# Review audit logs

## StorageGRID software

NetApp
February 12, 2026

This PDF was generated from https://docs.netapp.com/us-en/storagegrid/audit/index.html on February 12, 2026. Always check docs.netapp.com for the latest.

# Table of Contents

# Review audit logs

## Audit messages and logs

These instructions contain information about the structure and content of StorageGRID audit messages and audit logs. You can use this information to read and analyze the audit trail of system activity.

These instructions are for administrators responsible for producing reports of system activity and usage that require analysis of the StorageGRID system's audit messages.

To use the text log file, you must have access to the configured audit share on the Admin Node.

For information about configuring audit message levels and using an external syslog server, see Configure log management and external syslog server.

## Audit message flow and retention

All StorageGRID services generate audit messages during normal system operation. You should understand how these audit messages move through the StorageGRID system to the `audit.log` file.

The following workflows for audit messages and audit message retention are only applicable if StorageGRID is configured for **Admin Nodes/local nodes** or **Admin Node and external syslog server**. If StorageGRID is configured for "Local nodes only" (default) or "External syslog server", the audit messages are saved locally on each node in the `/var/local/log/localaudit.log` file and can't be processed by Admin Nodes or Storage Nodes.

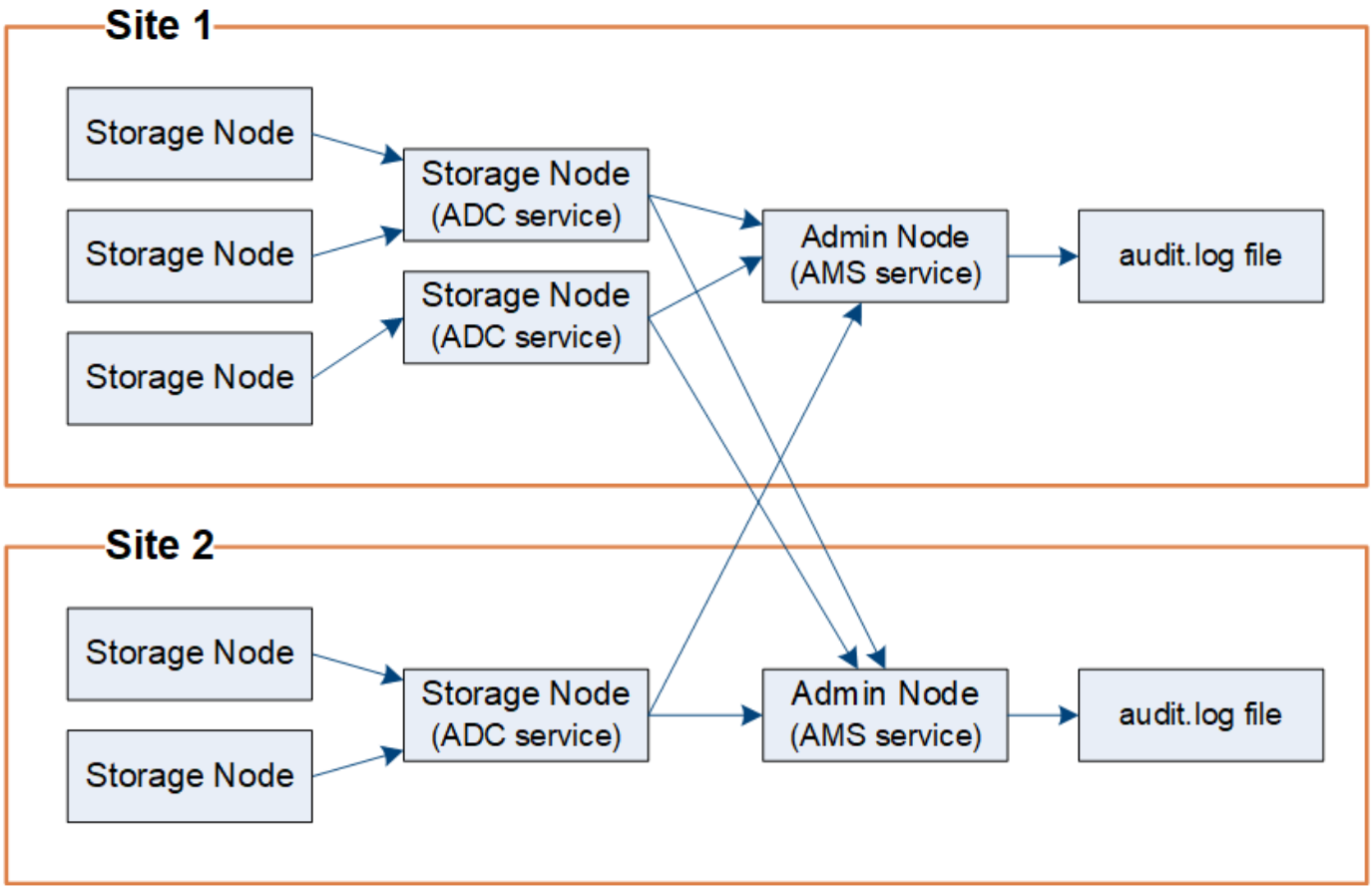
### Audit message flow

Audit messages are processed by Admin Nodes when StorageGRID is configured for **Admin Nodes/local nodes** or **Admin Node and external syslog server** and by those Storage Nodes that have an Administrative Domain Controller (ADC) service.

As shown in the audit message flow diagram, each StorageGRID node sends its audit messages to one of the ADC services at the data center site. The ADC service is automatically enabled for the first three Storage Nodes installed at each site.

In turn, each ADC service acts as a relay and sends its collection of audit messages to every Admin Node in the StorageGRID system, which gives each Admin Node a complete record of system activity.

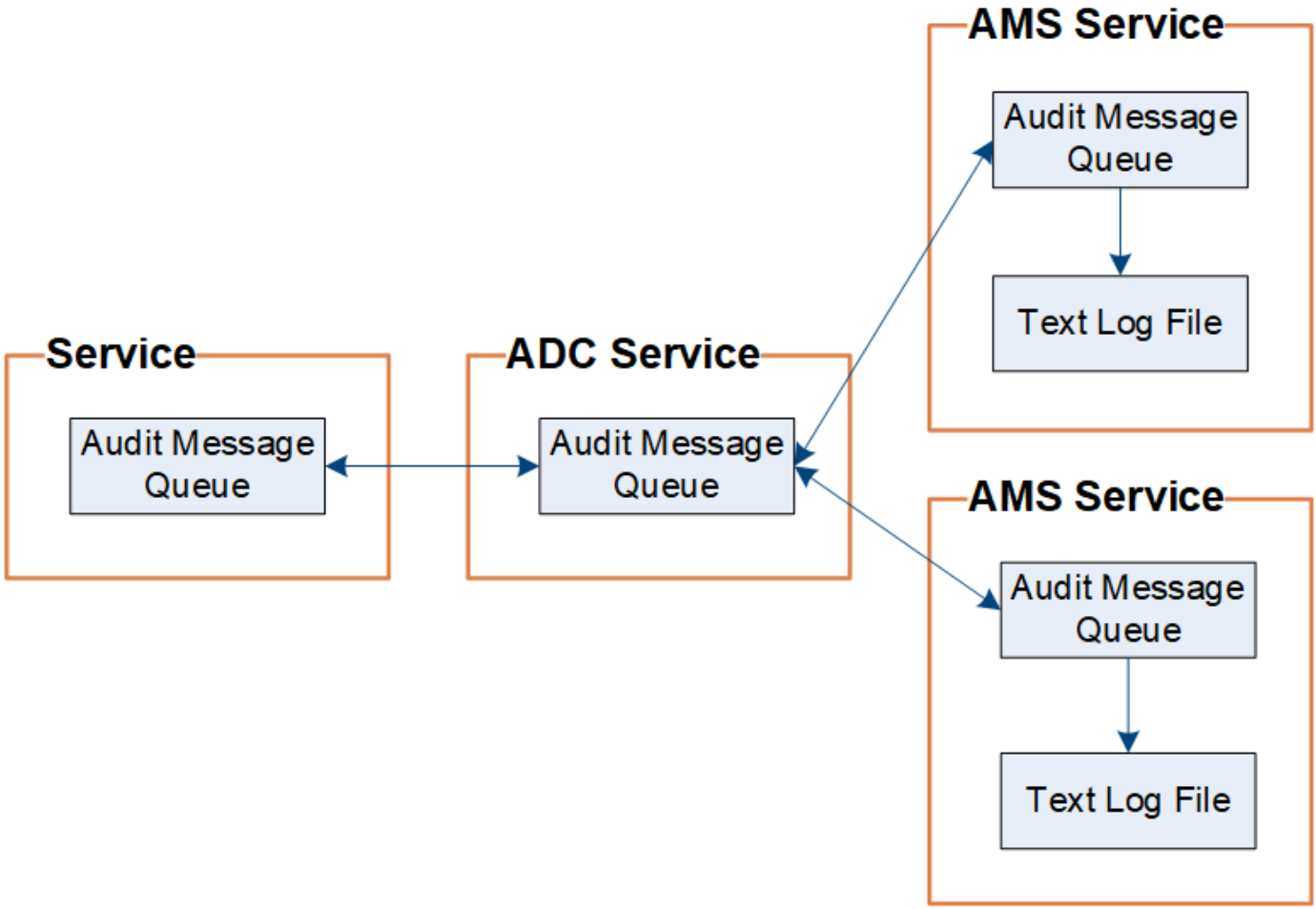
Each Admin Node stores audit messages in text log files; the active log file is named `audit.log`.




**Audit message retention**

StorageGRID uses a copy-and-delete process to ensure that no audit messages are lost before they can be written to the audit log.

When a node generates or relays an audit message, the message is stored in an audit message queue on the system disk of the grid node. A copy of the message is always held in an audit message queue until the message is written to the audit log file in the Admin Node's `/var/local/audit/export` directory. This helps prevent loss of an audit message during transport.




The audit message queue can temporarily increase due to network connectivity issues or insufficient audit capacity. As the queues increase, they consume more of the available space in each node's `/var/local/` directory. If the issue persists and a node's audit message directory becomes too full, the individual nodes prioritize processing their backlog and become temporarily unavailable for new messages.

Specifically, you might see the following behaviors:

- If the `/var/local/audit/export` directory used by an Admin Node becomes full, the Admin Node is flagged as unavailable to new audit messages until the directory is no longer full. S3 client requests aren't affected. The XAMS (Unreachable Audit Repositories) alarm is triggered when an audit repository is unreachable.
- If the `/var/local/` directory used by a Storage Node with the ADC service becomes 92% full, the node is flagged as unavailable to audit messages until the directory is only 87% full. S3 client requests to other nodes aren't affected. The NRLY (Available Audit Relays) alarm is triggered when audit relays are unreachable.

> If there are no available Storage Nodes with the ADC service, the Storage Nodes store the audit messages locally in the `/var/local/log/localaudit.log` file.

- If the `/var/local/` directory used by a Storage Node becomes 85% full, the node starts refusing S3 client requests with `503 Service Unavailable`.

The following types of issues can cause audit message queues to grow very large:

- The outage of an Admin Node or a Storage Node with the ADC service. If one of the system's nodes is down, the remaining nodes might become backlogged.
- A sustained activity rate that exceeds the audit capacity of the system.
- The `/var/local/` space on an ADC Storage Node becoming full for reasons unrelated to audit messages. When this happens, the node stops accepting new audit messages and prioritizes its current backlog, which can cause backlogs on other nodes.

#### Large audit queue alert and Audit Messages Queued (AMQS) alarm

To help you monitor the size of audit message queues over time, the **Large audit queue** alert and the legacy AMQS alarm are triggered when the number of messages in a Storage Node queue or Admin Node queue reaches certain thresholds.

If the **Large audit queue** alert or the legacy AMQS alarm is triggered, start by checking the load on the system—if there have been a significant number of recent transactions, the alert and the alarm should resolve over time and can be ignored.

If the alert or alarm persists and increases in severity, view a chart of the queue size. If the number is steadily increasing over hours or days, the audit load has likely exceeded the audit capacity of the system. Reduce the client operation rate or decrease the number of audit messages logged by changing the audit level for Client Writes and Client Reads to Error or Off. See Configure log management and external syslog server.

#### Duplicate messages

The StorageGRID system takes a conservative approach if a network or node failure occurs. For this reason, duplicate messages might exist in the audit log.

# Access audit log file

The audit share contains the active `audit.log` file and any compressed audit log files. You can access audit log files directly from the command line of the Admin Node.

The `audit.log` file remains empty unless you configure StorageGRID for **Admin Nodes/local nodes** or **Admin Node and external syslog server**. For more information, refer to Select log location.

#### Before you begin

- You have specific access permissions.
- You must have the `Passwords.txt` file.
- You must know the IP address of an Admin Node.

#### Steps

1. Log in to an Admin Node:
   a. Enter the following command: `ssh admin@primary_Admin_Node_IP`
   b. Enter the password listed in the `Passwords.txt` file.
   c. Enter the following command to switch to root: `su -`
   d. Enter the password listed in the `Passwords.txt` file.

      When you are logged in as root, the prompt changes from `$` to `#`.

2. Go to the directory containing the audit log files:

```
cd /var/local/audit/export/
```

3. View the current or a saved audit log file, as required.

# Audit log file rotation

If StorageGRID is configured for **Admin Nodes/local nodes** or **Admin Node and external syslog server**, the audit logs files are saved to the Admin Node's `/var/local/audit/export/` directory. The active audit log files are named `audit.log`.

