic Curve Diffie-Hellman algorithm with 256-bit key length.
 - ECDSA_P256 – Elliptic Curve Digital Signature Algorithm with 256-bit key length.
 - ECDSA_P384 – Elliptic Curve Digital Signature Algorithm with 384-bit key length.
 - ECDSA_P521 – Elliptic Curve Digital Signature Algorithm with 521-bit key length.
- **Key Name** [Type = UnicodeString]: the name of the key (key container) with which operation was performed. For example, to get the list of **Key Names** for certificates for logged in user you can use "**certutil -store -user my**" command and check **Key Container** parameter in the output. Here is an output example:

Select Administrator: Windows PowerShell

```
PS C:\Windows\system32> certutil -store -user my
my "Personal"
=====
Certificate 0 =====
Serial Number: 1d000000259b1725bb8bce3a2d000200000025
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:03 PM
NotAfter: 10/13/2016 12:03 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Certificate Template Name (Certificate Type): Administrator
Non-root Certificate
Template: Administrator
Cert Hash(sha1): c4 6d 2b d3 e6 14 01 50 ff 99 44 2c a5 49 00 29 49 d8 b4 01
Key Container = dd28c3381af03257f4a229b9b31ad49a_7albf91d-ebdd-449c-825d-c97f2f47cd01
Simple container name: le-Administrator-ctfcda7f-6505-49bc-b390-46db58501934
Provider = Microsoft Enhanced Cryptographic Provider v1.0
Encryption test passed

=====
Certificate 1 =====
Serial Number: 1d0000002cd7a5d81a2148e5b100020000002c
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:22 PM
NotAfter: 10/13/2016 12:22 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Non-root Certificate
Template: SuperAdmin
Cert Hash(sha1): 92 47 28 4d 10 b7 bb 0a 3c 0a dd ce cf ef 4e 97 54 c7 a6 49
Key Container = le-SuperAdmin-5e350d8e-ae46-458c-bac0-d8f3279c944e
Unique container name: c0a496c6/86t0d25e8624tee96e4e580/_a1bf91d-ebdd-449c-825d-c97f2f47cd01
Provider = Microsoft Software Key Storage Provider
Encryption test passed
```

- **Key Type** [Type = UnicodeString]: can have one of the following values:
 - “User key.” – user’s cryptographic key.
 - “Machine key.” – machine’s cryptographic key.

Key File Operation Information:

- **File Path** [Type = UnicodeString]: full path and filename of the key file on which the operation was performed.
- **Operation** [Type = UnicodeString]: performed operation. Examples:
 - Write persisted key to file.
 - Read persisted key from file.
 - Delete key file.
- **Return Code** [Type = HexInt32]: has “**0x0**” value for Success events. For failure events, provides a hexadecimal error code number.

Security Monitoring Recommendations:

For 5058(S, F): Key file operation.

- Typically this event is required for detailed monitoring of KSP-related actions with cryptographic keys. If you need to monitor actions related to specific cryptographic keys (“**Key Name**”) or a specific “**Operation**”, such as “**Delete key file**”, create monitoring rules and use this event as an information source.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

5059(S, F): Key migration operation.

Event Properties - Event 5059, Microsoft Windows security auditing. X

General Details

Key migration operation.

Subject: Security ID: CONTOSO\admind

Logon ID: 0x00000000000000000000000000000000

Cryptographic Parameters:

- Provider Name: Microsoft Software Key Storage Provider
- Algorithm Name: ECDH_P521
- Key Name: le-SuperAdmin-795fd6c1-2fae-4bef-a6bc-4f4d464bc083
- Key Type: User key.

Additional Information:

Operation:	Export of persistent cryptographic key.
Return Code:	0x0

Log Name: Security
 Source: Microsoft Windows security audit
 Event ID: 5059
 Level: Information
 User: N/A
 OpCode: Info
 More Information: [Event Log Online](#)

Logged: 10/14/2015 12:42:08 PM
 Task Category: Other System Events
 Keywords: Audit Success
 Computer: DC01.contoso.local

Copy Close

Event Description:

This event generates when a cryptographic key is exported or imported using a [Key Storage Provider](#) (KSP). This event generates only if one of the following KSPs were used:

- Microsoft Software Key Storage Provider
- Microsoft Smart Card Key Storage Provider

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5059</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12292</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-14T19:42:08.135265600Z" />
<EventRecordID>1048447</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="3496" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmin</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38e2d</Data>
<Data Name="ProviderName">Microsoft Software Key Storage Provider</Data>
<Data Name="AlgorithmName">ECDH_P521</Data>
<Data Name="KeyName">le-SuperAdmin-795fd6c1-2fae-4bef-a6bc-4f4d464bc083</Data>
<Data Name="KeyType">%2500</Data>
<Data Name="Operation">%2464</Data>
<Data Name="ReturnCode">0x0</Data>
```

```
</EventData>  
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested key migration operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

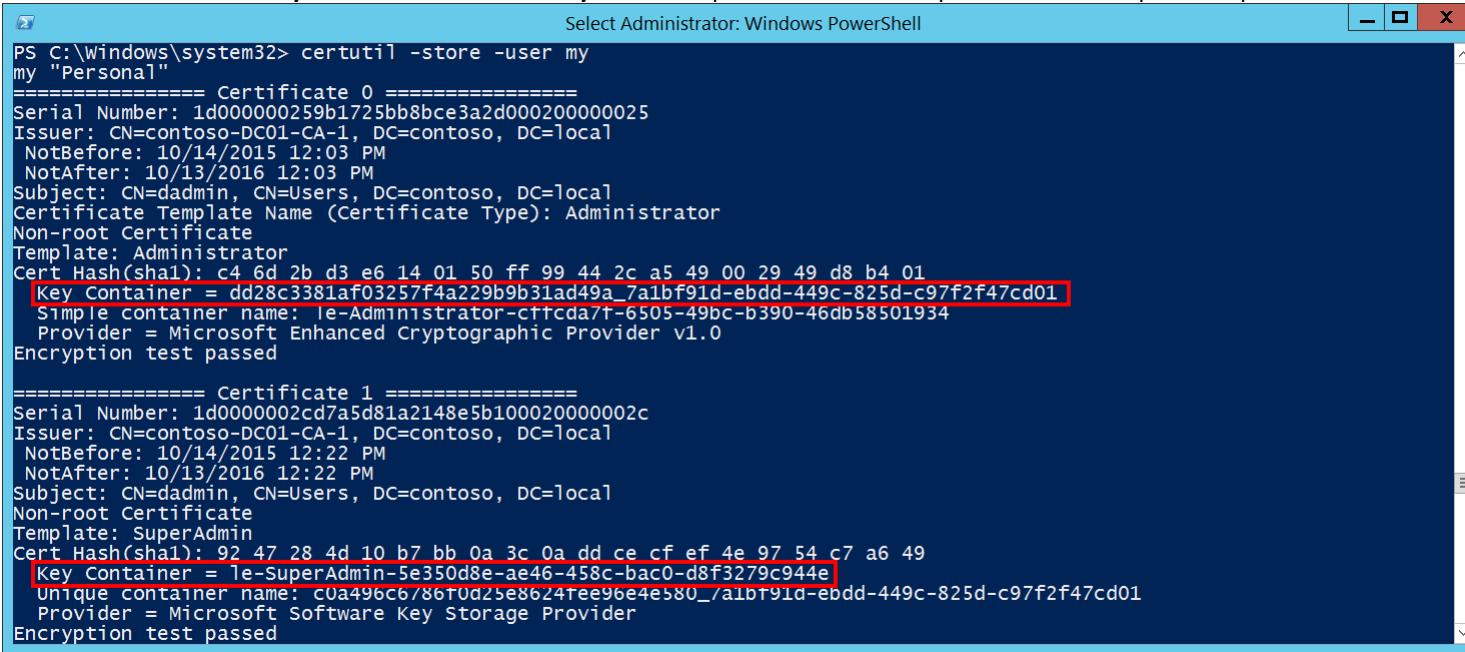
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested key migration operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Cryptographic Parameters:

- **Provider Name** [Type = UnicodeString]: the name of KSP through which the operation was performed. Can have one of the following values:
 - Microsoft Software Key Storage Provider
 - Microsoft Smart Card Key Storage Provider
- **Algorithm Name** [Type = UnicodeString]: the name of cryptographic algorithm through which the key was used or accessed. For "Read persisted key from file" operation, this typically has "**UNKNOWN**" value. Can also have one of the following values:
 - RSA – algorithm created by Ron Rivest, Adi Shamir, and Leonard Adleman.
 - DSA – Digital Signature Algorithm.
 - DH – Diffie-Hellman.
 - ECDH_P521 – Elliptic Curve Diffie-Hellman algorithm with 512-bit key length.
 - ECDH_P384 – Elliptic Curve Diffie-Hellman algorithm with 384-bit key length.
 - ECDH_P256 – Elliptic Curve Diffie-Hellman algorithm with 256-bit key length.
 - ECDSA_P256 – Elliptic Curve Digital Signature Algorithm with 256-bit key length.
 - ECDSA_P384 – Elliptic Curve Digital Signature Algorithm with 384-bit key length.

- ECDSA_P521 – Elliptic Curve Digital Signature Algorithm with 521-bit key length.
- **Key Name** [Type = UnicodeString]: the name of the key (key container) with which operation was performed. For example, to get the list of **Key Names** for certificates for logged in user you can use “**certutil -store -user my**” command and check **Key Container** parameter in the output. Here is an output example:



```

Select Administrator: Windows PowerShell
PS C:\Windows\system32> certutil -store -user my
my "Personal"
=====
Certificate 0 =====
Serial Number: 1d000000259b1725bb8bce3a2d000200000025
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:03 PM
NotAfter: 10/13/2016 12:03 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Certificate Template Name (Certificate Type): Administrator
Non-root Certificate
Template: Administrator
Cert Hash(sh1): c4 6d 2b d3 e6 14 01 50 ff 99 44 2c a5 49 00 29 49 d8 b4 01
Key Container = dd28c3381af03257f4a229b9b31ad49a_7a1bf91d-ebdd-449c-825d-c97f2f47cd01
  Simple container name: le-Administrator-cttcd7f-6505-49bc-b390-46db58501934
  Provider = Microsoft Enhanced Cryptographic Provider v1.0
Encryption test passed

=====
Certificate 1 =====
Serial Number: 1d0000002cd7a5d81a2148e5b100020000002c
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:22 PM
NotAfter: 10/13/2016 12:22 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Non-root Certificate
Template: SuperAdmin
Cert Hash(sh1): 92 47 28 4d 10 b7 bb 0a 3c 0a dd ce cf ef 4e 97 54 c7 a6 49
Key Container = le-SuperAdmin-5e350d8e-ae46-458c-bac0-d8f3279c944e
  Unique container name: cu0496c6/86f0d25e8624feef96e4e580/_a1bf91d-ebdd-449c-825d-c97f2f47cd01
  Provider = Microsoft Software Key Storage Provider
Encryption test passed
  
```

- **Key Type** [Type = UnicodeString]: can have one of the following values:
 - “User key.” – user’s cryptographic key.
 - “Machine key.” – machine’s cryptographic key.

