



# **SMB events that can be audited**

## **ONTAP 9**

NetApp  
August 22, 2022

# Table of Contents

- SMB events that can be audited ..... 1
  - SMB events that can be audited overview ..... 1
  - Determine what the complete path to the audited object is ..... 3
  - Considerations when auditing symlinks and hard links ..... 4
  - Considerations when auditing alternate NTFS data streams ..... 5

# SMB events that can be audited

## SMB events that can be audited overview

ONTAP can audit certain SMB events, including certain file and folder access events, certain logon and logoff events, and central access policy staging events. Knowing which access events can be audited is helpful when interpreting results from the event logs.

The following additional SMB events can be audited in ONTAP 9.2 and later:

| Event ID (EVT/EVTX) | Event                                    | Description                            | Category    |
|---------------------|--|--|-------------|
| 4670                | Object permissions were changed          | OBJECT ACCESS: Permissions changed.    | File Access |
| 4907                | Object auditing settings were changed    | OBJECT ACCESS: Audit settings changed. | File Access |
| 4913                | Object Central Access Policy was changed | OBJECT ACCESS: CAP changed.            | File Access |

The following SMB events can be audited in ONTAP 9.0 and later:

| Event ID (EVT/EVTX) | Event                                 | Description  | Category         |
|---------------------|---------------------------------------|--|------------------|
| 540/4624            | An account was successfully logged on | LOGON/LOGOFF: Network (SMB) logon.                 | Logon and Logoff |
| 529/4625            | An account failed to log on           | LOGON/LOGOFF: Unknown user name or bad password.   | Logon and Logoff |
| 530/4625            | An account failed to log on           | LOGON/LOGOFF: Account logon time restriction.      | Logon and Logoff |
| 531/4625            | An account failed to log on           | LOGON/LOGOFF: Account currently disabled.          | Logon and Logoff |
| 532/4625            | An account failed to log on           | LOGON/LOGOFF: User account has expired.            | Logon and Logoff |
| 533/4625            | An account failed to log on           | LOGON/LOGOFF: User cannot log on to this computer. | Logon and Logoff |
| 534/4625            | An account failed to log on           | LOGON/LOGOFF: User not granted logon type here.    | Logon and Logoff |

|          |  |   |                  |
|----------|--|---|------------------|
| 535/4625 | An account failed to log on  | LOGON/LOGOFF: User's password has expired.  | Logon and Logoff |
| 537/4625 | An account failed to log on  | LOGON/LOGOFF: Logon failed for reasons other than above.  | Logon and Logoff |
| 539/4625 | An account failed to log on  | LOGON/LOGOFF: Account locked out.   | Logon and Logoff |
| 538/4634 | An account was logged off  | LOGON/LOGOFF: Local or network user logoff.   | Logon and Logoff |
| 560/4656 | Open Object/Create Object  | OBJECT ACCESS: Object (file or directory) open.   | File Access      |
| 563/4659 | Open Object with the Intent to Delete  | OBJECT ACCESS: A handle to an object (file or directory) was requested with the Intent to Delete.   | File Access      |
| 564/4660 | Delete Object  | OBJECT ACCESS: Delete Object (file or directory). ONTAP generates this event when a Windows client attempts to delete the object (file or directory).   | File Access      |
| 567/4663 | Read Object/Write Object/Get Object Attributes/Set Object Attributes   | <p>OBJECT ACCESS: Object access attempt (read, write, get attribute, set attribute).</p> <p><b>Note:</b> For this event, ONTAP audits only the first SMB read and first SMB write operation (success or failure) on an object. This prevents ONTAP from creating excessive log entries when a single client opens an object and performs many successive read or write operations to the same object.</p> | File Access      |
| NA/4664  | Hard link  | OBJECT ACCESS: An attempt was made to create a hard link.   | File Access      |
| NA/4818  | Proposed central access policy does not grant the same access permissions as the current central access policy | OBJECT ACCESS: Central Access Policy Staging.   | File Access      |

|                                   |               |  |             |
|-----------------------------------|---------------|--|-------------|
| NA/NA Data ONTAP<br>Event ID 9999 | Rename Object | OBJECT ACCESS: Object renamed.<br>This is an ONTAP event. It is not currently supported by Windows as a single event.  | File Access |
| NA/NA Data ONTAP<br>Event ID 9998 | Unlink Object | OBJECT ACCESS: Object unlinked.<br>This is an ONTAP event. It is not currently supported by Windows as a single event. | File Access |

## Additional information about Event 4656

The `HandleID` tag in the audit XML event contains the handle of the object (file or directory) accessed. The `HandleID` tag for the EVTX 4656 event contains different information depending on whether the open event is for creating a new object or for opening an existing object:

- If the open event is an open request to create a new object (file or directory), the `HandleID` tag in the audit XML event shows an empty `HandleID` (for example: `<Data Name="HandleID">0000000000000000;00;00000000;00000000</Data>` ).

The `HandleID` is empty because the OPEN (for creating a new object) request gets audited before the actual object creation happens and before a handle exists. Subsequent audited events for the same object have the right object handle in the `HandleID` tag.

- If the open event is an open request to open an existing object, the audit event will have the assigned handle of that object in the `HandleID` tag (for example: `<Data Name="HandleID">000000000000401;00;000000ea;00123ed4</Data>` ).

## Determine what the complete path to the audited object is

The object path printed in the `<ObjectName>` tag for an audit record contains the name of the volume (in parentheses) and the relative path from the root of the containing volume. If you want to determine the complete path of the audited object, including the junction path, there are certain steps you must take.