> Optionally, you can change the destination of audit logs and send audit information to an external syslog server. Local logs of audit records continue to be generated and stored when an external syslog server is configured. Refer to Configure audit messages and external syslog server.

Once a day, the active `audit.log` file is saved, and a new `audit.log` file is started. The name of the saved file indicates when it was saved, in the format *`yyyy-mm-dd.txt`*. If more than one audit log is created in a single day, the file names use the date the file was saved, appended by a number, in the format *`yyyy-mm-dd.txt.n`*. For example, `2018-04-15.txt` and `2018-04-15.txt.1` are the first and second log files created and saved on 15 April 2018.

After a day, the saved file is compressed and renamed, in the format *`yyyy-mm-dd.txt.gz`*, which preserves the original date. Over time, Admin Node storage allocated for audit logs is consumed. A script monitors the audit log space consumption and deletes log files as necessary to free space in the `/var/local/audit/export/` directory. Audit logs are deleted based on the date they were created. The oldest logs are deleted first. You can monitor the script's actions in the following file: `/var/local/log/manage-audit.log`.

This example shows the active `audit.log` file, the previous day's file (`2018-04-15.txt`), and the compressed file for the prior day (`2018-04-14.txt.gz`).

```
audit.log
2018-04-15.txt
2018-04-14.txt.gz
```

# Audit log file format

## Audit log file format

The audit log files are found on every Admin Node and contain a collection of individual audit messages.

Each audit message contains the following:

- The Coordinated Universal Time (UTC) of the event that triggered the audit message (ATIM) in ISO 8601

format, followed by a space:

*YYYY-MM-DDTHH:MM:SS.UUUUUU*, where *UUUUUU* are microseconds.

- The audit message itself, enclosed within square brackets and beginning with `AUDT`.

The following example shows three audit messages in an audit log file (line breaks added for readability). These messages were generated when a tenant created an S3 bucket and added two objects to that bucket.

```
2019-08-07T18:43:30.247711
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991681][TIME(UI64):73520][SAI
P(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::17530064241597054
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][AVER(UI32):10][ATIM(UI64):1565203410247711]
[ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(FC32):S3RQ][ATID(UI64):7074142
142472611085]]

2019-08-07T18:43:30.783597
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991696][TIME(UI64):120713][SA
IP(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::17530064241597054
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][S3KY(CSTR):"fh-small-0"]
[CBID(UI64):0x779557A069B2C037][UUID(CSTR):"94BA6949-38E1-4B0C-BC80-
EB44FB4FCC7F"][CSIZ(UI64):1024][AVER(UI32):10]
[ATIM(UI64):1565203410783597][ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(F
C32):S3RQ][ATID(UI64):8439606722108456022]]

2019-08-07T18:43:30.784558
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991693][TIME(UI64):121666][SA
IP(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::17530064241597054
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][S3KY(CSTR):"fh-small-2000"]
[CBID(UI64):0x180CBD8E678EED17][UUID(CSTR):"19CE06D0-D2CF-4B03-9C38-
E578D66F7ADD"][CSIZ(UI64):1024][AVER(UI32):10]
[ATIM(UI64):1565203410784558][ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(F
C32):S3RQ][ATID(UI64):13489590586043706682]]
```

In their default format, the audit messages in the audit log files aren’t easy to read or interpret. You can use the

audit-explain tool to obtain simplified summaries of the audit messages in the audit log. You can use the audit-sum tool to summarize how many write, read, and delete operations were logged and how long these operations took.

## Use audit-explain tool

You can use the `audit-explain` tool to translate the audit messages in the audit log in to an easy-to-read format.

**Before you begin**

- You have specific access permissions.
- You must have the `Passwords.txt` file.
- You must know the IP address of the primary Admin Node.

**About this task**

The `audit-explain` tool, available on the primary Admin Node, provides simplified summaries of the audit messages in an audit log.

> The `audit-explain` tool is primarily intended for use by technical support during troubleshooting operations. Processing `audit-explain` queries can consume a large amount of CPU power, which might impact StorageGRID operations.

This example shows typical output from the `audit-explain` tool. These four SPUT audit messages were generated when the S3 tenant with account ID 92484777680322627870 used S3 PUT requests to create a bucket named "bucket1" and add three objects to that bucket.

```
SPUT S3 PUT bucket bucket1 account:92484777680322627870 usec:124673
SPUT S3 PUT object bucket1/part1.txt tenant:92484777680322627870
cbid:9DCB157394F99FE5 usec:101485
SPUT S3 PUT object bucket1/part2.txt tenant:92484777680322627870
cbid:3CFBB07AB3D32CA9 usec:102804
SPUT S3 PUT object bucket1/part3.txt tenant:92484777680322627870
cbid:5373D73831ECC743 usec:93874
```

The `audit-explain` tool can do the following:

- Process plain or compressed audit logs. For example:

  ```
  audit-explain audit.log

  audit-explain 2019-08-12.txt.gz
  ```

- Process multiple files simultaneously. For example:

  ```
  audit-explain audit.log 2019-08-12.txt.gz 2019-08-13.txt.gz

  audit-explain /var/local/audit/export/*
  ```

- Accept input from a pipe, which allows you to filter and preprocess the input using the `grep` command or other means. For example:

  ```
  grep SPUT audit.log | audit-explain

  grep bucket-name audit.log | audit-explain
  ```

Because audit logs can be very large and slow to parse, you can save time by filtering parts that you want to look at and running `audit-explain` on the parts, instead of the entire file.

> The `audit-explain` tool does not accept compressed files as piped input. To process compressed files, provide their file names as command-line arguments, or use the `zcat` tool to decompress the files first. For example:
>
> ```
> zcat audit.log.gz | audit-explain
> ```

Use the `help (-h)` option to see the available options. For example:

```
$ audit-explain -h
```

**Steps**

1. Log in to the primary Admin Node:
   a. Enter the following command: `ssh admin@`*`primary_Admin_Node_IP`*
   b. Enter the password listed in the `Passwords.txt` file.
   c. Enter the following command to switch to root: `su -`
   d. Enter the password listed in the `Passwords.txt` file.

      When you are logged in as root, the prompt changes from `$` to `#`.

2. Enter the following command, where `/var/local/audit/export/audit.log` represents the name and the location of the file or files you want to analyze:

   ```
   $ audit-explain /var/local/audit/export/audit.log
   ```

   The `audit-explain` tool prints human-readable interpretations of all messages in the specified file or files.

   > To reduce line lengths and to aid readability, timestamps aren't shown by default. If you want to see the timestamps, use the timestamp (`-t`) option.

## Use audit-sum tool

You can use the `audit-sum` tool to count the write, read, head, and delete audit messages and to see the minimum, maximum, and average time (or size) for each operation type.

**Before you begin**

- You have specific access permissions.
- You have the `Passwords.txt` file.
- You know the IP address of the primary Admin Node.

**About this task**

The `audit-sum` tool, available on the primary Admin Node, summarizes how many write, read, and delete operations were logged and how long these operations took.

> The `audit-sum` tool is primarily intended for use by technical support during troubleshooting operations. Processing `audit-sum` queries can consume a large amount of CPU power, which might impact StorageGRID operations.

This example shows typical output from the `audit-sum` tool. This example shows how long protocol operations took.

```
  message group           count     min(sec)        max(sec)
average(sec)
  =============           =====     ========        ========
============
  IDEL                      274
  SDEL                   213371        0.004          20.934
0.352
  SGET                   201906        0.010        1740.290
1.132
  SHEA                    22716        0.005           2.349
0.272
  SPUT                 1771398        0.011        1770.563
0.487
```

The `audit-sum` tool provides counts and times for the following S3 and ILM audit messages in an audit log.

> Audit codes are removed from the product and documentation as features are deprecated. If you encounter an audit code that isn't listed here, check the previous versions of this topic for older StorageGRID releases. For example, StorageGRID 11.8 Using audit sum tool.

| Code | Description | Refer to |
|---|---|---|
| IDEL | ILM Initiated Delete: Logs when ILM starts the process of deleting an object. | IDEL: ILM Initiated Delete |
| SDEL | S3 DELETE: Logs a successful transaction to delete an object or bucket. | SDEL: S3 DELETE |
| SGET | S3 GET: Logs a successful transaction to retrieve an object or list the objects in a bucket. | SGET: S3 GET |
| SHEA | S3 HEAD: Logs a successful transaction to check for the existence of an object or bucket. | SHEA: S3 HEAD |

| Code | Description | Refer to |
|---|---|---|
| SPUT | S3 PUT: Logs a successful transaction to create a new object or bucket. | SPUT: S3 PUT |

The `audit-sum` tool can do the following:

- Process plain or compressed audit logs. For example:

```
audit-sum audit.log

audit-sum 2019-08-12.txt.gz
```

- Process multiple files simultaneously. For example:

```
audit-sum audit.log 2019-08-12.txt.gz 2019-08-13.txt.gz

audit-sum /var/local/audit/export/*
```

- Accept input from a pipe, which allows you to filter and preprocess the input using the `grep` command or other means. For example:

```
grep WGET audit.log | audit-sum

grep bucket1 audit.log | audit-sum

grep SPUT audit.log | grep bucket1 | audit-sum
```

> This tool doesn't accept compressed files as piped input. To process compressed files, provide their file names as command-line arguments, or use the `zcat` tool to decompress the files first. For example:
>
> ```
> audit-sum audit.log.gz
>
> zcat audit.log.gz | audit-sum
> ```

You can use command-line options to summarize operations on buckets separately from operations on objects or to group message summaries by bucket name, by time period, or by target type. By default, the summaries show the minimum, maximum, and average operation time, but you can use the `size (-s)` option to look at object size instead.

Use the `help (-h)` option to see the available options. For example:

```
$ audit-sum -h
```

**Steps**

1. Log in to the primary Admin Node:
   a. Enter the following command: `ssh admin@`*`primary_Admin_Node_IP`*
   b. Enter the password listed in the `Passwords.txt` file.
   c. Enter the following command to switch to root: `su -`

d. Enter the password listed in the `Passwords.txt` file.

   When you are logged in as root, the prompt changes from `$` to `#`.

2. If you want to analyze all messages related to write, read, head, and delete operations, follow these steps:

   a. Enter the following command, where `/var/local/audit/export/audit.log` represents the name and the location of the file or files you want to analyze:

      ```
      $ audit-sum /var/local/audit/export/audit.log
      ```

      This example shows typical output from the `audit-sum` tool. This example shows how long protocol operations took.

      ```
        message group           count     min(sec)        max(sec)
      average(sec)
        =============           =====     ========        ========
      ============
        IDEL                      274
        SDEL                   213371        0.004          20.934
      0.352
        SGET                   201906        0.010        1740.290
      1.132
        SHEA                    22716        0.005           2.349
      0.272
        SPUT                 1771398        0.011        1770.563
      0.487
      ```

      In this example, SGET (S3 GET) operations are the slowest on average at 1.13 seconds, but SGET and SPUT (S3 PUT) operations both show long worst-case times of about 1,770 seconds.

   b. To show the slowest 10 retrieval operations, use the grep command to select only SGET messages and add the long output option (`-l`) to include object paths:

      ```
      grep SGET audit.log | audit-sum -l
      ```

      The results include the type (object or bucket) and path, which allows you to grep the audit log for other messages relating to these particular objects.

```
Total:          201906 operations
    Slowest:      1740.290 sec
    Average:         1.132 sec
    Fastest:         0.010 sec
    Slowest operations:
        time(usec)       source ip         type      size(B) path
        ========== ============== ============ ============ ====
        1740289662   10.96.101.125       object   5663711385
backup/r9O1OaQ8JB-1566861764-4519.iso
        1624414429   10.96.101.125       object   5375001556
backup/r9O1OaQ8JB-1566861764-6618.iso
        1533143793   10.96.101.125       object   5183661466
backup/r9O1OaQ8JB-1566861764-4518.iso
             70839   10.96.101.125       object        28338
bucket3/dat.1566861764-6619
             68487   10.96.101.125       object        27890
bucket3/dat.1566861764-6615
             67798   10.96.101.125       object        27671
bucket5/dat.1566861764-6617
             67027   10.96.101.125       object        27230
bucket5/dat.1566861764-4517
             60922   10.96.101.125       object        26118
bucket3/dat.1566861764-4520
             35588   10.96.101.125       object        11311
bucket3/dat.1566861764-6616
             23897   10.96.101.125       object        10692
bucket3/dat.1566861764-4516
```

From this example output, you can see that the three slowest S3 GET requests were for objects about 5 GB in size, which is much larger than the other objects. The large size accounts for the slow worst-case retrieval times.

3. If you want to determine what sizes of objects are being ingested into and retrieved from your grid, use the size option (`-s`):

   ```
   audit-sum -s audit.log
   ```

```
  message group           count       min(MB)          max(MB)
average(MB)
  =============           =====     ========        ========
============
  IDEL                      274        0.004        5000.000
1654.502
  SDEL                   213371        0.000          10.504
1.695
  SGET                   201906        0.000        5000.000
14.920
  SHEA                    22716        0.001          10.504
2.967
  SPUT                 1771398        0.000        5000.000
2.495
```

In this example, the average object size for SPUT is under 2.5 MB, but the average size for SGET is much larger. The number of SPUT messages is much higher than the number of SGET messages, indicating that most objects are never retrieved.