Additional Information:

- **Operation** [Type = UnicodeString]: performed operation. Examples:
 - “**Export of persistent cryptographic key**.” – typically generates during key read operations, which means that the key was taken for read purposes. But it also generates during real key export operations (export certificate with private key, for example).
 - “**Import of persistent cryptographic key**.” – key import operation was performed (import certificate with private key, for example).
- **Return Code** [Type = HexInt32]: has “**0x0**” value for Success events. For failure events, provides a hexadecimal error code number.

Security Monitoring Recommendations:

For 5059(S, F): Key migration operation.

- Typically this event is required for detailed monitoring of KSP-related actions with cryptographic keys. If you need to monitor actions related to specific cryptographic keys (“**Key Name**”) or a specific “**Operation**”, such as “**Export of persistent cryptographic key**”, create monitoring rules and use this event as an information source.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

6400(-): BranchCache: Received an incorrectly formatted response while discovering availability of content.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: Received an incorrectly formatted response while discovering availability of content.

IP address of the client that sent this response:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6401(-): BranchCache: Received invalid data from a peer. Data discarded.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: Received invalid data from a peer. Data discarded.

IP address of the client that sent this data:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6402(-): BranchCache: The message to the hosted cache offering it data is incorrectly formatted.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: The message to the hosted cache offering it data is incorrectly formatted.

IP address of the client that sent this message: %1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6403(-): BranchCache: The hosted cache sent an incorrectly formatted response to the client.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: The hosted cache sent an incorrectly formatted response to the client's message to offer it data.

Domain name of the hosted cache is:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6404(-): BranchCache: Hosted cache could not be authenticated using the provisioned SSL certificate.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: Hosted cache could not be authenticated using the provisioned SSL certificate.

Domain name of the hosted cache:%1

Error Code:%2

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6405(-): BranchCache: %2 instance(s) of event id %1 occurred.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: %2 instance(s) of event id %1 occurred.

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6406(-): %1 registered to Windows Firewall to control filtering for the following: %2.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

%1 registered to Windows Firewall to control filtering for the following:

%2.

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6407(-): 1%.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6408(-): Registered product %1 failed and Windows Firewall is now controlling the filtering for %2.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

Registered product %1 failed and Windows Firewall is now controlling the filtering for %2.

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

6409(-): BranchCache: A service connection point object could not be parsed.

[BranchCache](#) events are outside the scope of this document.

There is no example of this event in this document.

Event Schema:

BranchCache: A service connection point object could not be parsed.

SCP object GUID: %1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

Audit Security State Change

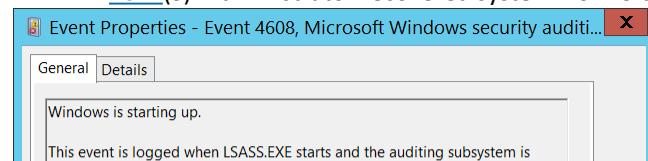
Audit Security State Change contains Windows startup, recovery, and shutdown events, and information about changes in system time.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	The volume of events in this subcategory is very low and all of them are important events and have security relevance. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Member Server	Yes	No	Yes	No	The volume of events in this subcategory is very low and all of them are important events and have security relevance. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.
Workstation	Yes	No	Yes	No	The volume of events in this subcategory is very low and all of them are important events and have security relevance. This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.

Events List:

- [4608\(S\)](#): Windows is starting up.
- [4609\(S\)](#): Windows is shutting down.
- [4616\(S\)](#): The system time was changed.
- [4621\(S\)](#): Administrator recovered system from CrashOnAuditFail.

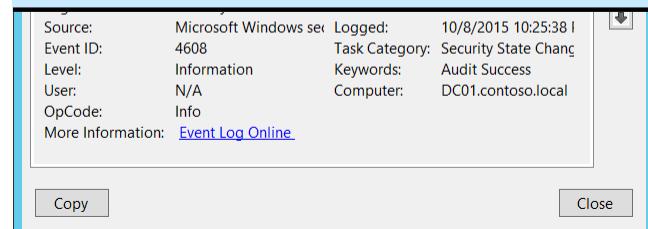


4608(S): Windows is starting up.

Event Description:

This event is logged when LSASS.EXE process starts and the auditing subsystem is initialized.
It typically generates during operating system startup process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.



Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4608</EventID>
```

```

<Version>0</Version>
<Level>0</Level>
<Task>12288</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T05:25:38.222242500Z" />
<EventRecordID>1101704</EventRecordID>
<Correlation />
<Execution ProcessID="508" ThreadID="512" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
<EventData />
</Event>

```

Event Properties - Event 4616, Microsoft Windows security auditing.

General **Details**

The system time was changed.

Subject:

Security ID:	CONTOSO\adadmin
Account Name:	adadmin
Account Domain:	CONTOSO
Logon ID:	0x48F29

Process Information:

Process ID:	0x1074
Name:	C:\Windows\WinSxS\amd64_microsoft-windows-com-surrogate-core_31bf3856ad364e35_6.3.9600.16384_none_25a8f0faa8f185c\dllhost.exe

Previous Time: 2015-10-09T05:04:30.000941900Z
New Time: 2015-10-09T05:04:30.000000000Z

This event is generated when the system time is changed. It is normal for the Windows Time Service, which runs with System privilege, to change the system time on a regular basis. Other system time changes may be indicative of attempts to tamper with the computer.

Log Name: Security

User: N/A Computer: DC01.contoso.local

OpCode: Info More Information: [Event Log Online](#)

Copy **Close**

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 4608(S): Windows is starting up.

- With this event, you can track system startup events.

4609(S): Windows is shutting down.

Currently this event doesn't generate. It is a defined event, but it is never invoked by the operating system.

4616(S): The system time was changed.

Event Description:

This event generates every time system time was changed.

This event is always logged regardless of the "Audit Security State Change" sub-category setting.

You will typically see these events with "Subject\Security ID" = "**LOCAL SERVICE**", these are normal time correction actions.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
  © 2016 Microsoft. All rights reserved.

