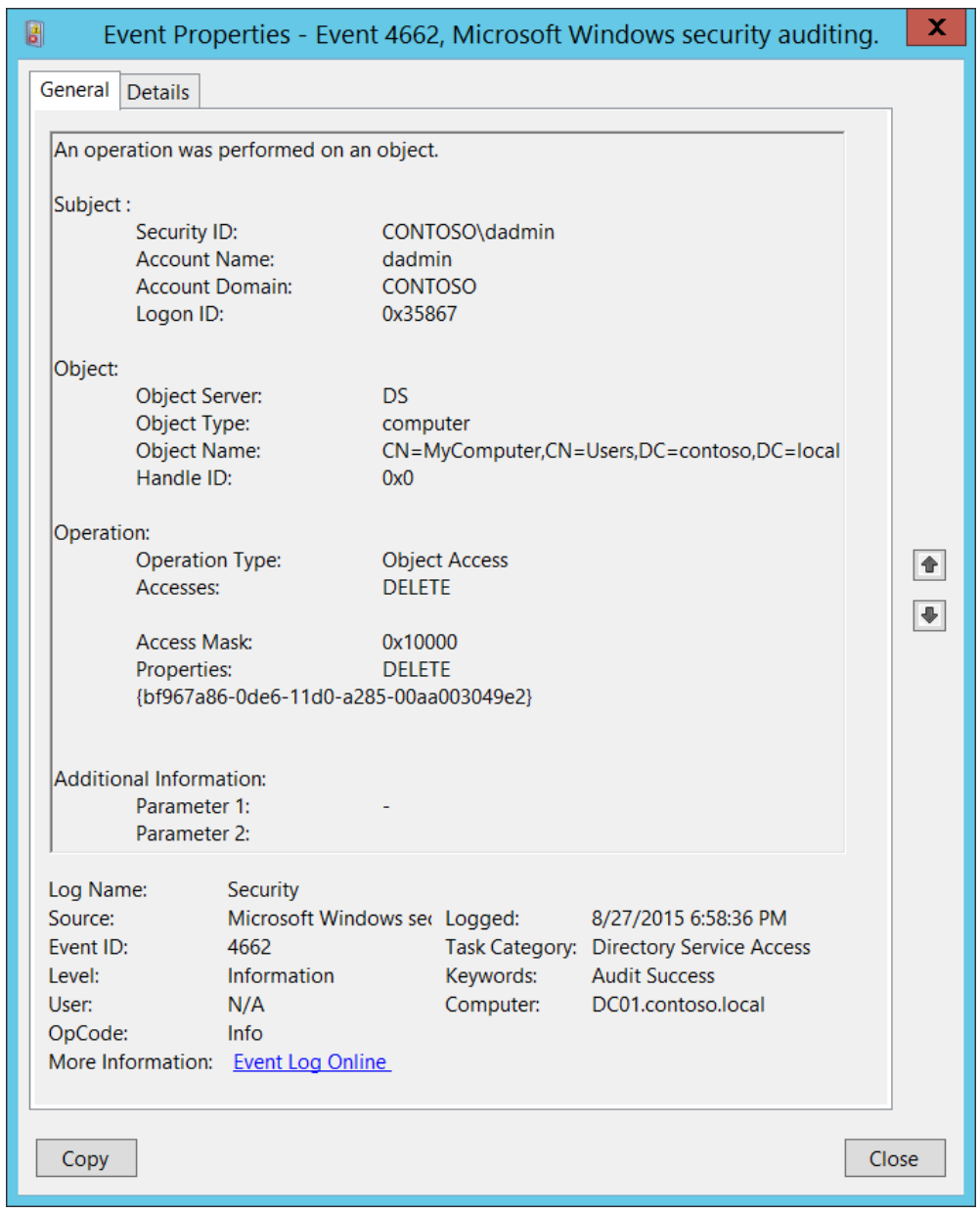




# 4662(S, F): An operation was performed on an object.

Article • 09/07/2021 • 1 contributor




***Subcategory:*** Audit Directory Service Access

***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:***

- Audit Other Privilege Use Events
- Audit IPsec Driver
- Audit Other System Events

Copy

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

> **Note** 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/library/Cc755885(v=WS.10).aspx) and navigate to **Active Directory Schema\Classes**. Or use this document: https://msdn.microsoft.com/library/cc221630.aspx
- **Object Name** [Type = UnicodeString]: distinguished name of the object that was accessed.

> **Note** 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.

Expand table

| 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 is 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.<br>The right to perform an operation controlled by an extended access right. |
| 0x10000 | DELETE | The right to delete the object.<br>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_SYSTEM_SECURITY | The right to get or set the SACL in the object security descriptor. |
| 0x80000000 | ADS_RIGHT_GENERIC_READ | 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_WRITE | 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_EXECUTE | The right to read permissions on, and list the contents of, a container object. |

| | | |
|---|---|---|
| 0x10000000 | ADS_RIGHT_GENERIC_ALL | 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.

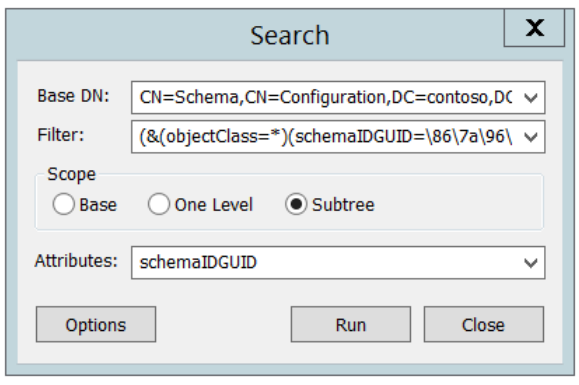
- **Properties** [Type = UnicodeString]: first part is the type of access that was used. Typically has the same value as **Accesses** field.

  Second part is a tree of **GUID** values of Active Directory classes or property sets, for which operation was performed.

**Note** **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\a2\85\00\aa\00\30\49\e2
      - Filter example: (&(objectClass=*)(schemaIDGUID=\86\7a\96\bf\e6\0d\d0\11\a2\85\00\aa\00\30\49\e2))
  - 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/library/ms683990(v=vs.85).aspx.

Here is an example of decoding of **Properties** field:

Expand table

| Properties | Translation |
| --- | --- |
| {bf967a86-0de6-11d0-a285-00aa003049e2} | Computer |
| {91e647de-d96f-4b70-9557-d63ff4f3ccd8} | Private-Information property set |
| {6617e4ac-a2f1-43ab-b60c-11fbd1facf05} | 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.

> **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.
- 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.