4. If you want to determine if retrievals were slow yesterday:
   a. Issue the command on the appropriate audit log and use the group-by-time option (`-gt`), followed by the time period (for example, 15M, 1H, 10S):

      ```
      grep SGET audit.log | audit-sum -gt 1H
      ```

```
  message group           count    min(sec)       max(sec)
average(sec)
  =============           =====     ========        ========
============
  2019-09-05T00            7591        0.010        1481.867
1.254
  2019-09-05T01            4173        0.011        1740.290
1.115
  2019-09-05T02           20142        0.011        1274.961
1.562
  2019-09-05T03           57591        0.010        1383.867
1.254
  2019-09-05T04          124171        0.013        1740.290
1.405
  2019-09-05T05          420182        0.021        1274.511
1.562
  2019-09-05T06         1220371        0.015        6274.961
5.562
  2019-09-05T07          527142        0.011        1974.228
2.002
  2019-09-05T08          384173        0.012        1740.290
1.105
  2019-09-05T09           27591        0.010        1481.867
1.354
```

These results show that S3 GET traffic spiked between 06:00 and 07:00. The max and average times are both considerably higher during this time span, and they didn't ramp up gradually as the count increased. These metrics suggest that capacity was exceeded, possibly in the network or in the grid's ability to process requests.

b. To determine what size objects were being retrieved each hour yesterday, add the size option (`-s`) to the command:

```
grep SGET audit.log | audit-sum -gt 1H -s
```

```
  message group           count       min(B)          max(B)
average(B)
  =============           =====     ========        ========
============
  2019-09-05T00            7591        0.040        1481.867
1.976
  2019-09-05T01            4173        0.043        1740.290
2.062
  2019-09-05T02           20142        0.083        1274.961
2.303
  2019-09-05T03           57591        0.912        1383.867
1.182
  2019-09-05T04          124171        0.730        1740.290
1.528
  2019-09-05T05          420182        0.875        4274.511
2.398
  2019-09-05T06         1220371        0.691  5663711385.961
51.328
  2019-09-05T07          527142        0.130        1974.228
2.147
  2019-09-05T08          384173        0.625        1740.290
1.878
  2019-09-05T09           27591        0.689        1481.867
1.354
```

These results indicate that some very large retrievals occurred when the overall retrieval traffic was at its maximum.

c. To see more detail, use the audit-explain tool to review all the SGET operations during that hour:

```
grep 2019-09-05T06 audit.log | grep SGET | audit-explain | less
```

If the output of the grep command is expected to be many lines, add the `less` command to show the contents of the audit log file one page (one screen) at a time.

5. If you want to determine if SPUT operations on buckets are slower than SPUT operations for objects:

a. Start by using the `-go` option, which groups messages for object and bucket operations separately:

```
grep SPUT sample.log | audit-sum -go
```

```
  message group           count     min(sec)        max(sec)
average(sec)
  =============           =====     ========        ========
============
  SPUT.bucket                 1        0.125           0.125
0.125
  SPUT.object                12        0.025           1.019
0.236
```

The results show that SPUT operations for buckets have different performance characteristics than SPUT operations for objects.

b. To determine which buckets have the slowest SPUT operations, use the `-gb` option, which groups messages by bucket:

```
grep SPUT audit.log | audit-sum -gb
```

```
  message group                  count     min(sec)        max(sec)
average(sec)
  =============                  =====     ========        ========
============
  SPUT.cho-non-versioning        71943        0.046        1770.563
1.571
  SPUT.cho-versioning            54277        0.047        1736.633
1.415
  SPUT.cho-west-region           80615        0.040          55.557
1.329
  SPUT.ldt002                  1564563        0.011          51.569
0.361
```

c. To determine which buckets have the largest SPUT object size, use both the `-gb` and the `-s` options:

```
grep SPUT audit.log | audit-sum -gb -s
```

```
  message group                  count       min(B)          max(B)
average(B)
  =============                  =====     ========        ========
============
  SPUT.cho-non-versioning        71943        2.097        5000.000
21.672
  SPUT.cho-versioning            54277        2.097        5000.000
21.120
  SPUT.cho-west-region           80615        2.097         800.000
14.433
  SPUT.ldt002                  1564563        0.000         999.972
0.352
```

# Audit message format

## Audit message format

Audit messages exchanged within the StorageGRID system include standard information common to all messages and specific content describing the event or activity being reported.

If the summary information provided by the audit-explain and audit-sum tools is insufficient, refer to this section to understand the general format of all audit messages.

The following is an example audit message as it might appear in the audit log file:

```
2014-07-17T03:50:47.484627
[AUDT:[RSLT(FC32):VRGN][AVER(UI32):10][ATIM(UI64):1405569047484627][ATYP(F
C32):SYSU][ANID(UI32):11627225][AMID(FC32):ARNI][ATID(UI64):94457363265006
03516]]
```

Each audit message contains a string of attribute elements. The entire string is enclosed in brackets (`[ ]`), and each attribute element in the string has the following characteristics:

- Enclosed in brackets `[ ]`
- Introduced by the string `AUDT`, which indicates an audit message
- Without delimiters (no commas or spaces) before or after
- Terminated by a line feed character `\n`

Each element includes an attribute code, a data type, and a value that are reported in this format:

```
[ATTR(type):value][ATTR(type):value]...
[ATTR(type):value]\n
```

The number of attribute elements in the message depends on the event type of the message. The attribute elements aren't listed in any particular order.

The following list describes the attribute elements:

- `ATTR` is a four-character code for the attribute being reported. There are some attributes that are common to all audit messages and others that are event-specific.
- `type` is a four-character identifier of the programming data type of the value, such as UI64, FC32, and so on. The type is enclosed in parentheses `( )`.
- `value` is the content of the attribute, typically a numeric or text value. Values always follow a colon (`:`). Values of data type CSTR are surrounded by double quotes " ".

## Data types

Different data types are used to store information in audit messages.

| Type | Description |
|---|---|
| UI32 | Unsigned long integer (32 bits); it can store the numbers 0 to 4,294,967,295. |
| UI64 | Unsigned double long integer (64 bits); it can store the numbers 0 to 18,446,744,073,709,551,615. |
| FC32 | Four-character constant; a 32-bit unsigned integer value represented as four ASCII characters such as "ABCD." |
| IPAD | Used for IP addresses. |
| CSTR | A variable-length array of UTF-8 characters. Characters can be escaped with the following conventions:<br>• Backslash is \\.<br>• Carriage return is \r.<br>• Double quotes is \".<br>• Line feed (new line) is \n.<br>• Characters can be replaced by their hexadecimal equivalents (in the format \xHH, where HH is the hexadecimal value representing the character). |

## Event-specific data

Each audit message in the audit log records data specific to a system event.

Following the opening `[AUDT:` container that identifies the message itself, the next set of attributes provide

information about the event or action described by the audit message. These attributes are highlighted in the following example:

2018-12-05T08:24:45.921845 [AUDT:*\[RSLT\(FC32\):SUCS\]* **\[TIME\(UI64\):11454\]\[SAIP\(IPAD\):"10.224.0.100"\]\[S3AI\(CSTR\):"60025621595611246499"\] \[SACC\(CSTR\):"account"\]\[S3AK\(CSTR\):"SGKH4_Nc8SO1H6w3w0nCOFCGgk__E6dYzKlumRs KJA=="\] \[SUSR\(CSTR\):"urn:sgws:identity::60025621595611246499:root"\] \[SBAI\(CSTR\):"60025621595611246499"\]\[SBAC\(CSTR\):"account"\]\[S3BK\(CSTR\):"bucket"\] \[S3KY\(CSTR\):"object"\]\[CBID\(UI64\):0xCC128B9B9E428347\] \[UUID\(CSTR\):"B975D2CE-E4DA-4D14-8A23-1CB4B83F2CD8"\]\[CSIZ\(UI64\):30720\]**[AVER(UI32):10] \[ATIM(UI64):1543998285921845]\[ATYP\(FC32\):SHEA\][ANID(UI32):12281045][AMID(FC32):S3RQ] \[ATID(UI64):15552417629170647261]]

The `ATYP` element (underlined in the example) identifies which event generated the message. This example message includes the SHEA message code ([ATYP(FC32):SHEA]), indicating it was generated by a successful S3 HEAD request.

## Common elements in audit messages

All audit messages contain the common elements.

| Code | Type | Description |
|---|---|---|
| AMID | FC32 | Module ID: A four-character identifier of the module ID that generated the message. This indicates the code segment within which the audit message was generated. |
| ANID | UI32 | Node ID: The grid node ID assigned to the service that generated the message. Each service is allocated a unique identifier at the time the StorageGRID system is configured and installed. This ID can't be changed. |
| ASES | UI64 | Audit Session Identifier: In previous releases, this element indicated the time at which the audit system was initialized after the service started up. This time value was measured in microseconds since the operating system epoch (00:00:00 UTC on 1 January, 1970).<br><br>**Note:** This element is obsolete and no longer appears in audit messages. |
| ASQN | UI64 | Sequence Count: In previous releases, this counter was incremented for each generated audit message on the grid node (ANID) and reset to zero at service restart.<br><br>**Note:** This element is obsolete and no longer appears in audit messages. |
| ATID | UI64 | Trace ID: An identifier that is shared by the set of messages that were triggered by a single event. |

| Code | Type | Description |
|---|---|---|
| ATIM | UI64 | Timestamp: The time the event was generated that triggered the audit message, measured in microseconds since the operating system epoch (00:00:00 UTC on 1 January, 1970). Note that most available tools for converting the timestamp to local date and time are based on milliseconds.<br><br>Rounding or truncation of the logged timestamp might be required. The human-readable time that appears at the beginning of the audit message in the `audit.log` file is the ATIM attribute in ISO 8601 format. The date and time are represented as *`YYYY-MMDDTHH:MM:SS.UUUUUU`*, where the `T` is a literal string character indicating the beginning of the time segment of the date. *`UUUUUU`* are microseconds. |
| ATYP | FC32 | Event Type: A four-character identifier of the event being logged. This governs the "payload" content of the message: the attributes that are included. |
| AVER | UI32 | Version: The version of the audit message. As the StorageGRID software evolves, new versions of services might incorporate new features in audit reporting. This field enables backward compatibility in the AMS service to process messages from older versions of services. |
| RSLT | FC32 | Result: The result of event, process, or transaction. If is not relevant for a message, NONE is used rather than SUCS so that the message is not accidentally filtered. |

## Audit message examples

You can find detailed information in each audit message. All audit messages use the same format.

The following is an example audit message as it might appear in the `audit.log` file:

```
2014-07-17T21:17:58.959669
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):246979][S3AI(CSTR):"bc644d
381a87d6cc216adcd963fb6f95dd25a38aa2cb8c9a358e8c5087a6af5f"][
S3AK(CSTR):"UJXDKKQOXB7YARDS71Q2"][S3BK(CSTR):"s3small1"][S3K
Y(CSTR):"hello1"][CBID(UI64):0x50C4F7AC2BC8EDF7][CSIZ(UI64):0
][AVER(UI32):10][ATIM(UI64):1405631878959669][ATYP(FC32):SPUT
][ANID(UI32):12872812][AMID(FC32):S3RQ][ATID(UI64):1579224144
102530435]]
```

The audit message contains information about the event being recorded, as well as information about the audit message itself.

To identify which event is recorded by the audit message, look for the ATYP attribute (highlighted below):

```
2014-07-17T21:17:58.959669
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):246979][S3AI(CSTR):"bc644d
381a87d6cc216adcd963fb6f95dd25a38aa2cb8c9a358e8c5087a6af5f"][
S3AK(CSTR):"UJXDKKQOXB7YARDS71Q2"][S3BK(CSTR):"s3small1"][S3K
Y(CSTR):"hello1"][CBID(UI64):0x50C4F7AC2BC8EDF7][CSIZ(UI64):0
][AVER(UI32):10][ATIM(UI64):1405631878959669][ATYP(FC32):SP
UT][ANID(UI32):12872812][AMID(FC32):S3RQ][ATID(UI64):1579224
144102530435]]
```

The value of the ATYP attribute is SPUT. SPUT represents an S3 PUT transaction, which logs the ingest of an object to a bucket.

The following audit message also shows the bucket to which the object is associated:

```
2014-07-17T21:17:58.959669
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):246979][S3AI(CSTR):"bc644d
381a87d6cc216adcd963fb6f95dd25a38aa2cb8c9a358e8c5087a6af5f"][
S3AK(CSTR):"UJXDKKQOXB7YARDS71Q2"][S3BK\(CSTR\):"s3small1"][S3
KY(CSTR):"hello1"][CBID(UI64):0x50C4F7AC2BC8EDF7][CSIZ(UI64):
0][AVER(UI32):10][ATIM(UI64):1405631878959669][ATYP(FC32):SPU
T][ANID(UI32):12872812][AMID(FC32):S3RQ][ATID(UI64):157922414
4102530435]]
```

To discover when the PUT event occurred, note the Universal Coordinated Time (UTC) timestamp at the beginning of the audit message. This value is a human-readable version of the ATIM attribute of the audit message itself:

```
2014-07-17T21:17:58.959669
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):246979][S3AI(CSTR):"bc644d
381a87d6cc216adcd963fb6f95dd25a38aa2cb8c9a358e8c5087a6af5f"][
S3AK(CSTR):"UJXDKKQOXB7YARDS71Q2"][S3BK(CSTR):"s3small1"][S3K
Y(CSTR):"hello1"][CBID(UI64):0x50C4F7AC2BC8EDF7][CSIZ(UI64):0
][AVER(UI32):10][ATIM\(UI64\):1405631878959669][ATYP(FC32):SP
UT][ANID(UI32):12872812][AMID(FC32):S3RQ][ATID(UI64):15792241
44102530435]]
```

ATIM records the time, in microseconds, since the beginning of the UNIX epoch. In the example, the value `1405631878959669` translates to Thursday, 17-Jul-2014 21:17:59 UTC.