```

```
<EventID>4616</EventID>
<Version>1</Version>
<Level>0</Level>
<Task>12288</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-09T05:04:29.995794600Z" />
<EventRecordID>1101699</EventRecordID>
<Correlation />
<Execution ProcessID="4" ThreadID="148" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SubjectUserSid">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
  <Data Name="SubjectUserName">dadmin</Data>
  <Data Name="SubjectDomainName">CONTOSO</Data>
  <Data Name="SubjectLogonId">0x48f29</Data>
  <Data Name="PreviousTime">2015-10-09T05:04:30.000941900Z</Data>
  <Data Name="NewTime">2015-10-09T05:04:30.000000000Z</Data>
  <Data Name="ProcessId">0x1074</Data>
  <Data Name="ProcessName">C:\Windows\WinSxS\amd64_microsoft-windows-com-surrogate-core_31bf3856ad364e35_6.3.9600.16384_none_25a8f00faa8f185c\dllhost.exe</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions:

- 0 - Windows Server 2008, Windows Vista.
- 1 - Windows Server 2008 R2, Windows 7.
 - Added “Process Information” section.

Field Descriptions:

Subject:

- **Security ID [Type = SID]:** SID of account that requested the “change system time” operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

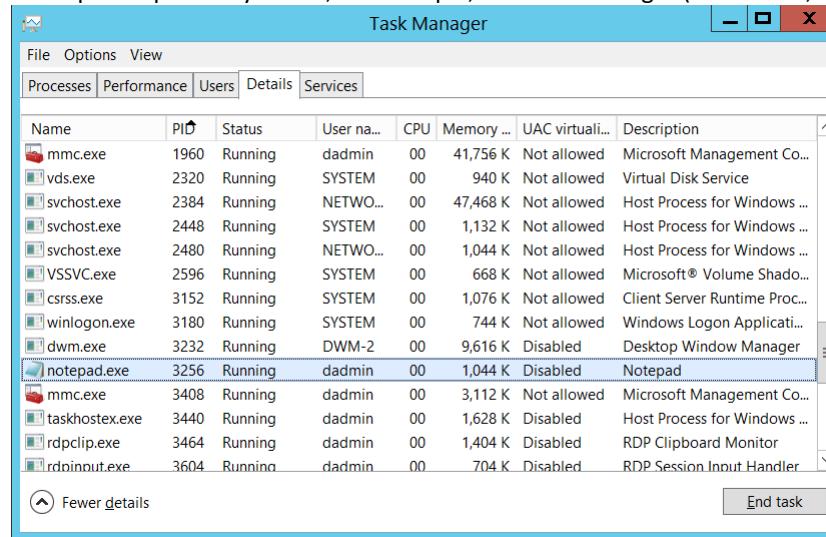
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it

in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested the “change system time” operation.
- **Account Domain** [Type = UnicodeString]: subject’s domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is “NT AUTHORITY”.
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: “Win81”.
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, “[4624](#): An account was successfully logged on.”

Process Information [Version 1]:

- **Process ID** [Type = Pointer] [Version 1]: hexadecimal Process ID of the process that changed the system time. Process ID (PID) is a number used by the operating system to uniquely identify an active process. To see the PID for a specific process you can, for example, use Task Manager (Details tab, PID column):



If you convert the hexadecimal value to decimal, you can compare it to the values in Task Manager.

You can also correlate this process ID with a process ID in other events, for example, “[4688](#): A new process has been created” **Process Information\New Process ID**.

- **Name** [Type = UnicodeString] [Version 1]: full path and the name of the executable for the process.

Previous Time [Type = FILETIME]: previous time in UTC time zone. The format is YYYY-MM-DDThh:mm:ss.nnnnnnnnZ:

- Y - years
- M - months

- D - days
- T - the beginning of the time element, as specified in [ISO 8601](#).
- h - hours
- m - minutes
- s - seconds
- n - fractional seconds
- Z - the zone designator for the zero UTC offset. "09:30 UTC" is therefore represented as "09:30Z". "14:45:15 UTC" would be "14:45:15Z".

New Time [Type = FILETIME]: new time that was set in **UTC** time zone. The format is **YYYY-MM-DDThh:mm:ss.nnnnnnnnZ:**

- Y - years
- M - months
- D - days
- T - the beginning of the time element, as specified in [ISO 8601](#).
- h - hours
- m - minutes
- s - seconds
- n - fractional seconds
- Z - the zone designator for the zero UTC offset. "09:30 UTC" is therefore represented as "09:30Z". "14:45:15 UTC" would be "14:45:15Z".

Security Monitoring Recommendations:

For 4616(S): The system time was changed.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Report all “**Subject\Security ID**” not equals “**LOCAL SERVICE**”, which means that the time change was not made not by Windows Time service.
- Report all “**Process Information\Name**” not equals “**C:\Windows\System32\svchost.exe**” (path to svchost.exe can be different, you can search for “svchost.exe” substring), which means that the time change was not made not by Windows Time service.
- If you have a pre-defined “**Process Name**” for the process reported in this event, monitor all events with “**Process Name**” not equal to your defined value.
- You can monitor to see if “**Process Name**” is not in a standard folder (for example, not in **System32** or **Program Files**) or is in a restricted folder (for example, **Temporary Internet Files**).
- If you have a pre-defined list of restricted substrings or words in process names (for example, “**mimikatz**” or “**cain.exe**”), check for these substrings in “**Process Name**.”

4621(S): Administrator recovered system from CrashOnAuditFail.

This event is logged after a system reboots following [CrashOnAuditFail](#). It generates when CrashOnAuditFail = 2.

There is no example of this event in this document.

Event Schema:

Administrator recovered system from CrashOnAuditFail. Users who are not administrators will now be allowed to log on. Some auditable activity might not have been recorded.

Value of CrashOnAuditFail:%1

This event is logged after a system reboots following CrashOnAuditFail.

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- We recommend triggering an alert for any occurrence of this event. The event shows that the system halted because it could not record an auditable event in the Security Log, as described in [CrashOnAuditFail](#).
- If your computers don't have the [CrashOnAuditFail](#) flag enabled, then this event will be a sign that some settings are not set to baseline settings or were changed.

Audit Security System Extension

Audit Security System Extension contains information about the loading of an authentication package, notification package, or security package, plus information about trusted logon process registration events.

Changes to security system extensions in the operating system include the following activities:

- Security extension code is loaded (for example, an authentication, notification, or security package). Security extension code registers with the Local Security Authority and will be used and trusted to authenticate logon attempts, submit logon requests, and be notified of any account or password changes. Examples of this extension code are Security Support Providers, such as Kerberos and NTLM.
- A service is installed. An audit log is generated when a service is registered with the Service Control Manager. The audit log contains information about the service name, binary, type, start type, and service account.

Attempts to install or load security system extensions or services are critical system events that could indicate a security breach.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	No	Yes	No	<p>The main reason why we recommend Success auditing for this subcategory is “4697(S): A service was installed in the system.”</p> <p>For other events we strongly recommend monitoring a whitelist of allowed security extensions (authenticated packages, logon processes, notification packages, and security packages). Otherwise it's hard to pull useful information from these events, except event 4611 which typically should have “SYSTEM” as value for “Subject” field.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Member Server	Yes	No	Yes	No	<p>The main reason why we recommend Success auditing for this subcategory is “4697(S): A service was installed in the system.”</p> <p>For other events we strongly recommend monitoring a whitelist of allowed security extensions (authenticated packages, logon processes, notification packages, and security packages). Otherwise it's hard to pull useful information from these events, except event 4611 which typically should display “SYSTEM” for the “Subject” field.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>
Workstation	Yes	No	Yes	No	<p>The main reason why we recommend Success auditing for this subcategory is “4697(S): A service was installed in the system.”</p> <p>For other events we strongly recommend monitoring a whitelist of allowed security extensions (authenticated packages, logon processes, notification packages, and security packages). Otherwise it's hard to pull useful information from these events, except event 4611 which typically should display “SYSTEM” for the “Subject” field.</p> <p>This subcategory doesn't have Failure events, so there is no recommendation to enable Failure auditing for this subcategory.</p>

Events List:

- [4610\(S\)](#): An authentication package has been loaded by the Local Security Authority.
- [4611\(S\)](#): A trusted logon process has been registered with the Local Security Authority.
- [4614\(S\)](#): A notification package has been loaded by the Security Account Manager.
- [4622\(S\)](#): A security package has been loaded by the Local Security Authority.
- [4697\(S\)](#): A service was installed in the system.

4610(S): An authentication package has been loaded by the Local Security Authority.

 Event Properties - Event 4610, Microsoft Windows security auditing.

General		Details	
<p>An authentication package has been loaded by the Local Security Authority. This authentication package will be used to authenticate logon attempts.</p> <p>Authentication Package Name: C:\Windows\system32\msv1_0.dll : MICROSOFT_AUTHENTICATION_PACKAGE_V1_0</p>			
Event ID:	4610	Task Category:	Security System Extension
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	DC01.contoso.local
OpCode:	Info		
More Information: Event Log Online			
<input type="button" value="Copy"/>		<input type="button" value="Close"/>	

```

<EventID>4610</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12289</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-14T03:36:41.391489300Z" />
<EventRecordID>1048138</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="520" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
```

Event Description:

This event generates every time [Authentication Package](#) has been loaded by the Local Security Authority ([LSA](#)).

Each time the system starts, the LSA loads the Authentication Package DLLs from **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Authentication Packages** registry value and performs the initialization sequence for every package located in these DLLs.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-5BA-3E3B0328C30D}" />

```

```
<Data Name="AuthenticationPackageName">C:\Windows\system32\msv1_0.DLL : MICROSOFT_AUTHENTICATION_PACKAGE_V1_0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Authentication Package Name [Type = UnicodeString]: the name of loaded [Authentication Package](#). The format is: **DLL_PATH_AND_NAME**: AUTHENTICATION_PACKAGE_NAME. By default the only one Authentication Package loaded by Windows 10 is "[MICROSOFT AUTHENTICATION PACKAGE V1_0](#)".

Security Monitoring Recommendations:

For 4610(S): An authentication package has been loaded by the Local Security Authority.

- Report all "**Authentication Package Name**" not equals "C:\Windows\system32\msv1_0.DLL : MICROSOFT_AUTHENTICATION_PACKAGE_V1_0", because by default this is the only Authentication Package loaded by Windows 10.
- Typically this event has an informational purpose. If you have a pre-defined list of allowed Authentication Packages in the system, then you can check whether "**Authentication Package Name**" is in your defined list.

4611(S): A trusted logon process has been registered with the Local Security Authority.

 Event Properties - Event 4611, Microsoft Windows security audit...

General **Details**

A trusted logon process has been registered with the Local Security Authority. This logon process will be trusted to submit logon requests.

Subject:

Security ID:	SYSTEM
Account Name:	DC01\$
Account Domain:	CONTOSO
Logon ID:	0x3E7

Log Name: Security
Source: Microsoft Windows sec
Event ID: 4611
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/13/2015 8:43:29 I
Task Category: Security System Ext
Keywords: Audit Success
Computer: DC01.contoso.local

Copy **Close**

Event Description:

This event indicates that a logon process has registered with the Local Security Authority ([LSA](#)). Also, logon requests will now be accepted from this source.

At the technical level, the event does not come from the registration of a trusted logon process, but from a confirmation that the process is a trusted logon process. If it is a trusted logon process, the event generates.

A logon process is a trusted part of the operating system that handles the overall logon function for different logon methods (network, interactive, etc.).

You typically see these events during operating system startup or user logon and authentication actions.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4611</EventID>
<Version>0</Version>
<Level>0</Level>
```