### Steps

1. Determine what the volume name and relative path to audited object is by looking at the `<ObjectName>` tag in the audit event.

In this example, the volume name is “data1” and the relative path to the file is `/dir1/file.txt`:

```
<Data Name="ObjectName"> (data1);/dir1/file.txt </Data>
```

2. Using the volume name determined in the previous step, determine what the junction path is for the volume containing the audited object:

In this example, the volume name is “data1” and the junction path for the volume containing the audited object is `/data/data1`:

```
volume show -junction -volume data1
```

| Vserver | Volume | Junction    |        | Junction Path | Junction Path Source |
|---------|--------|-------------|--------|---------------|----------------------|
|         |        | Language    | Active |               |                      |
| vs1     | data1  | en_US.UTF-8 | true   | /data/data1   | RW_volume            |

- Determine the full path to the audited object by appending the relative path found in the `<ObjectName>` tag to the junction path for the volume.

In this example, the junction path for the volume:

```
/data/data1/dir1/file.text
```

## Considerations when auditing symlinks and hard links

There are certain considerations you must keep in mind when auditing symlinks and hard links.

An audit record contains information about the object being audited including the path to the audited object, which is identified in the `ObjectName` tag. You should be aware of how paths for symlinks and hard links are recorded in the `ObjectName` tag.

### Symlinks

A symlink is a file with a separate inode that contains a pointer to the location of a destination object, known as the target. When accessing an object through a symlink, ONTAP automatically interprets the symlink and follows the actual canonical protocol agnostic path to the target object in the volume.

In the following example output, there are two symlinks, both pointing to a file named `target.txt`. One of the symlinks is a relative symlink and one is an absolute symlink. If either of the symlinks are audited, the `ObjectName` tag in the audit event contains the path to the file `target.txt`:

```
[root@host1 audit]# ls -l
total 0
lrwxrwxrwx 1 user1 group1 37 Apr  2 10:09 softlink_fullpath.txt ->
/data/audit/target.txt
lrwxrwxrwx 1 user1 group1 10 Apr  2 09:54 softlink.txt -> target.txt
-rwxrwxrwx 1 user1 group1 16 Apr  2 10:05 target.txt
```

### Hard links

A hard link is a directory entry that associates a name with an existing file on a file system. The hard link points to the inode location of the original file. Similar to how ONTAP interprets symlinks, ONTAP interprets the hard link and follows the actual canonical path to the target object in the volume. When access to a hard link object is audited, the audit event records this absolute canonical path in the `ObjectName` tag rather than the hard link path.

# Considerations when auditing alternate NTFS data streams

There are certain considerations you must keep in mind when auditing files with NTFS alternate data streams.

The location of an object being audited is recorded in an event record using two tags, the `ObjectName` tag (the path) and the `HandleID` tag (the handle). To properly identify which stream requests are being logged, you must be aware of what ONTAP records in these fields for NTFS alternate data streams:

- EVTX ID: 4656 events (open and create audit events)
  - The path of the alternate data stream is recorded in the `ObjectName` tag.
  - The handle of the alternate data stream is recorded in the `HandleID` tag.
- EVTX ID: 4663 events (all other audit events, such as read, write, setattr, and so on)
  - The path of the base file, not the alternate data stream, is recorded in the `ObjectName` tag.
  - The handle of the alternate data stream is recorded in the `HandleID` tag.

## Example

The following example illustrates how to identify EVTX ID: 4663 events for alternate data streams using the `HandleID` tag. Even though the `ObjectName` tag (path) recorded in the read audit event is to the base file path, the `HandleID` tag can be used to identify the event as an audit record for the alternate data stream.

Stream file names take the form `base_file_name:stream_name`. In this example, the `dir1` directory contains a base file with an alternate data stream having the following paths:

```
/dir1/file1.txt  
/dir1/file1.txt:stream1
```



The output in the following event example is truncated as indicated; the output does not display all of the available output tags for the events.

For an EVTX ID 4656 (open audit event), the audit record output for the alternate data stream records the alternate data stream name in the `ObjectName` tag:

```

- <Event>
- <System>
  <Provider Name="Netapp-Security-Auditing" />
  <EventID>4656</EventID>
  <EventName>Open Object</EventName>
  [...]
</System>
- <EventData>
  [...]
  **<Data Name="ObjectType">Stream</Data>
  <Data Name="HandleID">00000000000401;00;000001e4;00176767</Data>
  <Data Name="ObjectName">\(data1\);/dir1/file1.txt:stream1</Data>
  **
  [...]
</EventData>
</Event>
- <Event>

```

For an EVT\_X ID 4663 (read audit event), the audit record output for the same alternate data stream records the base file name in the `ObjectName` tag; however, the handle in the `HandleID` tag is the alternative data stream's handle and can be used to correlate this event with the alternative data stream:

```

- <Event>
- <System>
  <Provider Name="Netapp-Security-Auditing" />
  <EventID>4663</EventID>
  <EventName>Read Object</EventName>
  [...]
</System>
- <EventData>
  [...]
  **<Data Name="ObjectType">Stream</Data>
  <Data Name="HandleID">00000000000401;00;000001e4;00176767</Data>
  <Data Name="ObjectName">\(data1\);/dir1/file1.txt</Data> **
  [...]
</EventData>
</Event>
- <Event>

```



## Copyright Information

Copyright © 2022 NetApp, Inc. All rights reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means-graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system- without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

## Trademark Information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.