# Audit messages and the object lifecycle

## When are audit message generated?

Audit messages are generated each time an object is ingested, retrieved, or deleted. You can identify these transactions in the audit log by locating S3 API-specific audit messages.

Audit messages are linked through identifiers specific to each protocol.

| Protocol | Code |
|---|---|
| Linking S3 operations | S3BK (bucket), S3KY (key), or both |
| Linking internal operations | CBID (object's internal identifier) |

### Timing of audit messages

Because of factors such as timing differences between grid nodes, object size, and network delays, the order of audit messages generated by the different services can vary from that shown in the examples in this section.

## Object ingest transactions

You can identify client ingest transactions in the audit log by locating S3 API-specific audit messages.

Not all audit messages generated during an ingest transaction are listed in the following table. Only the messages required to trace the ingest transaction are included.

### S3 ingest audit messages

| Code | Name | Description | Trace | See |
|---|---|---|---|---|
| SPUT | S3 PUT transaction | An S3 PUT ingest transaction has completed successfully. | CBID, S3BK, S3KY | SPUT: S3 PUT |
| ORLM | Object Rules Met | The ILM policy has been satisfied for this object. | CBID | ORLM: Object Rules Met |

### Example: S3 object ingest

The series of audit messages below is an example of the audit messages generated and saved to the audit log when an S3 client ingests an object to a Storage Node (LDR service).

In this example, the active ILM policy includes the Make 2 Copies ILM rule.

> Not all audit messages generated during a transaction are listed in the example below. Only those related to the S3 ingest transaction (SPUT) are listed.

This example assumes that an S3 bucket has been previously created.

#### SPUT: S3 PUT

The SPUT message is generated to indicate that an S3 PUT transaction has been issued to create an object in a specific bucket.

```
2017-07-
17T21:17:58.959669[AUDT:[RSLT(FC32):SUCS][TIME(UI64):25771][SAIP(IPAD):"10
.96.112.29"][S3AI(CSTR):"70899244468554783528"][SACC(CSTR):"test"][S3AK(CS
TR):"SGKHyalRU_5cLflqajtaFmxJn946lAWRJfBF33gAOg=="][SUSR(CSTR):"urn:sgws:i
dentity::70899244468554783528:root"][SBAI(CSTR):"70899244468554783528"][SB
AC(CSTR):"test"][S3BK(CSTR):"example"][S3KY(CSTR):"testobject-0-
3"][CBID\(UI64\):0x8EF52DF8025E63A8][CSIZ(UI64):30720][AVER(UI32):10][ATIM
(UI64):150032627859669][ATYP\(FC32\):SPUT][ANID(UI32):12086324][AMID(FC32)
:S3RQ][ATID(UI64):14399932238768197038]]
```

#### ORLM: Object Rules Met

The ORLM message indicates that the ILM policy has been satisfied for this object. The message includes the object's CBID and the name of the ILM rule that was applied.

For replicated objects, the LOCS field includes the LDR node ID and volume ID of the object locations.

```
2019-07-
17T21:18:31.230669[AUDT:[CBID\(UI64\):0x50C4F7AC2BC8EDF7][RULE(CSTR):"Make
2 Copies"][STAT(FC32):DONE][CSIZ(UI64):0][UUID(CSTR):"0B344E18-98ED-4F22-
A6C8-A93ED68F8D3F"][LOCS(CSTR):"CLDI 12828634 2148730112, CLDI 12745543
2147552014"][RSLT(FC32):SUCS][AVER(UI32):10][ATYP\(FC32\):ORLM][ATIM(UI64)
:1563398230669][ATID(UI64):15494889725796157557][ANID(UI32):13100453][AMID
(FC32):BCMS]]
```

For erasure-coded objects, the LOCS field includes the erasure-coding profile ID and the erasure coding group ID

```
2019-02-23T01:52:54.647537
[AUDT:[CBID(UI64):0xFA8ABE5B5001F7E2][RULE(CSTR):"EC_2_plus_1"][STAT(FC32)
:DONE][CSIZ(UI64):10000][UUID(CSTR):"E291E456-D11A-4701-8F51-
D2F7CC9AFECA"][LOCS(CSTR):"CLEC 1 A471E45D-A400-47C7-86AC-
12E77F229831"][RSLT(FC32):SUCS][AVER(UI32):10][ATIM(UI64):1550929974537]\[
ATYP\(FC32\):ORLM\][ANID(UI32):12355278][AMID(FC32):ILMX][ATID(UI64):41685
59046473725560]]
```

The PATH field includes S3 bucket and key information.

```
2019-09-15.txt:2018-01-24T13:52:54.131559
[AUDT:[CBID(UI64):0x82704DFA4C9674F4][RULE(CSTR):"Make 2
Copies"][STAT(FC32):DONE][CSIZ(UI64):3145729][UUID(CSTR):"8C1C9CAC-22BB-
4880-9115-
CE604F8CE687"][PATH(CSTR):"frisbee_Bucket1/GridDataTests151683676324774_1_
1vf9d"][LOCS(CSTR):"CLDI 12525468, CLDI
12222978"][RSLT(FC32):SUCS][AVER(UI32):10][ATIM(UI64):1568555574559][ATYP(
FC32):ORLM][ANID(UI32):12525468][AMID(FC32):OBDI][ATID(UI64):3448338865383
69336]]
```

## Object delete transactions

You can identify object delete transactions in the audit log by locating S3 API-specific audit messages.

Not all audit messages generated during a delete transaction are listed in the following tables. Only messages required to trace the delete transaction are included.

#### S3 delete audit messages

| Code | Name | Description | Trace | See |
|---|---|---|---|---|
| SDEL | S3 Delete | Request made to delete the object from a bucket. | CBID, S3KY | SDEL: S3 DELETE |

#### Example: S3 object deletion

When an S3 client deletes an object from a Storage Node (LDR service), an audit message is generated and saved to the audit log.

> Not all audit messages generated during a delete transaction are listed in the example below. Only those related to the S3 delete transaction (SDEL) are listed.

##### SDEL: S3 Delete

Object deletion begins when the client sends a DeleteObject request to an LDR service. The message contains the bucket from which to delete the object and the object's S3 Key, which is used to identify the object.

```
2017-07-
17T21:17:58.959669[AUDT:[RSLT(FC32):SUCS][TIME(UI64):14316][SAIP(IPAD):"10
.96.112.29"][S3AI(CSTR):"70899244468554783528"][SACC(CSTR):"test"][S3AK(CS
TR):"SGKHyalRU_5cLflqajtaFmxJn946lAWRJfBF33gAOg=="][SUSR(CSTR):"urn:sgws:i
dentity::70899244468554783528:root"][SBAI(CSTR):"70899244468554783528"][SB
AC(CSTR):"test"]\[S3BK\(CSTR\):"example"\]\[S3KY\(CSTR\):"testobject-0-
7"\][CBID\(UI64\):0x339F21C5A6964D89][CSIZ(UI64):30720][AVER(UI32):10][ATI
M(UI64):150032627859669][ATYP\(FC32\):SDEL][ANID(UI32):12086324][AMID(FC32
):S3RQ][ATID(UI64):4727861330952970593]]
```

## Object retrieve transactions

You can identify object retrieve transactions in the audit log by locating S3 API-specific audit messages.

Not all audit messages generated during a retrieve transaction are listed in the following table. Only messages required to trace the retrieve transaction are included.

#### S3 retrieval audit messages

| Code | Name | Description | Trace | See |
|---|---|---|---|---|
| SGET | S3 GET | Request made to retrieve an object from a bucket. | CBID, S3BK, S3KY | SGET: S3 GET |

#### Example: S3 object retrieval

When an S3 client retrieves an object from a Storage Node (LDR service), an audit message is generated and saved to the audit log.

Note that not all audit messages generated during a transaction are listed in the example below. Only those related to the S3 retrieval transaction (SGET) are listed.

##### SGET: S3 GET

Object retrieval begins when the client sends a GetObject request to an LDR service. The message contains the bucket from which to retrieve the object and the object's S3 Key, which is used to identify the object.

```
2017-09-20T22:53:08.782605
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):47807][SAIP(IPAD):"10.96.112.26"][S3AI(
CSTR):"43979298178977966408"][SACC(CSTR):"s3-account-
a"][S3AK(CSTR):"SGKHt7GzEcu0yXhFhT_rL5mep4nJt1w75GBh-
O_FEw=="][SUSR(CSTR):"urn:sgws:identity::43979298178977966408:root"][SBAI(
CSTR):"43979298178977966408"][SBAC(CSTR):"s3-account-
a"]\[S3BK\(CSTR\):"bucket-
anonymous"\]\[S3KY\(CSTR\):"Hello.txt"\][CBID(UI64):0x83D70C6F1F662B02][CS
IZ(UI64):12][AVER(UI32):10][ATIM(UI64):1505947988782605]\[ATYP\(FC32\):SGE
T\][ANID(UI32):12272050][AMID(FC32):S3RQ][ATID(UI64):17742374343649889669]
]
```

If the bucket policy allows, a client can anonymously retrieve objects, or can retrieve objects from a bucket that is owned by a different tenant account. The audit message contains information about the bucket owner's tenant account so that you can track these anonymous and cross-account requests.

In the following example message, the client sends a GetObject request for an object stored in a bucket that they don't own. The values for SBAI and SBAC record the bucket owner's tenant account ID and name, which differs from the tenant account ID and name of the client recorded in S3AI and SACC.

```
2017-09-20T22:53:15.876415
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):53244][SAIP(IPAD):"10.96.112.26"]\[S3AI
\(CSTR\):"17915054115450519830"\]\[SACC\(CSTR\):"s3-account-
b"\][S3AK(CSTR):"SGKHpoblWlP_kBkqSCbTi754Ls8lBUog67I2LlSiUg=="][SUSR(CSTR)
:"urn:sgws:identity::17915054115450519830:root"]\[SBAI\(CSTR\):"4397929817
8977966408"\]\[SBAC\(CSTR\):"s3-account-a"\][S3BK(CSTR):"bucket-
anonymous"][S3KY(CSTR):"Hello.txt"][CBID(UI64):0x83D70C6F1F662B02][CSIZ(UI
64):12][AVER(UI32):10][ATIM(UI64):1505947995876415][ATYP(FC32):SGET][ANID(
UI32):12272050][AMID(FC32):S3RQ][ATID(UI64):6888780247515624902]]
```

**Example: S3 Select on an object**

When an S3 client issues an S3 Select query on an object, audit messages are generated and saved to the audit log.

Note that not all audit messages generated during a transaction are listed in the example below. Only those related to the S3 Select transaction (SelectObjectContent) are listed.

Each query results in two audit messages: one that performs the authorization of the S3 Select request (the S3SR field is set to "select") and a subsequent standard GET operation that retrieves the data from storage during processing.

```
2021-11-08T15:35:30.750038
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1636385730715700][TIME(UI64):29173][SAI
P(IPAD):"192.168.7.44"][S3AI(CSTR):"63147909414576125820"][SACC(CSTR):"Ten
ant1636027116"][S3AK(CSTR):"AUFD1XNVZ905F3TW7KSU"][SUSR(CSTR):"urn:sgws:id
entity::63147909414576125820:root"][SBAI(CSTR):"63147909414576125820"][SBA
C(CSTR):"Tenant1636027116"][S3BK(CSTR):"619c0755-9e38-42e0-a614-
05064f74126d"][S3KY(CSTR):"SUB-
EST2020_ALL.csv"][CBID(UI64):0x0496F0408A721171][UUID(CSTR):"D64B1A4A-
9F01-4EE7-B133-
08842A099628"][CSIZ(UI64):0][S3SR(CSTR):"select"][AVER(UI32):10][ATIM(UI64
):1636385730750038][ATYP(FC32):SPOS][ANID(UI32):12601166][AMID(FC32):S3RQ]
[ATID(UI64):1363009709396895985]]
```

```
2021-11-08T15:35:32.604886
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1636383069486504][TIME(UI64):430690][SA
IP(IPAD):"192.168.7.44"][HTRH(CSTR):"{\"x-forwarded-
for\":\"unix:\"}"][S3AI(CSTR):"63147909414576125820"][SACC(CSTR):"Tenant16
36027116"][S3AK(CSTR):"AUFD1XNVZ905F3TW7KSU"][SUSR(CSTR):"urn:sgws:identit
y::63147909414576125820:root"][SBAI(CSTR):"63147909414576125820"][SBAC(CST
R):"Tenant1636027116"][S3BK(CSTR):"619c0755-9e38-42e0-a614-
05064f74126d"][S3KY(CSTR):"SUB-
EST2020_ALL.csv"][CBID(UI64):0x0496F0408A721171][UUID(CSTR):"D64B1A4A-
9F01-4EE7-B133-
08842A099628"][CSIZ(UI64):10185581][MTME(UI64):1636380348695262][AVER(UI32
):10][ATIM(UI64):1636385732604886][ATYP(FC32):SGET][ANID(UI32):12733063][A
MID(FC32):S3RQ][ATID(UI64):16562288121152341130]]
```

## Metadata update messages

Audit messages are generated when an S3 client updates an object's metadata.

### S3 metadata update audit messages

| Code | Name | Description | Trace | See |
|---|---|---|---|---|
| SUPD | S3 Metadata Updated | Generated when an S3 client updates the metadata for an ingested object. | CBID, S3KY, HTRH | SUPD: S3 Metadata Updated |

### Example: S3 metadata update

The example shows a successful transaction to update the metadata for an existing S3 object.