<Task>12289</Task>

<Opcode>0</Opcode>

<Keywords>0x8020000000000000</Keywords>

```
<TimeCreated SystemTime="2015-10-14T03:43:29.604031000Z" />
<EventRecordID>1048175</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="548" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">DC01$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="LogonProcessName">Winlogon</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that registered the trusted logon process. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that registered the trusted logon process.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Logon Process Name [Type = UnicodeString]: the name of registered logon process.

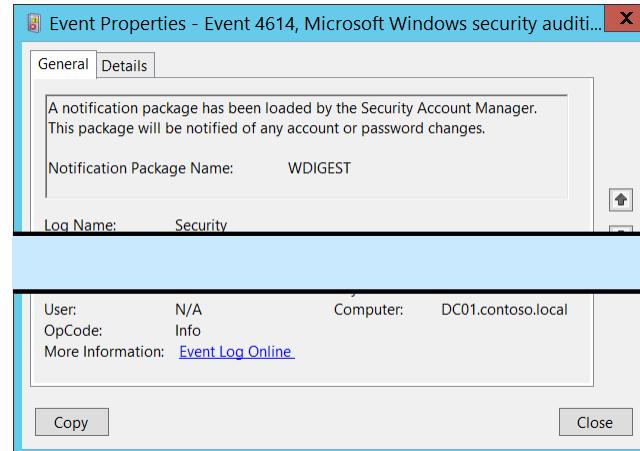
Security Monitoring Recommendations:

For 4611(S): A trusted logon process has been registered with the Local Security Authority.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. Because this event is typically triggered by the SYSTEM account, we recommend that you report it whenever “**Subject\Security ID**” is not SYSTEM.
- Typically this event has an informational purpose. If you defined the list of allowed Logon Processes in the system, then you can check is “**Logon Process Name**” field value in the whitelist or not.
-

4614(S): A notification package has been loaded by the Security Account Manager.



Event Description:

This event generates every time a Notification Package has been loaded by the [Security Account Manager](#).

In reality, starting with Windows Vista, a notification package should be interpreted as afs [Password Filter](#).

Password Filters are DLLs that are loaded or called when passwords are set or changed.

Each time a system starts, it loads the notification package DLLs from

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Notification Packages registry value and performs the initialization sequence for every package.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
  <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
```

```
<EventID>4614</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12289</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-14T03:36:43.073484900Z" />
<EventRecordID>1048140</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="520" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
```

```
</System>
- <EventData>
<Data Name="NotificationPackageName">WDIGEST</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

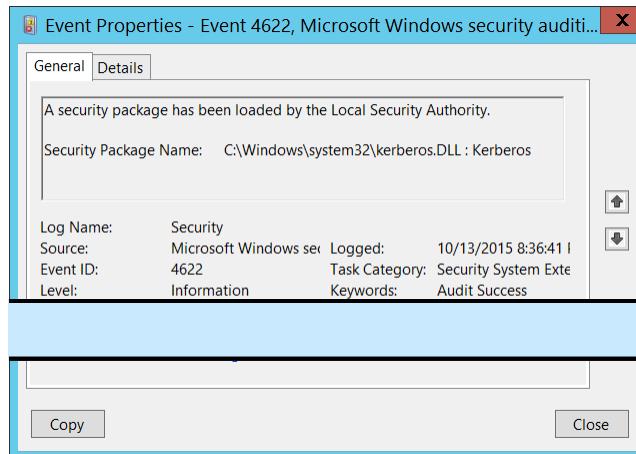
Notification Package Name [Type = UnicodeString]: the name of loaded Notification Package.

Security Monitoring Recommendations:

For 4614(S): A notification package has been loaded by the Security Account Manager.

- Typically this event has an informational purpose. If you defined the list of allowed Notification Packages in the system, then you can check is “**Notification Package Name**” field value in the whitelist or not.

4622(S): A security package has been loaded by the Local Security Authority.



Event Description:

This event generates every time [Security Package](#) has been loaded by the Local Security Authority ([LSA](#)).

Security Package is the software implementation of a security protocol (Kerberos, NTLM, for example). Security packages are contained in security support provider DLLs or security support provider/authentication package DLLs.

Each time the system starts, the LSA loads the Security Package DLLs from

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig\Security Packages registry value and performs the initialization sequence for every package located in these DLLs.

It is also possible to add security package dynamically using [AddSecurityPackage](#) function, not only during system startup process.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
```

```
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4622</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12289</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
```

```

<TimeCreated SystemTime="2015-10-14T03:36:41.359331100Z" />
<EventRecordID>1048131</EventRecordID>
<Correlation />
<Execution ProcessID="516" ThreadID="520" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
  <Data Name="SecurityPackageName">C:\Windows\system32\kerberos.DLL : Kerberos</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Security Package Name [Type = UnicodeString]: the name of loaded Security Package. The format is: **DLL_PATH_AND_NAME**: SECURITY_PACKAGE_NAME.

These are some Security Package DLLs loaded by default in Windows 10:

- C:\Windows\system32\schannel.DLL : Microsoft Unified Security Protocol Provider
- C:\Windows\system32\schannel.DLL : Schannel
- C:\Windows\system32\cloudAP.DLL : CloudAP

- C:\Windows\system32\wdigest.DLL : WDigest
- C:\Windows\system32\pku2u.DLL : pku2u
- C:\Windows\system32\tspkg.DLL : TSSP
- C:\Windows\system32\msv1_0.DLL : NTLM
- C:\Windows\system32\kerberos.DLL : Kerberos
- C:\Windows\system32\negoexts.DLL : NegoExtender
- C:\Windows\system32\lsasrv.dll : Negotiate

Security Monitoring Recommendations:

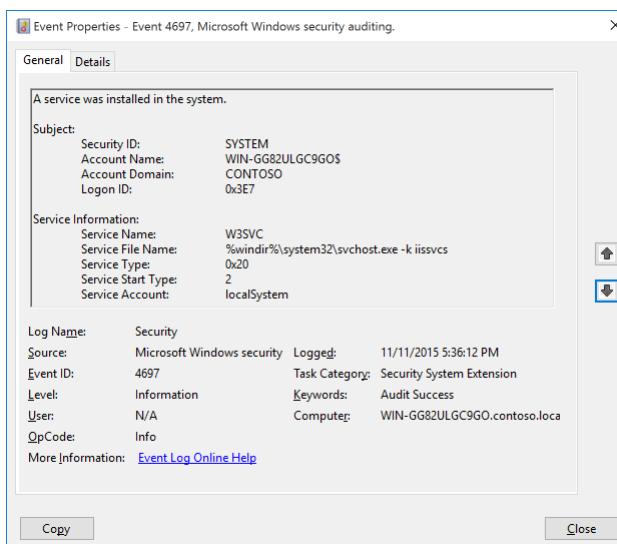
For 4622(S): A security package has been loaded by the Local Security Authority.

- Typically this event has an informational purpose. If you defined the list of allowed Security Packages in the system, then you can check is “**Security Package Name**” field value in the whitelist or not.

4697(S): A service was installed in the system.

Event Description:

This event generates when new service was installed in the system.



Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4697</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12289</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T01:36:11.991070500Z" />
<EventRecordID>2778</EventRecordID>
<Correlation ActivityID="{913FBE70-1CE6-0000-67BF-3F91E61CD101}" />
<Execution ProcessID="736" ThreadID="2800" />
<Channel>Security</Channel>
<Computer>WIN-GG82ULGC9GO.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserSid">S-1-5-18</Data>
<Data Name="SubjectUserName">WIN-GG82ULGC9GO$</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x3e7</Data>
<Data Name="ServiceName">AppHostSvc</Data>
<Data Name="ServiceFileName">%windir%\system32\svchost.exe -k apphost</Data>
<Data Name="ServiceType">0x20</Data>
<Data Name="ServiceStartType">2</Data>
<Data Name="ServiceAccount">localSystem</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2016, Windows 10.

Event Versions: 0.

Field Descriptions:

Subject:

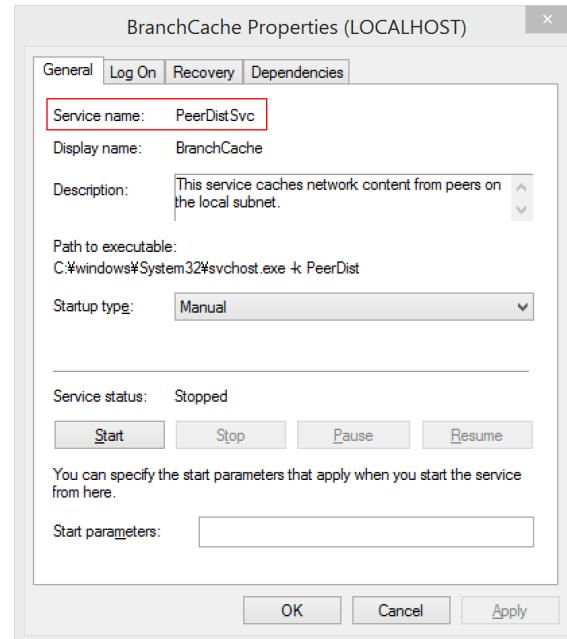
- **Security ID** [Type = SID]: SID of account that was used to install the service. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that was used to install the service.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Service Information:

- **Service Name** [Type = UnicodeString]: the name of installed service.



- **Service File Name** [Type = UnicodeString]: This is the fully rooted path to the file that the Service Control Manager will execute to start the service. If command-line parameters are specified as part of the image path, those are logged.

Note that this is the path to the file when the service is created. If the path is changed afterwards, the change is not logged. This would have to be tracked via Process Create events.

- **Service Type** [Type = HexInt32]: Indicates the [type](#) of service that was registered with the Service Control Manager. It can be one of the following:

Value	Service Type	Description
0x1	Kernel Driver	A Kernel device driver such as a hard disk or other low-level hardware device driver.
0x2	File System Driver	A file system driver, which is also a Kernel device driver.
0x8	Recognizer Driver	A file system driver used during startup to determine the file systems present on the system.
0x10	Win32 Own Process	A Win32 program that can be started by the Service Controller and that obeys the service control protocol. This type of Win32 service runs in a process by itself (this is the most common).
0x20	Win32 Share Process	A Win32 service that can share a process with other Win32 services. (see: http://msdn.microsoft.com/en-us/library/windows/desktop/ms685967(v=vs.85).aspx)
0x110	Interactive Own Process	A service that should be run as a standalone process and can communicate with the desktop. (see: http://msdn.microsoft.com/en-us/library/windows/desktop/ms683502(v=vs.85).aspx)
0x120	Interactive Share Process	A service that can share address space with other services of the same type and can communicate with the desktop.

- **Service Start Type** [Type = HexInt32]: The service start type can have one of the following values (see:

[http://msdn.microsoft.com/en-us/library/windows/desktop/ms682450\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/ms682450(v=vs.85).aspx)):

Value	Service Type	Description
0	Boot	A device driver started by the system loader. This value is valid only for driver services.
1	System	A device driver started by the <code>IoInitSystem()</code> function. This value is valid only for driver services.
2	Automatic	A service started automatically by the service control manager during system startup.
2	Automatic Delayed	A service started after all auto-start services have started, plus a delay. Delayed Auto Start services are started one at a time in a serial fashion.
3	Manual	Manual start. A service started by the service control manager when a process calls the <code>StartService</code> function.
4	Disabled	A service that cannot be started. Attempts to start the service result in the error code <code>ERROR_SERVICE_DISABLED</code> .

Most services installed are configured to **Auto Load**, so that they start automatically after Services.exe process is started.

- **Service Account** [Type = UnicodeString]: The security context that the service will run as when started. Note that this is what was configured when the service was installed, if the account is changed later that is not logged.

The service account parameter is only populated if the service type is a "Win32 Own Process" or "Win32 Share Process" (displayed as "User Mode Service."). Kernel drivers do not have a service account name logged.

If a service (Win32 Own/Share process) is installed but no account is supplied, then LocalSystem is used.

The token performing the logon is inspected, and if it has a SID then that SID value is populated in the event (in the System/Security node), if not, then it is blank.

Security Monitoring Recommendations:

For 4697(S): A service was installed in the system.

[Appendix A: Security monitoring recommendations for many audit events](#)

- Important For this event, also see Appendix A: Security monitoring recommendations for many audit events. We recommend monitoring for this event, especially on high value assets or computers, because a new service installation should be planned and expected. Unexpected service installation should trigger an alert.
- Monitor for all events where “**Service File Name**” is not located in %windir% or “**Program Files/Program Files (x86)**” folders. Typically new services are located in these folders.
- Report all “**Service Type**” equals “**0x1**”, “**0x2**” or “**0x8**”. These service types start first and have almost unlimited access to the operating system from the beginning of operating system startup. These types are very rarely installed.
- Report all “**Service Start Type**” equals “**0**” or “**1**”. These service start types are used by drivers, which have unlimited access to the operating system.
- Report all “**Service Start Type**” equals “**4**”. It is not common to install a new service in the **Disabled** state.
- Report all “**Service Account**” not equals “**localSystem**”, “**localService**” or “**networkService**” to identify services which are running under a user account.

Audit System Integrity

Audit System Integrity determines whether the operating system audits events that violate the integrity of the security subsystem.

Activities that violate the integrity of the security subsystem include the following:

- Audited events are lost due to a failure of the auditing system.
- A process uses an invalid local procedure call (LPC) port in an attempt to impersonate a client, reply to a client address space, read to a client address space, or write from a client address space.
- A remote procedure call (RPC) integrity violation is detected.
- A code integrity violation with an invalid hash value of an executable file is detected.
- Cryptographic tasks are performed.

Violations of security subsystem integrity are critical and could indicate a potential security attack.

Event volume: Low.

Computer Type	General		Stronger		Comments
	Success	Failure	Success	Failure	
Domain Controller	Yes	Yes	Yes	Yes	The main reason why we recommend Success auditing for this subcategory is to be able to get RPC integrity violation errors and auditing subsystem errors (event 4612). However, if you are planning to manually invoke " 4618(S): A monitored security event pattern has occurred ", then you also need to enable Success auditing for this subcategory. The main reason why we recommend Failure auditing for this subcategory is to be able to get Code Integrity failure events.
Member Server	Yes	Yes	Yes	Yes	The main reason why we recommend Success auditing for this subcategory is to be able to get RPC integrity violation errors and auditing subsystem errors (event 4612). However, if you are planning to manually invoke " 4618(S): A monitored security event pattern has occurred ", then you also need to enable Success auditing for this subcategory. The main reason why we recommend Failure auditing for this subcategory is to be able to get Code Integrity failure events.
Workstation	Yes	Yes	Yes	Yes	The main reason why we recommend Success auditing for this subcategory is to be able to get RPC integrity violation errors and auditing subsystem errors (event 4612). However, if you are planning to manually invoke " 4618(S): A monitored security event pattern has occurred ", then you also need to enable Success auditing for this subcategory. The main reason why we recommend Failure auditing for this subcategory is to be able to get Code Integrity failure events.

Events List:

- [4612\(S\): Internal resources allocated for the queuing of audit messages have been exhausted, leading to the loss of some audits.](#)
- [4615\(S\): Invalid use of LPC port.](#)
- [4618\(S\): A monitored security event pattern has occurred.](#)

- [4816\(S\)](#): RPC detected an integrity violation while decrypting an incoming message.
- [5038\(F\)](#): Code integrity determined that the image hash of a file is not valid. The file could be corrupt due to unauthorized modification or the invalid hash could indicate a potential disk device error.
- [5056\(S\)](#): A cryptographic self-test was performed.
- [5062\(S\)](#): A kernel-mode cryptographic self-test was performed.
- [5057\(F\)](#): A cryptographic primitive operation failed.
- [5060\(F\)](#): Verification operation failed.
- [5061\(S, F\)](#): Cryptographic operation.
- [6281\(F\)](#): Code Integrity determined that the page hashes of an image file are not valid. The file could be improperly signed without page hashes or corrupt due to unauthorized modification. The invalid hashes could indicate a potential disk device error.
- [6410\(F\)](#): Code integrity determined that a file does not meet the security requirements to load into a process.

4612(**S**): Internal resources allocated for the queuing of audit messages have been exhausted, leading to the loss of some audits.

This event is generated when audit queues are filled and events must be discarded. This most commonly occurs when security events are being generated faster than they are being written to disk.

This event doesn't generate when the event log service is stopped or event log is full and events retention is disabled.

There is no example of this event in this document.

Event Schema:

Internal resources allocated for the queuing of audit messages have been exhausted, leading to the loss of some audits.

Number of audit messages discarded: %1

This event is generated when audit queues are filled and events must be discarded. This most commonly occurs when security events are being generated faster than they are being written to disk, or when the auditing system loses connectivity to the event log, such as when the event log service is stopped.

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

[Security Monitoring Recommendations:](#)

- This event can be a sign of hardware issues or lack of system resources (for example, RAM). We recommend monitoring this event and investigating the reason for the condition.

4615(**S**): Invalid use of LPC port.

It appears that this event never occurs.

Event Schema:

Invalid use of LPC port.

Subject:

Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4

Process Information:

PID:%7
Name:%8

Invalid Use:%5

LPC Server Port Name:%6

Windows Local Security Authority (LSA) communicates with the Windows kernel using Local Procedure Call (LPC) ports. If you see this event, an application has inadvertently or intentionally accessed this port which is reserved exclusively for LSA's use. The application (process) should be investigated to ensure that it is not attempting to tamper with this communications channel."

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- There is no recommendation for this event in this document.

4618(S): A monitored security event pattern has occurred.

This event can be generated (invoked) only externally using the following command:

%windir%\system32\rundll32 %windir%\system32\authz.dll,AuthzGenerateAdminAudit Audit OrgEventId ComputerName UserSid UserName UserDomain UserLogonId EventCount

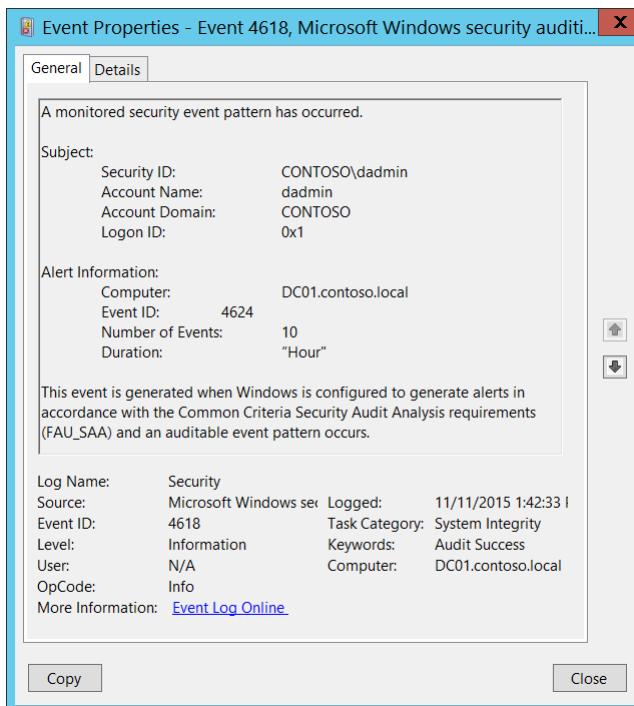
Duration

Account must have **SeAuditPrivilege** (Generate security audits) to be able to generate this event.

- **UserSid** is resolved when viewing the event in event viewer.
- Only **OrgEventID**, **ComputerName**, and **EventCount** are required—others are optional. Fields not specified appear with “-” in the event description field.
- If a field doesn't match the expected data type, the event is not generated. (i.e., if **EventCount** = “XYZ” then no event is generated.)
- **UserSid**, **UserName**, and **UserDomain** are not related to each other (think **SubjectUser** fields, where they are)
- Parameters are space delimited, even if a parameter is enclosed in double-quotes.
- Here are the expected data types for the parameters:

Parameter	Expected Data Type
OrgEventID	ULong
ComputerName	String

UserSid	SID (in string format)
UserName	String
UserDomain	String
UserLogonID	Luid (a UInt64 converted to Hex in the event)
EventCount	UInt64
Duration	String



Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>4618</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12290</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-11T21:42:33.264246700Z" />
<EventRecordID>1198759</EventRecordID>
<Correlation />
<Execution ProcessID="500" ThreadID="528" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="EventId">4624</Data>
<Data Name="ComputerName">DC01.contoso.local</Data>
<Data Name="TargetUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>

```

```

<Data Name="TargetUserName">admind</Data>
<Data Name="TargetUserDomain">CONTOSO</Data>
<Data Name="TargetLogonId">0x1</Data>
<Data Name="EventCount">10</Data>
<Data Name="Duration">"Hour"</Data>
</EventData>
</Event>

```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

For 4618(S): A monitored security event pattern has occurred.

- This event can be invoked only manually/intentionally, it is up to you how interpret this event depends on information you put inside of it.

4816(S): RPC detected an integrity violation while decrypting an incoming message.

This message generates if RPC detected an integrity violation while decrypting an incoming message.

There is no example of this event in this document.

Event Schema:

RPC detected an integrity violation while decrypting an incoming message.

Peer Name: %1

Protocol Sequence: %2

Security Error: %3

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- We recommend monitoring for this event, especially on high value assets or computers, because it can be a sign of a software or configuration issue, or a malicious action.

5038(F): Code integrity determined that the image hash of a file is not valid. The file could be corrupt due to unauthorized modification or the invalid hash could indicate a potential disk device error.

This event generates by [Code Integrity](#) feature, if signature of a file is not valid.

Code Integrity is a feature that improves the security of the operating system by validating the integrity of a driver or system file each time it is loaded into memory. Code Integrity detects whether an unsigned driver or system file is being loaded into the kernel, or whether a system file has been modified by malicious software that is being run by a user account with administrative permissions. On x64-based versions of the operating system, kernel-mode drivers must be digitally signed.

There is no example of this event in this document.

Event Schema:

Code integrity determined that the image hash of a file is not valid. The file could be corrupt due to unauthorized modification or the invalid hash could indicate a potential disk device error.

File Name: %filepath\filename%

Security Monitoring Recommendations:

- We recommend monitoring for this event, especially on high value assets or computers, because it can be a sign of a software or configuration issue, or a malicious action.

5056(S): A cryptographic self-test was performed.

This event generates in CNG Self-Test function. This is a Cryptographic Next Generation (CNG) function.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- [https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic self test was performed.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Module:%5

Return Code:%6

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related actions with cryptographic keys. If you need to monitor or troubleshoot actions related to specific cryptographic keys and operations, review this event to see if it provides the information you need.

5062(S): A kernel-mode cryptographic self-test was performed.

This event occurs rarely, and in some situations may be difficult to reproduce.

Event Schema:

A kernel-mode cryptographic self test was performed.

Module:%1

Return Code:%2

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related actions with cryptographic keys. If you need to monitor or troubleshoot actions related to specific cryptographic keys and operations, review this event to see if it provides the information you need.

5057(F): A cryptographic primitive operation failed.

This event generates in case of CNG primitive operation failure.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- [https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

A cryptographic primitive operation failed.

Subject:

Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4

Cryptographic Parameters:

Provider Name:%5
Algorithm Name%6

Failure Information:

Reason:%7
Return Code:%8

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related actions with cryptographic keys. If you need to monitor or troubleshoot actions related to specific cryptographic keys and operations, review this event to see if it provides the information you need.

5060(F): Verification operation failed.

This event generates in case of CNG verification operation failure.

For more information about Cryptographic Next Generation (CNG) visit these pages:

- [https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/aa376214(v=vs.85).aspx)
- [https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/bb204775(v=vs.85).aspx)
- <http://www.microsoft.com/en-us/download/details.aspx?id=1251>
- <http://www.microsoft.com/en-us/download/details.aspx?id=30688>

This event is mainly used for Cryptographic Next Generation (CNG) troubleshooting.

There is no example of this event in this document.

Event Schema:

Verification operation failed.

Subject:

*Security ID:%1
Account Name:%2
Account Domain:%3
Logon ID:%4*

Cryptographic Parameters:

*Provider Name:%5
Algorithm Name:%6
Key Name:%7
Key Type:%8*

Failure Information:

*Reason:%7
Return Code:%8*

Required Server Roles: None.