**SUPD: S3 Metadata Update**

The S3 client makes a request (SUPD) to update the specified metadata (`x-amz-meta-*`) for the S3 object (S3KY). In this example, request headers are included in the field HTRH because it has been configured as an audit protocol header (*Configuration* > **Monitoring** > **Audit and syslog server**). See Configure log management and external syslog server.

```
2017-07-11T21:54:03.157462
[AUDT:[RSLT(FC32):SUCS][TIME(UI64):17631][SAIP(IPAD):"10.96.100.254"]
[HTRH(CSTR):"{\"accept-encoding\":\"identity\",\"authorization\":\"AWS
LIUF17FGJARQHPY2E761:jul/hnZs/uNY+aVvV0lTSYhEGts=\",
\"content-length\":\"0\",\"date\":\"Tue, 11 Jul 2017 21:54:03
GMT\",\"host\":\"10.96.99.163:18082\",
\"user-agent\":\"aws-cli/1.9.20 Python/2.7.6 Linux/3.13.0-119-generic
botocore/1.3.20\",
\"x-amz-copy-source\":\"/testbkt1/testobj1\",\"x-amz-metadata-
directive\":\"REPLACE\",\"x-amz-meta-city\":\"Vancouver\"}"]
[S3AI(CSTR):"20956855414285633225"][SACC(CSTR):"acct1"][S3AK(CSTR):"SGKHyy
v9ZQqWRbJSQc5vI7mgioJwrdplShE02AUaww=="]
[SUSR(CSTR):"urn:sgws:identity::20956855414285633225:root"]
[SBAI(CSTR):"20956855414285633225"][SBAC(CSTR):"acct1"][S3BK(CSTR):"testbk
t1"]
[S3KY(CSTR):"testobj1"][CBID(UI64):0xCB1D5C213434DD48][CSIZ(UI64):10][AVER
(UI32):10]
[ATIM(UI64):1499810043157462][ATYP(FC32):SUPD][ANID(UI32):12258396][AMID(F
C32):S3RQ]
[ATID(UI64):8987436599021955788]]
```

# Audit messages

## Audit message descriptions

Detailed descriptions of audit messages returned by the system are listed in the following sections. Each audit message is first listed in a table that groups related messages by the class of activity that the message represents. These groupings are useful both for understanding the types of activities that are audited, and for selecting the desired type of audit message filtering.

The audit messages are also listed alphabetically by their four-character codes. This alphabetic list enables you to find information about specific messages.

The four-character codes used throughout this chapter are the ATYP values found in the audit messages as shown in the following example message:

```
2014-07-17T03:50:47.484627
\[AUDT:[RSLT(FC32):VRGN][AVER(UI32):10][ATIM(UI64):1405569047484627][ATYP\
(FC32\):SYSU][ANID(UI32):11627225][AMID(FC32):ARNI][ATID(UI64):94457363265
00603516]]
```

For information about setting audit message levels, changing log destinations, and using an external syslog server for your audit information, see Configure log management and external syslog server

## Audit message categories

### System audit messages

The audit messages belonging to the system audit category are used for events related to the auditing system itself, grid node states, system-wide task activity (grid tasks), and service backup operations.

| Code | Message title and description | See |
|---|---|---|
| ECMC | Missing Erasure-Coded Data Fragment: Indicates that a missing erasure-coded data fragment has been detected. | ECMC: Missing Erasure-Coded Data Fragment |
| ECOC | Corrupt Erasure-Coded Data Fragment: Indicates that a corrupt erasure-coded data fragment has been detected. | ECOC: Corrupt Erasure-Coded Data Fragment |
| ETAF | Security Authentication Failed: A connection attempt using Transport Layer Security (TLS) failed. | ETAF: Security Authentication Failed |
| GNRG | GNDS Registration: A service updated or registered information about itself in the StorageGRID system. | GNRG: GNDS Registration |
| GNUR | GNDS Unregistration: A service has unregistered itself from the StorageGRID system. | GNUR: GNDS Unregistration |
| GTED | Grid Task Ended: The CMN service finished processing the grid task. | GTED: Grid Task Ended |
| GTST | Grid Task Started: The CMN service started to process the grid task. | GTST: Grid Task Started |
| GTSU | Grid Task Submitted: A grid task was submitted to the CMN service. | GTSU: Grid Task Submitted |
| LLST | Location Lost: This audit message is generated when a location is lost. | LLST: Location Lost |

| Code | Message title and description | See |
|---|---|---|
| OLST | Object Lost: A requested object cannot be located within the StorageGRID system. | OLST: System Detected Lost Object |
| SADD | Security Audit Disable: Audit message logging was turned off. | SADD: Security Audit Disable |
| SADE | Security Audit Enable: Audit message logging has been restored. | SADE: Security Audit Enable |
| SVRF | Object Store Verify Fail: A content block failed verification checks. | SVRF: Object Store Verify Fail |
| SVRU | Object Store Verify Unknown: Unexpected object data detected in the object store. | SVRU: Object Store Verify Unknown |
| SYSD | Node Stop: A shutdown was requested. | SYSD: Node Stop |
| SYST | Node Stopping: A service initiated a graceful stop. | SYST: Node Stopping |
| SYSU | Node Start: A service started; the nature of the previous shutdown is indicated in the message. | SYSU: Node Start |

#### Object storage audit messages

The audit messages belonging to the object storage audit category are used for events related to the storage and management of objects within the StorageGRID system. These include object storage and retrievals, grid-node to grid-node transfers, and verifications.

Audit codes are removed from the product and documentation as features are deprecated. If you encounter an audit code that is not listed here, check the previous versions of this topic for older SG releases. For example, StorageGRID 11.8 object storage audit messages.

| Code | Description | See |
|---|---|---|
| BROR | Bucket Read Only Request: A bucket entered or exited read-only mode. | BROR: Bucket Read Only Request |
| CBSE | Object Send End: The source entity completed a grid-node to grid-node data transfer operation. | CBSE: Object Send End |
| CBRE | Object Receive End: The destination entity completed a grid-node to grid-node data transfer operation. | CBRE: Object Receive End |

| Code | Description | See |
|---|---|---|
| CGRR | Cross-Grid Replication Request: StorageGRID attempted a cross-grid replication operation to replicate objects between buckets in a grid federation connection. | CGRR: Cross-Grid Replication Request |
| EBDL | Empty Bucket Delete: The ILM scanner deleted an object in a bucket that is deleting all objects (performing an empty bucket operation). | EBDL: Empty Bucket Delete |
| EBKR | Empty Bucket Request: A user sent a request to turn empty bucket on or off (that is, to delete bucket objects or to stop deleting objects). | EBKR: Empty Bucket Request |
| SCMT | Object Store Commit: A content block was completely stored and verified, and can now be requested. | SCMT: Object Store Commit Request |
| SREM | Object Store Remove: A content block was deleted from a grid node, and can no longer be requested directly. | SREM: Object Store Remove |

#### Client read audit messages

Client read audit messages are logged when an S3 client application makes a request to retrieve an object.

| Code | Description | Used by | See |
|---|---|---|---|
| S3SL | S3 Select request: Logs a completion after an S3 Select request has been returned to the client. The S3SL message can include error message and error code details. The request might not have been successful. | S3 client | S3SL: S3 Select request |
| SGET | S3 GET: Logs a successful transaction to retrieve an object or list the objects in a bucket.<br><br>**Note:** If the transaction operates on a subresource, the audit message will include the field S3SR. | S3 client | SGET: S3 GET |
| SHEA | S3 HEAD: Logs a successful transaction to check for the existence of an object or bucket. | S3 client | SHEA: S3 HEAD |

#### Client write audit messages

Client write audit messages are logged when an S3 client application makes a request to create or modify an object.

| Code | Description | Used by | See |
|---|---|---|---|
| OVWR | Object Overwrite: Logs a transaction to overwrite one object with another object. | S3 client | OVWR: Object Overwrite |
| SDEL | S3 DELETE: Logs a successful transaction to delete an object or bucket.<br><br>**Note:** If the transaction operates on a subresource, the audit message will include the field S3SR. | S3 client | SDEL: S3 DELETE |
| SPOS | S3 POST: Logs a successful transaction to restore an object from AWS Glacier storage to a Cloud Storage Pool. | S3 client | SPOS: S3 POST |
| SPUT | S3 PUT: Logs a successful transaction to create a new object or bucket.<br><br>**Note:** If the transaction operates on a subresource, the audit message will include the field S3SR. | S3 client | SPUT: S3 PUT |
| SUPD | S3 Metadata Updated: Logs a successful transaction to update the metadata for an existing object or bucket. | S3 client | SUPD: S3 Metadata Updated |

#### Management audit message

The Management category logs user requests to the Management API.

| Code | Message title and description | See |
|---|---|---|
| MGAU | Management API audit message: A log of user requests. | MGAU: Management audit message |

#### ILM audit messages

The audit messages belonging to the ILM audit category are used for events related to information lifecycle management (ILM) operations.

| Code | Message title and description | See |
|---|---|---|
| IDEL | ILM Initiated Delete: This audit message is generated when ILM starts the process of deleting an object. | IDEL: ILM Initiated Delete |
| LKCU | Overwritten Object Cleanup. This audit message is generated when an overwritten object is automatically removed to free up storage space. | LKCU: Overwritten Object Cleanup |

| Code | Message title and description | See |
|---|---|---|
| ORLM | Object Rules Met: This audit message is generated when object data is stored as specified by the ILM rules. | ORLM: Object Rules Met |

## Audit message reference

### BROR: Bucket Read Only Request

The LDR service generates this audit message when a bucket enters or exits read-only mode. For example, a bucket enters read-only mode while all objects are being deleted.

| Code | Field | Description |
|---|---|---|
| BKHD | Bucket UUID | The bucket ID. |
| BROV | Bucket read-only request value | Whether the bucket is being made read-only or is leaving the read-only state (1 = read-only, 0 = not-read-only). |
| BROS | Bucket read-only reason | The reason the bucket is being made read-only or leaving the read-only state. For example, emptyBucket. |
| S3AI | S3 tenant account ID | The ID of the tenant account that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 bucket | The S3 bucket name. |

### CBRB: Object Receive Begin

During normal system operations, content blocks are continuously transferred between different nodes as data is accessed, replicated and retained. When transfer of a content block from one node to another is initiated, this message is issued by the destination entity.

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique identifier of the node-to-node session/connection. |
| CBID | Content Block Identifier | The unique identifier of the content block being transferred. |
| CTDR | Transfer Direction | Indicates if the CBID transfer was push-initiated or pull-initiated:<br><br>PUSH: The transfer operation was requested by the sending entity.<br><br>PULL: The transfer operation was requested by the receiving entity. |

| Code | Field | Description |
|---|---|---|
| CTSR | Source Entity | The node ID of the source (sender) of the CBID transfer. |
| CTDS | Destination Entity | The node ID of the destination (receiver) of the CBID transfer. |
| CTSS | Start Sequence Count | Indicates the first sequence count requested. If successful, the transfer begins from this sequence count. |
| CTES | Expected End Sequence Count | Indicates the last sequence count requested. If successful, the transfer is considered complete when this sequence count has been received. |
| RSLT | Transfer Start Status | Status at the time the transfer was started:<br><br>SUCS: Transfer started successfully. |

This audit message means a node-to-node data transfer operation was initiated on a single piece of content, as identified by its Content Block Identifier. The operation requests data from "Start Sequence Count" to "Expected End Sequence Count". Sending and receiving nodes are identified by their node IDs. This information can be used to track system data flow, and when combined with storage audit messages, to verify replica counts.

**CBRE: Object Receive End**

When transfer of a content block from one node to another is completed, this message is issued by the destination entity.

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique identifier of the node-to-node session/connection. |
| CBID | Content Block Identifier | The unique identifier of the content block being transferred. |
| CTDR | Transfer Direction | Indicates if the CBID transfer was push-initiated or pull-initiated:<br><br>PUSH: The transfer operation was requested by the sending entity.<br><br>PULL: The transfer operation was requested by the receiving entity. |
| CTSR | Source Entity | The node ID of the source (sender) of the CBID transfer. |
| CTDS | Destination Entity | The node ID of the destination (receiver) of the CBID transfer. |
| CTSS | Start Sequence Count | Indicates the sequence count on which the transfer started. |

| Code | Field | Description |
|---|---|---|
| CTAS | Actual End Sequence Count | Indicates the last sequence count successfully transferred. If the Actual End Sequence Count is the same as the Start Sequence Count, and the Transfer Result was not successful, no data was exchanged. |
| RSLT | Transfer Result | The result of the transfer operation (from the perspective of the sending entity):<br><br>SUCS: transfer successfully completed; all requested sequence counts were sent.<br><br>CONL: connection lost during transfer<br><br>CTMO: connection timed-out during establishment or transfer<br><br>UNRE: destination node ID unreachable<br><br>CRPT: transfer ended due to reception of corrupt or invalid data |

This audit message means a node-to-node data transfer operation was completed. If the Transfer Result was successful, the operation transferred data from "Start Sequence Count" to "Actual End Sequence Count". Sending and receiving nodes are identified by their node IDs. This information can be used to track system data flow and to locate, tabulate, and analyze errors. When combined with storage audit messages, it can also be used to verify replica counts.

**CBSB: Object Send Begin**

During normal system operations, content blocks are continuously transferred between different nodes as data is accessed, replicated and retained. When transfer of a content block from one node to another is initiated, this message is issued by the source entity.

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique identifier of the node-to-node session/connection. |
| CBID | Content Block Identifier | The unique identifier of the content block being transferred. |
| CTDR | Transfer Direction | Indicates if the CBID transfer was push-initiated or pull-initiated:<br><br>PUSH: The transfer operation was requested by the sending entity.<br><br>PULL: The transfer operation was requested by the receiving entity. |
| CTSR | Source Entity | The node ID of the source (sender) of the CBID transfer. |
| CTDS | Destination Entity | The node ID of the destination (receiver) of the CBID transfer. |

| Code | Field | Description |
|---|---|---|
| CTSS | Start Sequence Count | Indicates the first sequence count requested. If successful, the transfer begins from this sequence count. |
| CTES | Expected End Sequence Count | Indicates the last sequence count requested. If successful, the transfer is considered complete when this sequence count has been received. |
| RSLT | Transfer Start Status | Status at the time the transfer was started:<br><br>SUCS: transfer started successfully. |

This audit message means a node-to-node data transfer operation was initiated on a single piece of content, as identified by its Content Block Identifier. The operation requests data from "Start Sequence Count" to "Expected End Sequence Count". Sending and receiving nodes are identified by their node IDs. This information can be used to track system data flow, and when combined with storage audit messages, to verify replica counts.

**CBSE: Object Send End**

When transfer of a content block from one node to another is completed, this message is issued by the source entity.

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique identifier of the node-to-node session/connection. |
| CBID | Content Block Identifier | The unique identifier of the content block being transferred. |
| CTDR | Transfer Direction | Indicates if the CBID transfer was push-initiated or pull-initiated:<br><br>PUSH: The transfer operation was requested by the sending entity.<br><br>PULL: The transfer operation was requested by the receiving entity. |
| CTSR | Source Entity | The node ID of the source (sender) of the CBID transfer. |
| CTDS | Destination Entity | The node ID of the destination (receiver) of the CBID transfer. |
| CTSS | Start Sequence Count | Indicates the sequence count on which the transfer started. |
| CTAS | Actual End Sequence Count | Indicates the last sequence count successfully transferred. If the Actual End Sequence Count is the same as the Start Sequence Count, and the Transfer Result was not successful, no data was exchanged. |

| Code | Field | Description |
|---|---|---|
| RSLT | Transfer Result | The result of the transfer operation (from the perspective of the sending entity):<br><br>SUCS: Transfer successfully completed; all requested sequence counts were sent.<br><br>CONL: connection lost during transfer<br><br>CTMO: connection timed-out during establishment or transfer<br><br>UNRE: destination node ID unreachable<br><br>CRPT: transfer ended due to reception of corrupt or invalid data |

This audit message means a node-to-node data transfer operation was completed. If the Transfer Result was successful, the operation transferred data from "Start Sequence Count" to "Actual End Sequence Count". Sending and receiving nodes are identified by their node IDs. This information can be used to track system data flow and to locate, tabulate, and analyze errors. When combined with storage audit messages, it can also be used to verify replica counts.

#### CGRR: Cross-Grid Replication Request

This message is generated when StorageGRID attempts a cross-grid replication operation to replicate objects between buckets in a grid federation connection.

| Code | Field | Description |
|---|---|---|
| CSIZ | Object Size | The size of the object in bytes.<br><br>The CSIZ attribute was introduced in StorageGRID 11.8. As a result, cross-grid replication requests spanning a StorageGRID 11.7 to 11.8 upgrade might have an inaccurate total object size. |
| S3AI | S3 tenant account ID | The ID of the tenant account that owns the bucket from which the object is being replicated. |
| GFID | Grid federation connection ID | The ID of the grid federation connection being used for cross-grid replication. |
| OPER | CGR operation | The type of cross-grid replication operation that was attempted:<br><br>• 0 = Replicate object<br>• 1 = Replicate multipart object<br>• 2 = Replicate delete marker |
| S3BK | S3 bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. |

| Code | Field | Description |
|---|---|---|
| VSID | Version ID | The version ID of the specific version of an object that was being replicated. |
| RSLT | Result Code | Returns successful (SUCS) or general error (GERR). |

**EBDL: Empty Bucket Delete**

The ILM scanner deleted an object in a bucket that is deleting all objects (performing an empty bucket operation).

| Code | Field | Description |
|---|---|---|
| CSIZ | Object Size | The size of the object in bytes. |
| PATH | S3 Bucket/Key | The S3 bucket name and S3 key name. |
| SEGC | Container UUID | UUID of the container for the segmented object. This value is available only if the object is segmented. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| RSLT | Result of the delete operation | The result of event, process, or transaction. If is not relevant for a message, NONE is used rather than SUCS so that the message is not accidentally filtered. |

**EBKR: Empty Bucket Request**

This message indicates a user sent a request to turn empty bucket on or off (that is, to delete bucket objects or to stop deleting objects).

| Code | Field | Description |
|---|---|---|
| BUID | Bucket UUID | The bucket ID. |
| EBJS | Empty Bucket JSON Configuration | Contains the JSON representing the current Empty Bucket configuration. |
| S3AI | S3 tenant account ID | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |

#### ECMC: Missing Erasure-Coded Data Fragment

This audit message indicates that the system has detected a missing erasure-coded data fragment.

| Code | Field | Description |
|---|---|---|
| VCMC | VCS ID | The name of the VCS that contains the missing chunk. |
| MCID | Chunk ID | The identifier of the missing erasure-coded fragment. |
| RSLT | Result | This field has the value 'NONE'. RSLT is a mandatory message field, but is not relevant for this particular message. 'NONE' is used rather than 'SUCS' so that this message is not filtered. |

#### ECOC: Corrupt Erasure-Coded Data Fragment

This audit message indicates that the system has detected a corrupt erasure-coded data fragment.

| Code | Field | Description |
|---|---|---|
| VCCO | VCS ID | The name of the VCS that contains the corrupt chunk. |
| VLID | Volume ID | The RangeDB Volume that contains the corrupt erasure-coded fragment. |
| CCID | Chunk ID | The identifier of the corrupt erasure-coded fragment. |
| RSLT | Result | This field has the value 'NONE'. RSLT is a mandatory message field, but is not relevant for this particular message. 'NONE' is used rather than 'SUCS' so that this message is not filtered. |

#### ETAF: Security Authentication Failed

This message is generated when a connection attempt using Transport Layer Security (TLS) has failed.

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection over which the authentication failed. |
| RUID | User Identity | A service dependent identifier representing the identity of the remote user. |

| Code | Field | Description |
|---|---|---|
| RSLT | Reason Code | The reason for the failure:<br><br>SCNI: Secure connection establishment failed.<br><br>CERM: Certificate was missing.<br><br>CERT: Certificate was invalid.<br><br>CERE: Certificate was expired.<br><br>CERR: Certificate was revoked.<br><br>CSGN: Certificate signature was invalid.<br><br>CSGU: Certificate signer was unknown.<br><br>UCRM: User credentials were missing.<br><br>UCRI: User credentials were invalid.<br><br>UCRU: User credentials were disallowed.<br><br>TOUT: Authentication timed out. |

When a connection is established to a secure service that uses TLS, the credentials of the remote entity are verified using the TLS profile and additional logic built into the service. If this authentication fails due to invalid, unexpected, or disallowed certificates or credentials, an audit message is logged. This enables queries for unauthorized access attempts and other security-related connection problems.

The message could result from a remote entity having an incorrect configuration, or from attempts to present invalid or disallowed credentials to the system. This audit message should be monitored to detect attempts to gain unauthorized access to the system.

**GNRG: GNDS Registration**

The CMN service generates this audit message when a service has updated or registered information about itself in the StorageGRID system.

| Code | Field | Description |
|---|---|---|
| RSLT | Result | The result of the update request:<br><br>• SUCS: Successful<br>• SUNV: Service Unavailable<br>• GERR: Other failure |
| GNID | Node ID | The node ID of the service that initiated the update request. |
| GNTP | Device Type | The grid node's device type (for example, BLDR for an LDR service). |

| Code | Field | Description |
|---|---|---|
| GNDV | Device Model version | The string identifying the grid node's device model version in the DMDL bundle. |
| GNGP | Group | The group to which the grid node belongs (in the context of link costs and service-query ranking). |
| GNIA | IP Address | The grid node's IP address. |

This message is generated whenever a grid node updates its entry in the Grid Nodes Bundle.

#### GNUR: GNDS Unregistration

The CMN service generates this audit message when a service has unregistered information about itself from the StorageGRID system.

| Code | Field | Description |
|---|---|---|
| RSLT | Result | The result of the update request:<br>• SUCS: Successful<br>• SUNV: Service Unavailable<br>• GERR: Other failure |
| GNID | Node ID | The node ID of the service that initiated the update request. |

#### GTED: Grid Task Ended

This audit message indicates that the CMN service has finished processing the specified grid task and has moved the task to the Historical table. If the result is SUCS, ABRT, or ROLF, there will be a corresponding Grid Task Started audit message. The other results indicate that processing of this grid task never started.

| Code | Field | Description |
|---|---|---|
| TSID | Task ID | This field uniquely identifies a generated grid task and allows the grid task to be managed over its lifecycle.<br><br>**Note:** The Task ID is assigned at the time that a grid task is generated, not the time that it is submitted. It is possible for a given grid task to be submitted multiple times, and in this case the Task ID field is not sufficient to uniquely link the Submitted, Started, and Ended audit messages. |

| Code | Field | Description |
|---|---|---|
| RSLT | Result | The final status result of the grid task:<br>• SUCS: The grid task completed successfully.<br>• ABRT: The grid task was terminated without a rollback error.<br>• ROLF: The grid task was terminated and was unable to complete the rollback process.<br>• CANC: The grid task was canceled by the user before it was started.<br>• EXPR: The grid task expired before it was started.<br>• IVLD: The grid task was invalid.<br>• AUTH: The grid task was unauthorized.<br>• DUPL: The grid task was rejected as a duplicate. |

**GTST: Grid Task Started**

This audit message indicates that the CMN service has started to process the specified grid task. The audit message immediately follows the Grid Task Submitted message for grid tasks initiated by the internal Grid Task Submission service and selected for automatic activation. For grid tasks submitted into the Pending table, this message is generated when the user starts the grid task.

| Code | Field | Description |
|---|---|---|
| TSID | Task ID | This field uniquely identifies a generated grid task and allows the task to be managed over its lifecycle.<br>**Note:** The Task ID is assigned at the time that a grid task is generated, not the time that it is submitted. It is possible for a given grid task to be submitted multiple times, and in this case the Task ID field is not sufficient to uniquely link the Submitted, Started, and Ended audit messages. |
| RSLT | Result | The result. This field has only one value:<br>• SUCS: The grid task was started successfully. |

**GTSU: Grid Task Submitted**

This audit message indicates that a grid task has been submitted to the CMN service.

| Code | Field | Description |
|---|---|---|
| TSID | Task ID | Uniquely identifies a generated grid task and allows the task to be managed over its lifecycle.<br><br>**Note:** The Task ID is assigned at the time that a grid task is generated, not the time that it is submitted. It is possible for a given grid task to be submitted multiple times, and in this case the Task ID field is not sufficient to uniquely link the Submitted, Started, and Ended audit messages. |
| TTYP | Task Type | The type of grid task. |
| TVER | Task Version | A number indicating the version of the grid task. |
| TDSC | Task Description | A human-readable description of the grid task. |
| VATS | Valid After Timestamp | The earliest time (UINT64 microseconds from January 1, 1970 - UNIX time) at which the grid task is valid. |
| VBTS | Valid Before Timestamp | The latest time (UINT64 microseconds from January 1, 1970 - UNIX time) at which the grid task is valid. |
| TSRC | Source | The source of the task:<br><br>• TXTB: The grid task was submitted through the StorageGRID system as a signed text block.<br>• GRID: The grid task was submitted through the internal Grid Task Submission Service. |
| ACTV | Activation Type | The type of activation:<br><br>• AUTO: The grid task was submitted for automatic activation.<br>• PEND: The grid task was submitted into the pending table. This is the only possibility for the TXTB source. |
| RSLT | Result | The result of the submission:<br><br>• SUCS: The grid task was submitted successfully.<br>• FAIL: The task has been moved directly to the historical table. |

#### IDEL: ILM Initiated Delete

This message is generated when ILM starts the process of deleting an object.

The IDEL message is generated in either of these situations:

- **For objects in compliant S3 buckets**: This message is generated when ILM starts the process of auto-deleting an object because its retention period has expired (assuming the auto-delete setting is enabled

and legal hold is off).