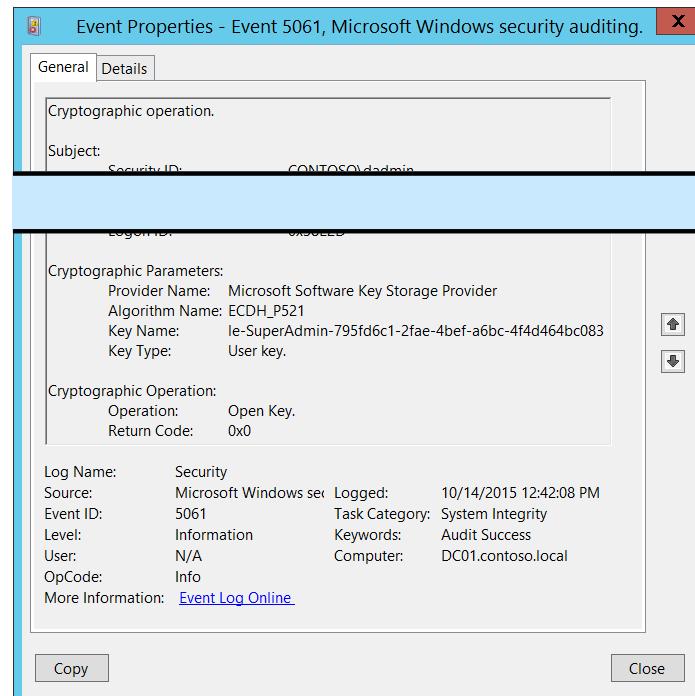
Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- Typically this event is required for detailed monitoring of CNG-related actions with cryptographic keys. If you need to monitor or troubleshoot actions related to specific cryptographic keys and operations, review this event to see if it provides the information you need.

5061(S, F): Cryptographic operation.



Event Description:

This event generates when a cryptographic operation (open key, create key, create key, and so on) was performed using a [Key Storage Provider](#) (KSP). This event generates only if one of the following KSPs were used:

- Microsoft Software Key Storage Provider
- Microsoft Smart Card Key Storage Provider

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
<EventID>5061</EventID>
<Version>0</Version>
<Level>0</Level>
<Task>12290</Task>
<Opcode>0</Opcode>
<Keywords>0x8020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-14T19:42:08.104008000Z" />
<EventRecordID>1048444</EventRecordID>
<Correlation />
<Execution ProcessID="520" ThreadID="3496" />
```

```
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <EventData>
<Data Name="SubjectUserId">S-1-5-21-3457937927-2839227994-823803824-1104</Data>
<Data Name="SubjectUserName">dadmind</Data>
<Data Name="SubjectDomainName">CONTOSO</Data>
<Data Name="SubjectLogonId">0x38e2d</Data>
<Data Name="ProviderName">Microsoft Software Key Storage Provider</Data>
<Data Name="AlgorithmName">ECDH_P521</Data>
```

```
<Data Name="KeyName">le-SuperAdmin-795fd6c1-2fae-4bef-a6bc-4f4d464bc083</Data>
<Data Name="KeyType">%%2500</Data>
<Data Name="Operation">%%2480</Data>
<Data Name="ReturnCode">0x0</Data>
</EventData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that requested specific cryptographic operation. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

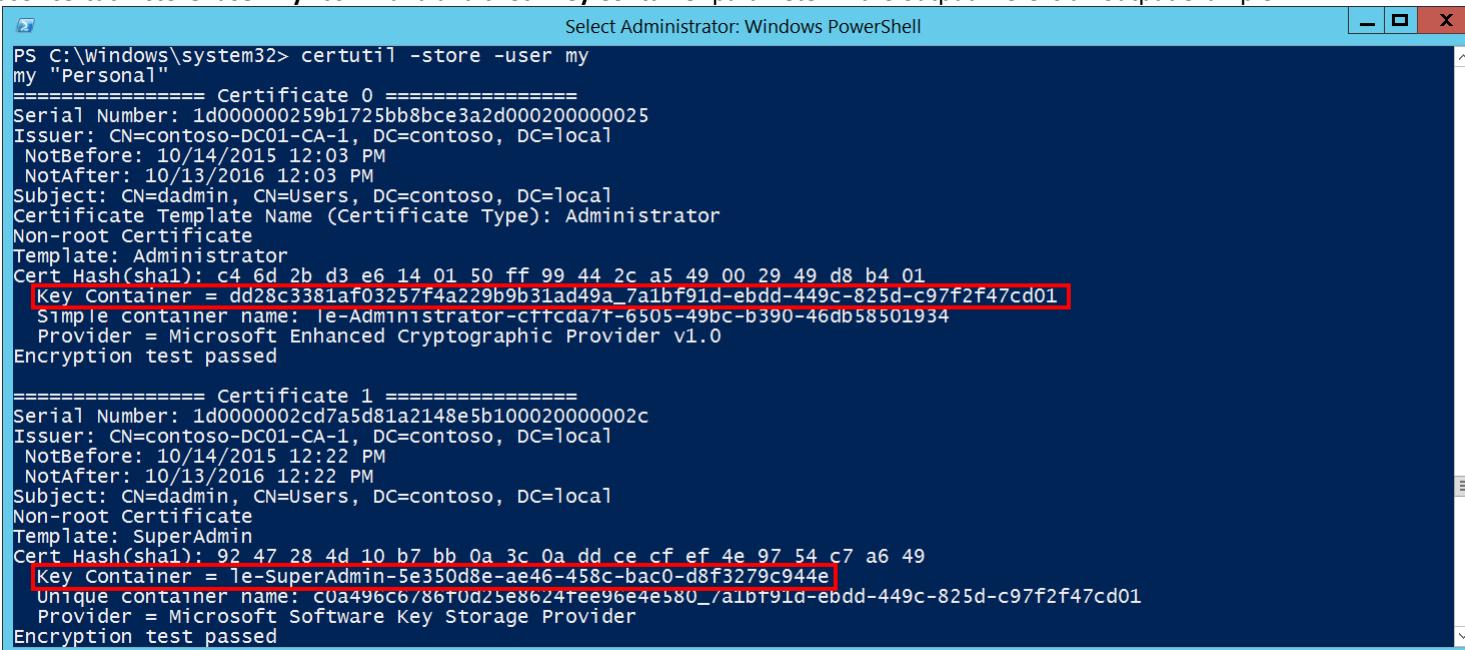
A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that requested specific cryptographic operation.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".
- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Cryptographic Parameters:

- **Provider Name** [Type = UnicodeString]: the name of KSP through which the operation was performed. Can have one of the following values:
 - Microsoft Software Key Storage Provider
 - Microsoft Smart Card Key Storage Provider
- **Algorithm Name** [Type = UnicodeString]: the name of cryptographic algorithm through which the key was used or accessed. For "Read persisted key from file" operation, this typically has "UNKNOWN" value. Can also have one of the following values:
 - RSA – algorithm created by Ron Rivest, Adi Shamir, and Leonard Adleman.
 - DSA – Digital Signature Algorithm.
 - DH – Diffie-Hellman.
 - ECDH_P521 – Elliptic Curve Diffie-Hellman algorithm with 512-bit key length.
 - ECDH_P384 – Elliptic Curve Diffie-Hellman algorithm with 384-bit key length.

- ECDH_P256 – Elliptic Curve Diffie-Hellman algorithm with 256-bit key length.
 - ECDSA_P256 – Elliptic Curve Digital Signature Algorithm with 256-bit key length.
 - ECDSA_P384 – Elliptic Curve Digital Signature Algorithm with 384-bit key length.
 - ECDSA_P521 – Elliptic Curve Digital Signature Algorithm with 521-bit key length.
- **Key Name** [Type = UnicodeString]: the name of the key (key container) with which operation was performed. For example, to get the list of **Key Names** for certificates for logged in user you can use “**certutil -store -user my**” command and check **Key Container** parameter in the output. Here is an output example:



```

Select Administrator: Windows PowerShell
PS C:\Windows\system32> certutil -store -user my
my "Personal"
=====
Certificate 0 =====
Serial Number: 1d000000259b1725bb8bce3a2d000200000025
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:03 PM
NotAfter: 10/13/2016 12:03 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Certificate Template Name (Certificate Type): Administrator
Non-root Certificate
Template: Administrator
Cert Hash(sha1): c4 6d 2b d3 e6 14 01 50 ff 99 44 2c a5 49 00 29 49 d8 b4 01
Key Container = dd28c3381af03257f4a229b9b31ad49a_7a1bf91d-ebdd-449c-825d-c97f2f47cd01
Simple container name: 1e-Administrator-ctfcda7f-6505-49bc-b390-46db58501934
Provider = Microsoft Enhanced Cryptographic Provider v1.0
Encryption test passed

=====
Certificate 1 =====
Serial Number: 1d0000002cd7a5d81a2148e5b100020000002c
Issuer: CN=contoso-DC01-CA-1, DC=contoso, DC=local
NotBefore: 10/14/2015 12:22 PM
NotAfter: 10/13/2016 12:22 PM
Subject: CN=dadmin, CN=Users, DC=contoso, DC=local
Non-root Certificate
Template: SuperAdmin
Cert Hash(sha1): 92 47 28 4d 10 b7 bb 0a 3c 0a dd ce cf ef 4e 97 54 c7 a6 49
Key Container = 1e-SuperAdmin-5e350d8e-ae46-458c-bac0-d8f3279c944e
Unique container name: c0a496c6/86t0d25e8624feef96e4e580/_a1bt91d-ebdd-449c-825d-c97f2f47cd01
Provider = Microsoft Software Key Storage Provider
Encryption test passed

```

- **Key Type** [Type = UnicodeString]: can have one of the following values:
 - “User key.” – user’s cryptographic key.
 - “Machine key.” – machine’s cryptographic key.

Cryptographic Operation:

- **Operation** [Type = UnicodeString]: performed operation. Possible values:
 - Open Key. – open existing cryptographic key.
 - Create Key. – create new cryptographic key.
 - Delete Key. – delete existing cryptographic key.
 - Sign hash. – cryptographic signing operation.
 - Secret agreement.
 - Key Derivation. – key derivation operation.
 - Encrypt. – encryption operation.

- Decrypt. – decryption operation.
- **Return Code** [Type = HexInt32]: has “**0x0**” value for Success events. For failure events, provides a hexadecimal error code number.

Security Monitoring Recommendations:

For 5061(S, F): Cryptographic operation.

- Typically this event is required for detailed monitoring of KSP-related actions with cryptographic keys. If you need to monitor actions related to specific cryptographic keys (“**Key Name**”) or a specific “**Operation**”, such as “**Delete Key**”, create monitoring rules and use this event as an information source.

[Appendix A: Security monitoring recommendations for many audit events](#)

Important For this event, also see Appendix A: Security monitoring recommendations for many audit events.

6281(F): Code Integrity determined that the page hashes of an image file are not valid. The file could be improperly signed without page hashes or corrupt due to unauthorized modification. The invalid hashes could indicate a potential disk device error.

[Code Integrity](#) is a feature that improves the security of the operating system by validating the integrity of a driver or system file each time it is loaded into memory. Code Integrity detects whether an unsigned driver or system file is being loaded into the kernel, or whether a system file has been modified by malicious software that is being run by a user account with administrative permissions. On x64-based versions of the operating system, kernel-mode drivers must be digitally signed.