- **For objects in non-compliant S3 buckets**. This message is generated when ILM starts the process of deleting an object because no placement instructions in the active ILM policies currently apply to the object.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The CBID of the object. |
| CMPA | Compliance: Auto delete | For objects in compliant S3 buckets only. 0 (false) or 1 (true), indicating whether a compliant object should be deleted automatically when its retention period ends, unless the bucket is under a legal hold. |
| CMPL | Compliance: Legal hold | For objects in compliant S3 buckets only. 0 (false) or 1 (true), indicating whether the bucket is currently under a legal hold. |
| CMPR | Compliance: Retention period | For objects in compliant S3 buckets only. The length of the object's retention period in minutes. |
| CTME | Compliance: Ingest time | For objects in compliant S3 buckets only. The object's ingest time. You can add the retention period in minutes to this value to determine when the object can be deleted from the bucket. |
| DMRK | Delete Marker Version ID | The version ID of the delete marker created when deleting an object from a versioned bucket. Operations on buckets don't include this field. |
| CSIZ | Content size | The size of the object in bytes. |
| LOCS | Locations | The storage location of object data within the StorageGRID system. The value for LOCS is "" if the object has no locations (for example, it has been deleted).<br><br>CLEC: for erasure-coded objects, the erasure-coding profile ID and the erasure coding group ID that is applied to the object's data.<br><br>CLDI: for replicated objects, the LDR node ID and the volume ID of the object's location.<br><br>CLNL: ARC node ID of the object's location if the object data is archived. |
| PATH | S3 Bucket/Key | The S3 bucket name and S3 key name. |
| RSLT | Result | The result of the ILM operation.<br><br>SUCS: The ILM operation was successful. |

| Code | Field | Description |
|---|---|---|
| RULE | Rules Label | • If an object in a compliant S3 bucket is being deleted automatically because its retention period has expired, this field is blank.<br>• If the object is being deleted because there are no more placement instructions that currently apply to the object, this field shows the human-readable label of the last ILM rule that applied to the object. |
| SGRP | Site (Group) | If present, the object was deleted at the site specified, which is not the site where the object was ingested. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object that was deleted. Operations on buckets and objects in unversioned buckets don't include this field. |

#### LKCU: Overwritten Object Cleanup

This message is generated when StorageGRID removes an overwritten object that previously required cleanup to free up storage space. An object is overwritten when an S3 client writes an object to a path already containing a object. The removal process occurs automatically and in the background.

| Code | Field | Description |
|---|---|---|
| CSIZ | Content size | The size of the object in bytes. |
| LTYP | Type of cleanup | *Internal use only.* |
| LUID | Removed Object UUID | The identifier of the object that was removed. |
| PATH | S3 Bucket/Key | The S3 bucket name and S3 key name. |
| SEGC | Container UUID | UUID of the container for the segmented object. This value is available only if the object is segmented. |
| UUID | Universally Unique Identifier | The identifier of the object that still exists. This value is available only if the object has not been deleted. |

#### LKDM: Leaked Object Cleanup

This message is generated when a leaked chunk has been cleaned or deleted. A chunk can be part of a replicated object or an erasure-encoded object.

| Code | Field | Description |
|---|---|---|
| CLOC | Chunk location | The file path of the leaked chunk that got deleted. |
| CTYP | Chunk type | Type of chunk:<br><br>`ec: Erasure-coded object chunk`<br><br>`repl: Replicated object chunk` |
| LTYP | Leak type | The five types of leaks that can be detected:<br><br>`object_leaked: Object doesn’t exist in the grid`<br><br>`location_leaked: Object exists in the grid, but found location doesn’t belong to object`<br><br>`mup_seg_leaked: Multipart upload was stopped or not completed, and the segment/part was left out`<br><br>`segment_leaked: Parent UUID/CBID (associated container object) is valid but doesn’t contain this segment`<br><br>`no_parent: Container object is deleted, but object segment was left out and not deleted` |
| CTIM | Chunk create time | Time the leaked chunk was created. |
| UUID | Universally Unique Identifier | The identifier of the object the chunk belongs to. |
| CBID | Content Block Identifier | CBID of the object the leaked chunk belongs to. |
| CSIZ | Content size | The size of the chunk in bytes. |

#### LLST: Location Lost

This message is generated whenever a location for an object copy (replicated or erasure-coded) can’t be found.

| Code | Field | Description |
|---|---|---|
| CBIL | CBID | The affected CBID. |

| Code | Field | Description |
|---|---|---|
| ECPR | Erasure-Coding Profile | For erasure-coded object data. The ID of the erasure-coding profile used. |
| LTYP | Location Type | CLDI (Online): For replicated object data<br><br>CLEC (Online): For erasure-coded object data<br><br>CLNL (Nearline): For archived replicated object data |
| NOID | Source Node ID | The node ID on which the locations were lost. |
| PCLD | Path to replicated object | The complete path to the disk location of the lost object data. Only returned when LTYP has a value of CLDI (that is, for replicated objects).<br><br>Takes the form `/var/local/rangedb/2/p/13/13/00oJs6X%{h{U)SeUFxE@` |
| RSLT | Result | Always NONE. RSLT is a mandatory message field, but is not relevant for this message. NONE is used rather than SUCS so that this message is not filtered. |
| TSRC | Triggering Source | USER: User triggered<br><br>SYST: System triggered |
| UUID | Universally Unique ID | The identifier of the affected object in the StorageGRID system. |

**MGAU: Management audit message**

The Management category logs user requests to the Management API. Every HTTP request that is not a GET or HEAD request to a valid API URI logs a response containing the username, IP, and type of request to the API. Invalid API URIs (such as /api/v3-authorize) and invalid requests to valid API URIs are not logged.

| Code | Field | Description |
|---|---|---|
| MDIP | Destination IP Address | The server (destination) IP address. |
| MDNA | Domain name | The host domain name. |
| MPAT | Request PATH | The request path. |
| MPQP | Request query parameters | The query parameters for the request. |

| Code | Field | Description |
|---|---|---|
| MRBD | Request body | The content of the request body. While the response body is logged by default, the request body is logged in certain cases when the response body is empty. Because the following information is not available in the response body, it is taken from the request body for the following POST methods:<br>• Username and account ID in **POST authorize**<br>• New subnets configuration in **POST /grid/grid-networks/update**<br>• New NTP servers in **POST /grid/ntp-servers/update**<br>• Decommissioned server IDs in **POST /grid/servers/decommission**<br>**Note:** Sensitive information is either deleted (for example, an S3 access key) or masked with asterisks (for example, a password). |
| MRMD | Request method | The HTTP request method:<br>• POST<br>• PUT<br>• DELETE<br>• PATCH |
| MRSC | Response code | The response code. |
| MRSP | Response body | The content of the response (the response body) is logged by default.<br>**Note:** Sensitive information is either deleted (for example, an S3 access key) or masked with asterisks (for example, a password). |
| MSIP | Source IP address | The client (source) IP address. |
| MUUN | User URN | The URN (uniform resource name) of the user who sent the request. |
| RSLT | Result | Returns successful (SUCS) or the error reported by the backend. |

#### OLST: System Detected Lost Object

This message is generated when the DDS service can’t locate any copies of an object within the StorageGRID system.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The CBID of the lost object. |

| Code | Field | Description |
|---|---|---|
| NOID | Node ID | If available, the last known direct or near-line location of the lost object. It is possible to have just the Node ID without a Volume ID if the volume information is not available. |
| PATH | S3 Bucket/Key | If available, the S3 bucket name and S3 key name. |
| RSLT | Result | This field has the value NONE. RSLT is a mandatory message field, but is not relevant for this message. NONE is used rather than SUCS so that this message is not filtered. |
| UUID | Universally Unique ID | The identifier of the lost object within the StorageGRID system. |
| VOLI | Volume ID | If available, the Volume ID of the Storage Node for the last known location of the lost object. |

**ORLM: Object Rules Met**

This message is generated when the object is successfully stored and copied as specified by the ILM rules.

> The ORLM message is not generated when an object is successfully stored by the default Make 2 Copies rule if another rule in the policy uses the Object Size advanced filter.

| Code | Field | Description |
|---|---|---|
| BUID | Bucket Header | Bucket ID field. Used for internal operations. Appears only if STAT is PRGD. |
| CBID | Content Block Identifier | The CBID of the object. |
| CSIZ | Content size | The size of the object in bytes. |
| LOCS | Locations | The storage location of object data within the StorageGRID system. The value for LOCS is "" if the object has no locations (for example, it has been deleted).<br><br>CLEC: for erasure-coded objects, the erasure-coding profile ID and the erasure coding group ID that is applied to the object's data.<br><br>CLDI: for replicated objects, the LDR node ID and the volume ID of the object's location.<br><br>CLNL: ARC node ID of the object's location if the object data is archived. |

| Code | Field | Description |
|---|---|---|
| PATH | S3 Bucket/Key | The S3 bucket name and S3 key name. |
| RSLT | Result | The result of the ILM operation.<br><br>SUCS: The ILM operation was successful. |
| RULE | Rules Label | The human-readable label given to the ILM rule applied to this object. |
| SEGC | Container UUID | UUID of the container for the segmented object. This value is available only if the object is segmented. |
| SGCB | Container CBID | CBID of the container for the segmented object. This value is available only for segmented and multipart objects. |
| STAT | Status | The status of ILM operation.<br><br>DONE: ILM operations against the object have completed.<br><br>DFER: The object has been marked for future ILM re-evaluation.<br><br>PRGD: The object has been deleted from the StorageGRID system.<br><br>NLOC: The object data can no longer be found in the StorageGRID system. This status might indicate that all copies of object data are missing or damaged. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of a new object created in a versioned bucket. Operations on buckets and objects in unversioned buckets don’t include this field. |

The ORLM audit message can be issued more than once for a single object. For instance, it is issued whenever one of the following events occur:

- ILM rules for the object are satisfied forever.
- ILM rules for the object are satisfied for this epoch.
- ILM rules have deleted the object.
- The background verification process detects that a copy of replicated object data is corrupt. The StorageGRID system performs an ILM evaluation to replace the corrupt object.

**Related information**

- Object ingest transactions
- Object delete transactions

#### OVWR: Object Overwrite

This message is generated when an external (client-requested) operation causes one object to be overwritten by another object.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier (new) | The CBID for the new object. |
| CSIZ | Previous Object Size | The size, in bytes, of the object being overwritten. |
| OCBD | Content Block Identifier (previous) | The CBID for the previous object. |
| UUID | Universally Unique ID (new) | The identifier of the new object within the StorageGRID system. |
| OUID | Universally Unique ID (previous) | The identifier for the previous object within the StorageGRID system. |
| PATH | S3 Object Path | The S3 object path used for both the previous and new object |
| RSLT | Result Code | Result of the Object Overwrite transaction. Result is always:<br><br>SUCS: Successful |
| SGRP | Site (Group) | If present, the overwritten object was deleted at the site specified, which is not the site where the overwritten object was ingested. |

#### S3SL: S3 Select request

This message logs a completion after an S3 Select request has been returned to the client. The S3SL message can include error message and error code details. The request might not have been successful.

| Code | Field | Description |
|---|---|---|
| BYSC | Bytes Scanned | Number of bytes scanned (received) from Storage Nodes.<br><br>BYSC and BYPR are likely to be different if the object is compressed. If the object is compressed BYSC would have the compressed byte count and BYPR would be the bytes after decompression. |
| BYPR | Bytes Processed | Number of bytes processed. Indicates how many bytes of "Bytes Scanned" were actually processed or acted upon by an S3 Select job. |

| Code | Field | Description |
|---|---|---|
| BYRT | Bytes Returned | Number of bytes that an S3 Select job returned to the client. |
| REPR | Records Processed | Number of records or rows that an S3 Select job received from Storage Nodes. |
| RERT | Records Returned | Number of records or rows an S3 Select job returned to the client. |
| JOFI | Job Finished | Indicates if the S3 Select job finished processing or not. If this is false, then the job failed to finish and the error fields will likely have data in them. The client might have received partial results, or no results at all. |
| REID | Request ID | Identifier for the S3 Select request. |
| EXTM | Execution Time | The time, in seconds, it took for the S3 Select Job to complete. |
| ERMG | Error Message | Error message that the S3 Select job generated. |
| ERTY | Error Type | Error type that the S3 Select job generated. |
| ERST | Error Stacktrace | Error Stacktrace that the S3 Select job generated. |
| S3BK | S3 bucket | The S3 bucket name. |
| S3AK | S3 Access Key ID (request sender) | The S3 access key ID for the user that sent the request. |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. |

**SADD: Security Audit Disable**

This message indicates that the originating service (node ID) has turned off audit message logging; audit messages are no longer being collected or delivered.

| Code | Field | Description |
|---|---|---|
| AETM | Enable Method | The method used to disable the audit. |
| AEUN | User Name | The user name that executed the command to disable audit logging. |

| Code | Field | Description |
|---|---|---|
| RSLT | Result | This field has the value NONE. RSLT is a mandatory message field, but is not relevant for this message. NONE is used rather than SUCS so that this message is not filtered. |

The message implies that logging was previously enabled, but has now been disabled. This is typically used only during bulk ingest to improve system performance. Following the bulk activity, auditing is restored (SADE) and the capability to disable auditing is then permanently blocked.

#### SADE: Security Audit Enable

This message indicates that the originating service (node ID) has restored audit message logging; audit messages are again being collected and delivered.

| Code | Field | Description |
|---|---|---|
| AETM | Enable Method | The method used to enable the audit. |
| AEUN | User Name | The user name that executed the command to enable audit logging. |
| RSLT | Result | This field has the value NONE. RSLT is a mandatory message field, but is not relevant for this message. NONE is used rather than SUCS so that this message is not filtered. |

The message implies that logging was previously disabled (SADD), but has now been restored. This is typically only used during bulk ingest to improve system performance. Following the bulk activity, auditing is restored and the capability to disable auditing is then permanently blocked.

#### SCMT: Object Store Commit

Grid content is not made available or recognized as stored until it has been committed (meaning it has been stored persistently). Persistently stored content has been completely written to disk, and has passed related integrity checks. This message is issued when a content block is committed to storage.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block committed to permanent storage. |
| RSLT | Result Code | Status at the time the object was stored to disk:<br><br>SUCS: Object successfully stored. |

This message means a given content block has been completely stored and verified, and can now be requested. It can be used to track data flow within the system.