This event generates when [code integrity](#) determined that the page hashes of an image file are not valid. The file could be improperly signed without page hashes or corrupt due to unauthorized modification. This event also generates when signing certificate was revoked. The invalid hashes could indicate a potential disk device error.

There is no example of this event in this document.

Event Schema:

Code Integrity determined that the page hashes of an image file are not valid. The file could be improperly signed without page hashes or corrupt due to unauthorized modification. The invalid hashes could indicate a potential disk device error.

File Name:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Security Monitoring Recommendations:

- We recommend monitoring for this event, especially on high value assets or computers, because it can be a sign of a software or configuration issue, or a malicious action.

6410(F): Code integrity determined that a file does not meet the security requirements to load into a process.

[Code Integrity](#) is a feature that improves the security of the operating system by validating the integrity of a driver or system file each time it is loaded into memory. Code Integrity detects whether an unsigned driver or system file is being loaded into the kernel, or whether a system file has been modified by malicious software that is being run by a user account with administrative permissions. On x64-based versions of the operating system, kernel-mode drivers must be digitally signed.

This event generates due to writable [shared sections](#) being present in a file image.

There is no example of this event in this document.

Event Schema:

Code integrity determined that a file does not meet the security requirements to load into a process. This could be due to the use of shared sections or other issues.

File Name:%1

Required Server Roles: None.

Minimum OS Version: Windows Server 2012 R2, Windows 8.1.

Event Versions: 0.

Security Monitoring Recommendations:

- We recommend monitoring for this event, especially on high value assets or computers, because it can be a sign of a software or configuration issue, or a malicious action.

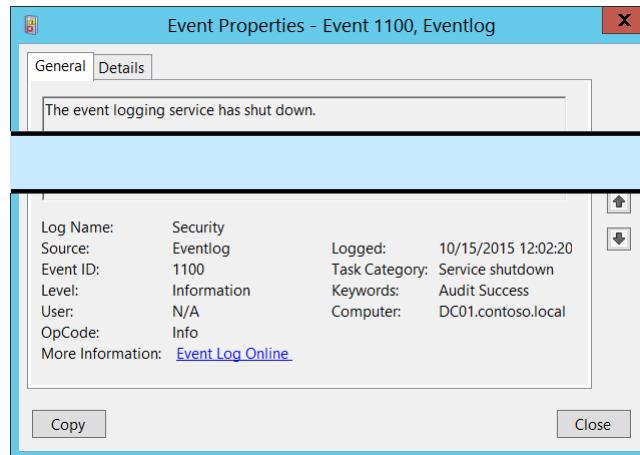
Other Events

Events in this section generate automatically and are enabled by default.

Events List:

- [1100\(S\)](#): The event logging service has shut down.
- [1102\(S\)](#): The audit log was cleared.
- [1104\(S\)](#): The security log is now full.
- [1105\(S\)](#): Event log automatic backup.
- [1108\(S\)](#): The event logging service encountered an error while processing an incoming event published from %1

[1100\(S\)](#): The event logging service has shut down.



The event logging service has shut down.

Event Description:

This event generates every time Windows Event Log service has shut down.
It also generates during normal system shutdown.
This event doesn't generate during emergency system reset.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Log Name: Security
Source: Eventlog
Event ID: 1100
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online](#)

Logged: 10/15/2015 12:02:20
Task Category: Service shutdown
Keywords: Audit Success
Computer: DC01.contoso.local

Copy **Close**

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Eventlog" Guid="{fc65ddd8-d6ef-4962-83d5-6e5cfe9ce148}" />
<EventID>1100</EventID>
<Version>0</Version>
<Level>4</Level>
<Task>103</Task>

<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-15T07:02:20.010585400Z" />
<EventRecordID>1048124</EventRecordID>
<Correlation />
<Execution ProcessID="820" ThreadID="964" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <UserData>
<ServiceShutdown xmlns="http://manifests.microsoft.com/win/2004/08/windows/eventlog" />
```

```

<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-15T07:02:20.010585400Z" />
<EventRecordID>1048124</EventRecordID>
<Correlation />
<Execution ProcessID="820" ThreadID="964" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <UserData>
<ServiceShutdown xmlns="http://manifests.microsoft.com/win/2004/08/windows/eventlog" />
```

</UserData>
</Event>

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

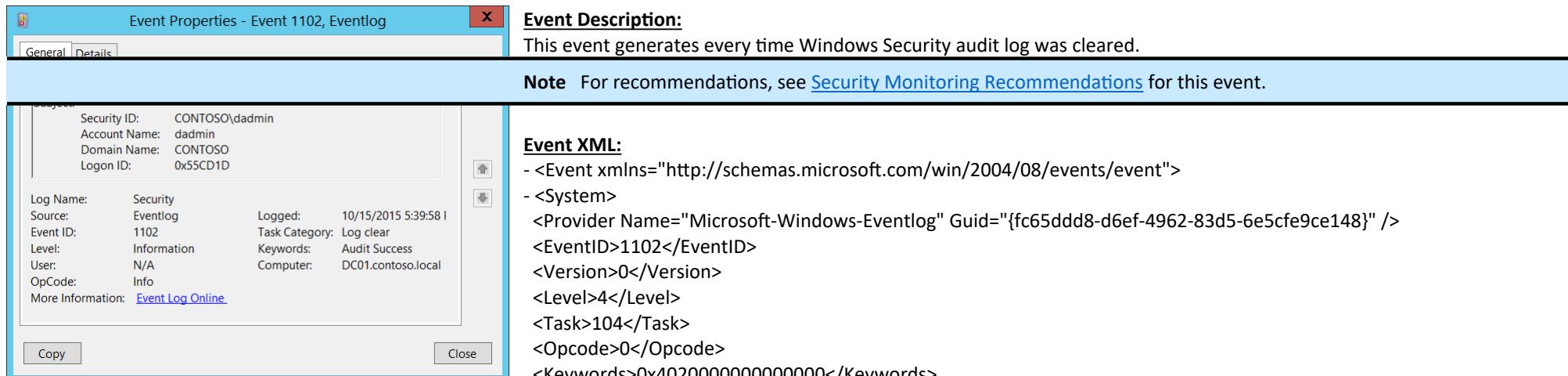
Event Versions: 0.

Security Monitoring Recommendations:

For 1100(S): The event logging service has shut down.

- With this event, you can track system shutdowns and restarts.
- This event also can be a sign of malicious action when someone tried to shut down the Log Service to cover his or her activity.

1102(S): The audit log was cleared.



Event Description:
This event generates every time Windows Security audit log was cleared.

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Eventlog" Guid="{fc65ddd8-d6ef-4962-83d5-6e5cfe9ce148}" />
<EventID>1102</EventID>
<Version>0</Version>
<Level>4</Level>
<Task>104</Task>
<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>

```

```

<TimeCreated SystemTime="2015-10-16T00:39:58.656871200Z" />
<EventRecordID>1087729</EventRecordID>
<Correlation />
<Execution ProcessID="820" ThreadID="2644" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <UserData>
- <LogFileCleared xmlns="http://schemas.microsoft.com/win/2004/08/windows/eventlog">
<SubjectUserSid>S-1-5-21-3457937927-2839227994-823803824-1104</SubjectUserSid>
<SubjectUserName>dadmin</SubjectUserName>

```

```
<SubjectDomainName>CONTOSO</SubjectDomainName>
<SubjectLogonId>0x55cd1d</SubjectLogonId>
</LogFileCleared>
</UserData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Subject:

- **Security ID** [Type = SID]: SID of account that cleared the system security audit log. Event Viewer automatically tries to resolve SIDs and show the account name. If the SID cannot be resolved, you will see the source data in the event.

A **security identifier (SID)** is a unique value of variable length used to identify a trustee (security principal). Each account has a unique SID that is issued by an authority, such as an Active Directory domain controller, and stored in a security database. Each time a user logs on, the system retrieves the SID for that user from the database and places it in the access token for that user. The system uses the SID in the access token to identify the user in all subsequent interactions with Windows security. When a SID has been used as the unique identifier for a user or group, it cannot ever be used again to identify another user or group. For more information about SIDs, see [Security Identifiers](#).

- **Account Name** [Type = UnicodeString]: the name of the account that cleared the system security audit log.
- **Account Domain** [Type = UnicodeString]: subject's domain or computer name. Formats vary, and include the following:
 - Domain NETBIOS name example: CONTOSO
 - Lowercase full domain name: contoso.local
 - Uppercase full domain name: CONTOSO.LOCAL
 - For some [well-known security principals](#), such as LOCAL SERVICE or ANONYMOUS LOGON, the value of this field is "NT AUTHORITY".
 - For local user accounts, this field will contain the name of the computer or device that this account belongs to, for example: "Win81".

- **Logon ID** [Type = HexInt64]: hexadecimal value that can help you correlate this event with recent events that might contain the same Logon ID, for example, "[4624](#): An account was successfully logged on."

Security Monitoring Recommendations:

For 1102(S): The audit log was cleared.

[Appendix A: Security monitoring recommendations for many audit events](#)

- **Important** For this event, also see [Appendix A: Security monitoring recommendations for many audit events](#). Typically you should not see this event. There is no need to manually clear the Security event log in most cases. We recommend monitoring this event and investigating why this action was performed.

1104(S): The security log is now full.

Event Description:

This event generates every time Windows security log becomes full.

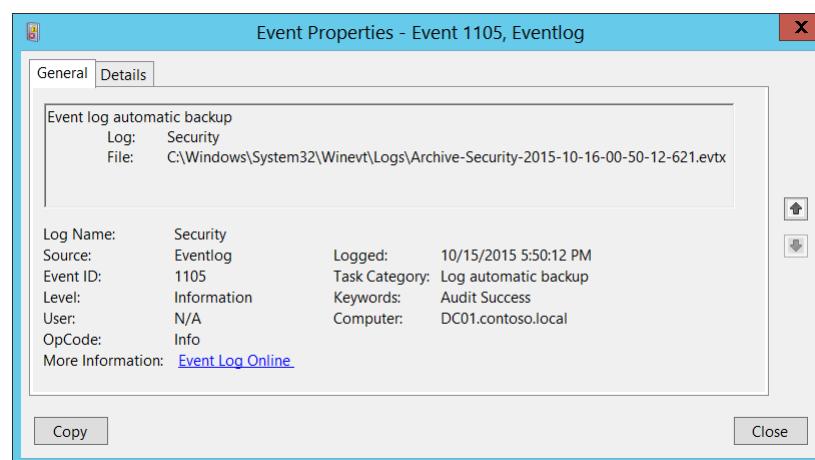
© 2016 Microsoft. All rights reserved.

This event generates, for example, if the maximum size of Security Event Log file was reached and event log retention method is: "[Do not overwrite events \(Clear logs manually\)](#)".

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Eventlog" Guid="{fc65ddd8-d6ef-4962-83d5-6e5fce9ce148}" />
<EventID>1104</EventID>
<Version>0</Version>
<Level>2</Level>
<Task>101</Task>
<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-15T23:36:50.479431200Z" />
<EventRecordID>1087728</EventRecordID>
<Correlation />
<Execution ProcessID="820" ThreadID="4224" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <UserData>
<FileIsFull xmlns="http://manifests.microsoft.com/win/2004/08/windows/eventlog" />
</UserData>
</Event>
```



Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Security Monitoring Recommendations:

- If the Security event log retention method is set to "[Do not overwrite events \(Clear logs manually\)](#)", then this event will indicate that log file is full and you need to perform immediate actions, for example, archive the log or clear it.