#### SDEL: S3 DELETE

When an S3 client issues a DELETE transaction, a request is made to remove the specified object or bucket, or to remove a bucket/object subresource. This message is issued by the server if the transaction is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. Operations on buckets don't include this field. |
| CNCH | Consistency Control Header | The value of the Consistency-Control HTTP request header, if present in the request. |
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the deleted object in bytes. Operations on buckets don't include this field. |
| DMRK | Delete Marker Version ID | The version ID of the delete marker created when deleting an object from a versioned bucket. Operations on buckets don't include this field. |
| GFID | Grid Federation Connection ID | The connection ID of the grid federation connection associated with a cross-grid replication delete request. Only included in audit logs on the destination grid. |
| GFSA | Grid Federation Source Account ID | The account ID of the tenant on the source grid for a cross-grid replication delete request. Only included in audit logs on the destination grid. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field).<br><br>`x-amz-bypass-governance-retention` is automatically included if it is present in the request. |
| MTME | Last Modified Time | The Unix timestamp, in microseconds, indicating when the object was last modified. |
| RSLT | Result Code | Result of the DELETE transaction. Result is always:<br><br>SUCS: Successful |

| Code | Field | Description |
| --- | --- | --- |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don't include this field. |
| S3SR | S3 Subresource | The bucket or object subresource being operated on, if applicable. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |
| SGRP | Site (Group) | If present, the object was deleted at the site specified, which is not the site where the object was ingested. |
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |

| Code | Field | Description |
|---|---|---|
| UUDM | Universally Unique Identifier for a Delete Marker | The identifier of a delete marker. Audit log messages specify either UUDM or UUID, where UUDM indicates a delete marker created as a result of an object delete request, and UUID indicates an object. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object that was deleted. Operations on buckets and objects in unversioned buckets don't include this field. |

**SGET: S3 GET**

When an S3 client issues a GET transaction, a request is made to retrieve an object or list the objects in a bucket, or to remove a bucket/object subresource. This message is issued by the server if the transaction is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. Operations on buckets don't include this field. |
| CNCH | Consistency Control Header | The value of the Consistency-Control HTTP request header, if present in the request. |
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the retrieved object in bytes. Operations on buckets don't include this field. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field). |
| LITY | ListObjectsV2 | A *v2 format* response was requested. For details, see AWS ListObjectsV2. For GET bucket operations only. |
| NCHD | Number of Children | Includes keys and common prefixes. For GET bucket operations only. |

| Code | Field | Description |
| --- | --- | --- |
| RANG | Range Read | For range read operations only. Indicates the range of bytes that was read by this request. The value after the slash (/) shows the size of the entire object. |
| RSLT | Result Code | Result of the GET transaction. Result is always:<br><br>SUCS: Successful |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don’t include this field. |
| S3SR | S3 Subresource | The bucket or object subresource being operated on, if applicable. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |

| Code | Field | Description |
|---|---|---|
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |
| TRNC | Truncated or Not Truncated | Set to false if all results were returned. Set to true if more results are available to return. For GET bucket operations only. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object that was requested. Operations on buckets and objects in unversioned buckets don’t include this field. |

**SHEA: S3 HEAD**

When an S3 client issues a HEAD operation, a request is made to check for the existence of an object or bucket and retrieve the metadata about an object. This message is issued by the server if the operation is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. Operations on buckets don’t include this field. |
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the checked object in bytes. Operations on buckets don’t include this field. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field). |
| RSLT | Result Code | Result of the GET transaction. Result is always:<br><br>SUCS: Successful |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |

| Code | Field | Description |
|---|---|---|
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don’t include this field. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object that was requested. Operations on buckets and objects in unversioned buckets don’t include this field. |

**SPOS: S3 POST**

When an S3 client issues a POST Object request, this message is issued by the server if the transaction is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. |
| CNCH | Consistency Control Header | The value of the Consistency-Control HTTP request header, if present in the request. |
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the retrieved object in bytes. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field).<br><br>(Not expected for SPOS). |
| RSLT | Result Code | Result of the RestoreObject request. Result is always:<br><br>SUCS: Successful |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don’t include this field. |
| S3SR | S3 Subresource | The bucket or object subresource being operated on, if applicable.<br><br>Set to "select" for an S3 Select operation. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |

| Code | Field | Description |
|---|---|---|
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |
| SRCF | Subresource Configuration | Restore information. |
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object that was requested. Operations on buckets and objects in unversioned buckets don’t include this field. |

**SPUT: S3 PUT**

When an S3 client issues a PUT transaction, a request is made to create a new object or bucket, or to remove a bucket/object subresource. This message is issued by the server if the transaction is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. Operations on buckets don’t include this field. |
| CMPS | Compliance Settings | The compliance settings used when creating the bucket, if present in the request (truncated to the first 1024 characters). |

| Code | Field | Description |
|---|---|---|
| CNCH | Consistency Control Header | The value of the Consistency-Control HTTP request header, if present in the request. |
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the retrieved object in bytes. Operations on buckets don't include this field. |
| GFID | Grid Federation Connection ID | The connection ID of the grid federation connection associated with a cross-grid replication PUT request. Only included in audit logs on the destination grid. |
| GFSA | Grid Federation Source Account ID | The account ID of the tenant on the source grid for a cross-grid replication PUT request. Only included in audit logs on the destination grid. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field).<br><br>`x-amz-bypass-governance-retention` is automatically included if it is present in the request. |
| LKEN | Object Lock Enabled | Value of the request header `x-amz-bucket-object-lock-enabled`, if present in the request. |
| LKLH | Object Lock Legal Hold | Value of the request header `x-amz-object-lock-legal-hold`, if present in the PutObject request. |
| LKMD | Object Lock Retention Mode | Value of the request header `x-amz-object-lock-mode`, if present in the PutObject request. |
| LKRU | Object Lock Retain Until Date | Value of the request header `x-amz-object-lock-retain-until-date`, if present in the PutObject request. Values are limited to within 100 years of the date the object was ingested. |
| MTME | Last Modified Time | The Unix timestamp, in microseconds, indicating when the object was last modified. |
| RSLT | Result Code | Result of the PUT transaction. Result is always:<br><br>SUCS: Successful |

| Code | Field | Description |
| --- | --- | --- |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don’t include this field. |
| S3SR | S3 Subresource | The bucket or object subresource being operated on, if applicable. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |
| SRCF | Subresource Configuration | The new subresource configuration (truncated to the first 1024 characters). |
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |

| Code | Field | Description |
|---|---|---|
| ULID | Upload ID | Included only in SPUT messages for CompleteMultipartUpload operations. Indicates that all parts have been uploaded and assembled. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of a new object created in a versioned bucket. Operations on buckets and objects in unversioned buckets don’t include this field. |
| VSST | Versioning State | The new versioning state of a bucket. Two states are used: "enabled" or "suspended." Operations on objects don’t include this field. |

**SREM: Object Store Remove**

This message is issued when content is removed from persistent storage and is no longer accessible through regular APIs.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block deleted from permanent storage. |
| RSLT | Result Code | Indicates the result of the content removal operations. The only defined value is:<br><br>SUCS: Content removed from persistent storage |

This audit message means a given content block has been deleted from a node and can no longer be requested directly. The message can be used to track the flow of deleted content within the system.

**SUPD: S3 Metadata Updated**

This message is generated by the S3 API when an S3 client updates the metadata for an ingested object. The message is issued by the server if the metadata update is successful.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block requested. If the CBID is unknown, this field is set to 0. Operations on buckets don’t include this field. |
| CNCH | Consistency Control Header | The value of the Consistency-Control HTTP request header, if present in the request, when updating a bucket’s compliance settings. |

| Code | Field | Description |
|---|---|---|
| CNID | Connection Identifier | The unique system identifier for the TCP/IP connection. |
| CSIZ | Content Size | The size of the retrieved object in bytes. Operations on buckets don't include this field. |
| HTRH | HTTP Request Header | List of logged HTTP request header names and values as selected during configuration.<br><br>`X-Forwarded-For` is automatically included if it is present in the request and if the `X-Forwarded-For` value is different from the request sender IP address (SAIP audit field). |
| RSLT | Result Code | Result of the GET transaction. Result is always:<br><br>SUCS: successful |
| S3AI | S3 tenant account ID (request sender) | The tenant account ID of the user who sent the request. An empty value indicates anonymous access. |
| S3AK | S3 Access Key ID (request sender) | The hashed S3 access key ID for the user that sent the request. An empty value indicates anonymous access. |
| S3BK | S3 Bucket | The S3 bucket name. |
| S3KY | S3 Key | The S3 key name, not including the bucket name. Operations on buckets don't include this field. |
| SACC | S3 tenant account name (request sender) | The name of the tenant account for the user who sent the request. Empty for anonymous requests. |
| SAIP | IP address (request sender) | The IP address of the client application that made the request. |
| SBAC | S3 tenant account name (bucket owner) | The tenant account name for the bucket owner. Used to identify cross-account or anonymous access. |
| SBAI | S3 tenant account ID (bucket owner) | The tenant account ID of the owner of the target bucket. Used to identify cross-account or anonymous access. |

| Code | Field | Description |
|---|---|---|
| SUSR | S3 User URN (request sender) | The tenant account ID and the user name of the user making the request. The user can either be a local user or an LDAP user. For example: `urn:sgws:identity::03393893651506583485:root`<br><br>Empty for anonymous requests. |
| TIME | Time | Total processing time for the request in microseconds. |
| TLIP | Trusted Load Balancer IP Address | If the request was routed by a trusted Layer 7 load balancer, the IP address of the load balancer. |
| UUID | Universally Unique Identifier | The identifier of the object within the StorageGRID system. |
| VSID | Version ID | The version ID of the specific version of an object whose metadata was updated. Operations on buckets and objects in unversioned buckets don’t include this field. |

**SVRF: Object Store Verify Fail**

This message is issued whenever a content block fails the verification process. Each time replicated object data is read from or written to disk, several verification and integrity checks are performed to ensure the data sent to the requesting user is identical to the data originally ingested into the system. If any of these checks fail, the system automatically quarantines the corrupt replicated object data to prevent it from being retrieved again.

| Code | Field | Description |
|---|---|---|
| CBID | Content Block Identifier | The unique identifier of the content block which failed verification. |

| Code | Field | Description |
|---|---|---|
| RSLT | Result Code | Verification failure type:<br><br>CRCF: Cyclic redundancy check (CRC) failed.<br><br>HMAC: Hash-based message authentication code (HMAC) check failed.<br><br>EHSH: Unexpected encrypted content hash.<br><br>PHSH: Unexpected original content hash.<br><br>SEQC: Incorrect data sequence on disk.<br><br>PERR: Invalid structure of disk file.<br><br>DERR: Disk error.<br><br>FNAM: Bad file name. |

This message should be monitored closely. Content verification failures can indicate impending hardware failures.

To determine what operation triggered the message, see the value of the AMID (Module ID) field. For example, an SVFY value indicates that the message was generated by the Storage Verifier module, that is, background verification, and STOR indicates that the message was triggered by content retrieval.

**SVRU: Object Store Verify Unknown**

The LDR service's Storage component continuously scans all copies of replicated object data in the object store. This message is issued when an unknown or unexpected copy of replicated object data is detected in the object store and moved to the quarantine directory.

| Code | Field | Description |
|---|---|---|
| FPTH | File Path | The file path of the unexpected object copy. |
| RSLT | Result | This field has the value 'NONE'. RSLT is a mandatory message field, but is not relevant for this message. 'NONE' is used rather than 'SUCS' so that this message is not filtered. |

The SVRU: Object Store Verify Unknown audit message should be monitored closely. It means unexpected copies of object data were detected in the object store. This situation should be investigated immediately to determine how theses copies were created, because it can indicate impending hardware failures.

**SYSD: Node Stop**

When a service is stopped gracefully, this message is generated to indicate the shutdown

was requested. Typically this message is sent only after a subsequent restart, because the audit message queue is not cleared before shutdown. Look for the SYST message, sent at the beginning of the shutdown sequence, if the service has not restarted.

| Code | Field | Description |
|---|---|---|
| RSLT | Clean Shutdown | The nature of the shutdown:<br><br>SUCS: System was cleanly shutdown. |

The message does not indicate if the host server is being stopped, only the reporting service. The RSLT of a SYSD can’t indicate a "dirty" shutdown, because the message is generated only by "clean" shutdowns.

#### SYST: Node Stopping

When a service is gracefully stopped, this message is generated to indicate the shutdown was requested and that the service has initiated its shutdown sequence. SYST can be used to determine if the shutdown was requested, before the service is restarted (unlike SYSD, which is typically sent after the service restarts.)

| Code | Field | Description |
|---|---|---|
| RSLT | Clean Shutdown | The nature of the shutdown:<br><br>SUCS: System was cleanly shutdown. |

The message does not indicate if the host server is being stopped, only the reporting service. The RSLT code of a SYST message can’t indicate a "dirty" shutdown, because the message is generated only by "clean" shutdowns.

#### SYSU: Node Start

When a service is restarted, this message is generated to indicate if the previous shutdown was clean (commanded) or disorderly (unexpected).

| Code | Field | Description |
|---|---|---|
| RSLT | Clean Shutdown | The nature of the shutdown:<br><br>SUCS: System was cleanly shut down.<br><br>DSDN: System was not cleanly shut down.<br><br>VRGN: System was started for the first time after server installation (or re-installation). |

The message does not indicate if the host server was started, only the reporting service. This message can be used to:

- Detect discontinuity in the audit trail.

- Determine if a service is failing during operation (as the distributed nature of the StorageGRID system can mask these failures). Server Manager restarts a failed service automatically.

## Copyright information

Copyright © 2026 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.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## 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.