1105(S): Event log automatic backup.

Event Description:

This event generates every time Windows security log becomes full and new event log file was created.

This event generates, for example, if the maximum size of Security Event Log file was reached and event log retention method is: "[Archive the log when full, do not overwrite events](#)".

Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
- <System>
<Provider Name="Microsoft-Windows-Eventlog" Guid="{fc65ddd8-d6ef-4962-83d5-6e5fce9ce148}" />
<EventID>1105</EventID>
<Version>0</Version>
<Level>4</Level>
<Task>105</Task>
<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>
<TimeCreated SystemTime="2015-10-16T00:50:12.715302700Z" />
<EventRecordID>1128551</EventRecordID>
<Correlation />
<Execution ProcessID="820" ThreadID="3660" />
<Channel>Security</Channel>
<Computer>DC01.contoso.local</Computer>
<Security />
</System>
- <UserData>
- <AutoBackup xmlns="http://manifests.microsoft.com/win/2004/08/windows/eventlog">
<Channel>Security</Channel>
<BackupPath>C:\Windows\System32\Winevt\Logs\Archive-Security-2015-10-16-00-50-12-621.evtx</BackupPath>
</AutoBackup>
</UserData>
</Event>
```

Required Server Roles: None.

Minimum OS Version: Windows Server 2008, Windows Vista.

Event Versions: 0.

Field Descriptions:

Log [Type = UnicodeString]: the name of the log which was archived (new event log file was created and previous event log was archived). Always "**Security**" for Security Event Logs.

File: [Type = FILETIME]: full path and filename of archived log file.

The format of archived log file name is: "Archive-**LOG_FILE_NAME**-YYYY-MM-DD-hh-mm-ss-nnn.evtx". Where:

- LOG_FILE_NAME – the name of archived file.
- Y – years.

- M – months.
- D – days.
- h – hours.
- m – minutes.
- s – seconds.
- n – fractional seconds.

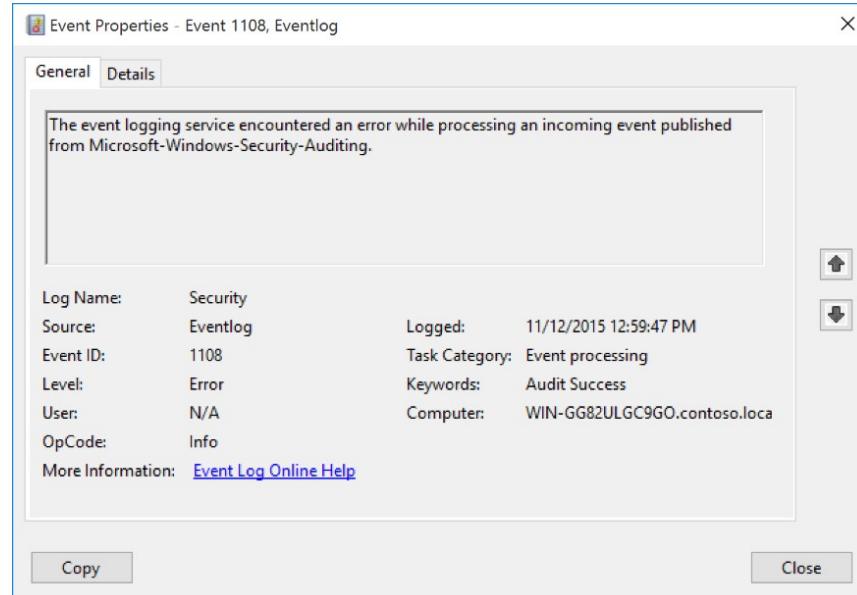
The time in this event is always in [GMT+0/UTC+0](#) time zone.

Security Monitoring Recommendations:

For 1105(S): Event log automatic backup.

- Typically it's an informational event and no actions are needed. But if your baseline settings are not set to [Archive the log when full, do not overwrite events](#), then this event will be a sign that some settings are not set to baseline settings or were changed.

1108(S): The event logging service encountered an error while processing an incoming event published from %1.

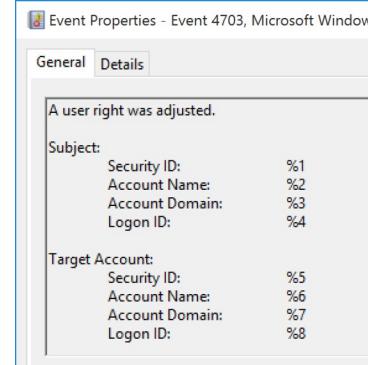


Event Description:

This event generates when event logging service encountered an error while processing an incoming event.

It typically generates when logging service will not be able to correctly write the event to the event log or some parameters were not passed to logging service to log the event correctly. You will typically see a defective or incorrect event before 1108.

For example, event 1108 might be generated after an incorrect [4703](#) event:



Note For recommendations, see [Security Monitoring Recommendations](#) for this event.

Event XML:

- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">

```
- <System>
<Provider Name="Microsoft-Windows-Eventlog" Guid="{fc65ddd8-d6ef-4962-83d5-6e5cfe9ce148}" />
<EventID>1108</EventID>
<Version>0</Version>
<Level>2</Level>
<Task>101</Task>
<Opcode>0</Opcode>
<Keywords>0x4020000000000000</Keywords>
<TimeCreated SystemTime="2015-11-12T20:59:47.431979300Z" />
<EventRecordID>5599</EventRecordID>
<Correlation />
<Execution ProcessID="972" ThreadID="1320" />
<Channel>Security</Channel>
<Computer>WIN-GG82ULGC9GO.contoso.local</Computer>
<Security />
</System>
- <UserData>
- <EventProcessingFailure xmlns="http://manifests.microsoft.com/win/2004/08/windows/eventlog">
<Error Code="15005" />
<EventID>0</EventID>
<PublisherID>Microsoft-Windows-Security-Auditing</PublisherID>
</EventProcessingFailure>
</UserData>
</Event>
```

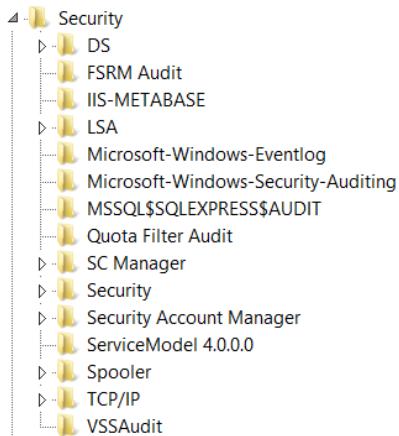
Required Server Roles: None.

Minimum OS Version: Windows Server 2008 R2, Windows 7.

Event Versions: 0.

Field Descriptions:

%1 [Type = UnicodeString]: the name of [security event source](#) from which event was received for processing. You can see all registered security event source names in this registry path: "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Security". Here is an example:



Security Monitoring Recommendations:

For 1108(S): The event logging service encountered an error while processing an incoming event published from %1.

- We recommend monitoring for all events of this type and checking what the cause of the error was.

Appendix A: Security monitoring recommendations for many audit events

This document provides reference information about individual audit events, and lists them within audit categories and subcategories. However, there are many events for which the following overall recommendations apply. There are links throughout this document from the “Recommendations” sections of the relevant events to this appendix.

Type of monitoring required	Recommendation
High-value accounts: You might have high-value domain or local accounts for which you need to monitor each action. Examples of high-value accounts are database administrators, built-in local administrator account, domain administrators, service accounts, domain controller accounts and so on.	Monitor relevant events for the “ Subject\Security ID ” that corresponds to the high-value account or accounts.
Anomalies or malicious actions: You might have specific requirements for detecting anomalies or monitoring potential malicious actions. For example, you might need to monitor for use of an account outside of working hours.	When you monitor for anomalies or malicious actions, use the “ Subject\Security ID ” (with other information) to monitor how or when a particular account is being used.
Non-active accounts: You might have non-active, disabled, or guest accounts, or other accounts that should never be used.	Monitor relevant events for the “ Subject\Security ID ” that corresponds to the accounts that should never be used.
Account whitelist: You might have a specific whitelist of accounts that are the only ones allowed to perform actions corresponding to particular events.	Monitor the relevant events for “ Subject\Security ID ” accounts that are outside the whitelist of accounts.
Accounts of different types: You might want to ensure that certain actions are performed only by certain account types, for example, local or domain account, machine or user account, vendor or employee account, and so on.	Identify events that correspond to the actions you want to monitor, and for those events, review the “ Subject\Security ID ” to see whether the account type is as expected.
External accounts: You might be monitoring accounts from another domain, or “external” accounts that are not allowed to perform certain actions (represented by certain specific events).	Monitor the specific events for the “ Subject\Account Domain ” corresponding to accounts from another domain or “external” accounts.
Restricted-use computers or devices: You might have certain computers, machines, or devices on which certain people (accounts) should not typically perform any actions.	Monitor the target Computer : (or other target device) for actions performed by the “ Subject\Security ID ” that you are concerned about.
Account naming conventions: Your organization might have specific naming conventions for account names.	Monitor “ Subject\Account Name ” for names that don’t comply with naming conventions.

Appendix B: List of Tables

Table 1. Winlogon Error Codes.....	5
Table 2. Kerberos ticket flags.....	11
Table 3. TGT/TGS issue error codes.....	16
Table 4. Kerberos encryption types.....	16
Table 5. Kerberos Pre-Authentication types.....	17
Table 6. Kerberos ticket flags.....	23
Table 7. User's or Computer's account UAC flags.....	51
Table 8. User Privileges.....	55
Table 9. Active Directory Access Codes and Rights.....	233
Table 10. LDAP Attribute Syntax OIDs.....	247
Table 11. Windows Logon Types.....	271
Table 12. Windows logon status codes.....	272
Table 13. File access codes.....	348
Table 14. File System objects access rights.....	375