Format der Revisionsprotokolldatei

StorageGRID software

NetApp
July 23, 2026

Inhalt

Format der Revisionsprotokolldatei

Erfahren Sie mehr über das Format von Revisionsprotokolldateien in StorageGRID

Die Revisionsprotokoll Logdateien befinden sich auf jedem Admin Node und enthalten eine Sammlung einzelner Audit-Meldungen.

${post_edited_translations.segment}

- ${post_edited_translations.segment}

  YYYY-MM-DDTHH:MM:SS.UUUUUU, wobei *UUUUUU* Mikrosekunden sind.

- Die Prüfmeldung selbst, in eckigen Klammern eingeschlossen und beginnend mit `AUDT`.

Das folgende Beispiel zeigt drei Audit-Meldungen in einer Revisionsprotokolldatei (Zeilenumbrüche wurden zur besseren Lesbarkeit hinzugefügt). Diese Meldungen wurden generiert, als ein Mandant einen S3-Bucket erstellte und diesem zwei Objekte hinzufügte.

```
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::175300642415970547
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::175300642415970547
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::175300642415970547
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]]
```

Im Standardformat sind die Audit-Meldungen in den Revisionsprotokolldateien nicht leicht lesbar oder interpretierbar. Mit "audit-explain Tool" lassen sich vereinfachte Zusammenfassungen der Audit-Meldungen im Revisionsprotokoll erhalten. Mit "${post_edited_translations.segment}" kann zusammengefasst werden, wie viele Schreib-, Lese- und Löschvorgänge protokolliert wurden und wie lange diese Vorgänge dauerten.

Verwenden Sie das Tool „audit-explain“, um StorageGRID Revisionsprotokollmeldungen zu übersetzen.

Das `audit-explain`Tool kann verwendet werden, um die Audit-Meldungen im Revisionsprotokoll in ein leicht lesbares Format zu übersetzen.

Bevor Sie beginnen

- Du hast "spezifische Zugriffsberechtigungen".
- Sie müssen über die Datei `Passwords.txt` verfügen.
- Sie müssen die IP-Adresse des primären Admin-Node kennen.

Über diese Aufgabe

Das `audit-explain`Tool, das auf dem primären Admin-Node verfügbar ist, bietet vereinfachte Zusammenfassungen der Audit-Meldungen in einem Revisionsprotokoll.

> Das `audit-explain` Tool ist primär für den Einsatz durch den technischen Support bei der Fehlerbehebung vorgesehen. Die Verarbeitung von `audit-explain` Abfragen kann viel CPU-Leistung beanspruchen, was StorageGRID Vorgänge beeinträchtigen kann.

Dieses Beispiel zeigt die typische Ausgabe des `audit-explain` Tools. Diese vier "${post_edited_translations.segment}" Revisionsprotokollmeldungen wurden generiert, als der S3-Tenant mit der Konto-ID 92484777680322627870 S3 PUT-Anfragen verwendete, um einen Bucket mit dem Namen „bucket1“ zu erstellen und diesem Bucket drei Objekte hinzuzufügen.

```
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
```

Das `audit-explain`Tool kann Folgendes ausführen:

- Einfache oder komprimierte Revisionsprotokolle werden verarbeitet. Beispielsweise:

  ```
  audit-explain audit.log

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

- Mehrere Dateien gleichzeitig verarbeiten. Zum Beispiel:

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

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

- Eingaben aus einer Pipe akzeptieren, was das Filtern und Vorverarbeiten der Eingabe mithilfe des `grep` Befehls oder anderer Methoden ermöglicht. Beispielsweise:

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

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

Da Revisionsprotokolle sehr groß sein und langsam analysiert werden können, lässt sich Zeit sparen, indem Sie die Teile herausfiltern, die Sie sich ansehen möchten, und `audit-explain` auf diese Teile anwenden, anstatt auf die gesamte Datei.

> Das `audit-explain` Tool akzeptiert keine komprimierten Dateien als Eingabe über eine Pipe. Um komprimierte Dateien zu verarbeiten, können deren Dateinamen als Befehlszeilenargumente angegeben oder das `zcat` Tool verwendet werden, um die Dateien zuerst zu dekomprimieren. Beispielsweise:
>
> ```
> zcat audit.log.gz | audit-explain
> ```

Die `help (-h)` Option zeigt die verfügbaren Optionen an. Zum Beispiel:

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

Schritte

1. ${post_edited_translations.segment}
   a. Geben Sie den folgenden Befehl ein: `ssh admin@`*`primary_Admin_Node_IP`*
   b. Geben Sie das in der `Passwords.txt` Datei aufgeführte Passwort ein.
   c. Geben Sie den folgenden Befehl ein, um zu root zu wechseln: `su -`
   d. Geben Sie das in der `Passwords.txt` Datei aufgeführte Passwort ein.

      Wenn Sie als Root angemeldet sind, ändert sich die Eingabeaufforderung von `$` zu `#`.

2. Geben Sie den folgenden Befehl ein, wobei `/var/local/audit/export/audit.log` den Namen und den Speicherort der Datei oder Dateien bezeichnet, die Sie analysieren möchten:

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

   Das `audit-explain` Tool gibt für Menschen lesbare Interpretationen aller Meldungen in der/den angegebenen Datei(en) aus.

   > Um die Zeilenlänge zu reduzieren und die Lesbarkeit zu verbessern, werden Zeitstempel standardmäßig nicht angezeigt. Wenn die Zeitstempel angezeigt werden sollen, kann die Option (`-t`timestamp verwendet werden.

Verwenden Sie das Tool audit-sum, um StorageGRID-Revisionsprotokollmeldungen zusammenzufassen.

Das `audit-sum` Tool kann verwendet werden, um die Anzahl der Revisionsprotokoll-Meldungen für Schreiben, Lesen, Head und Löschen zu zählen sowie die minimale, maximale und durchschnittliche Zeit (oder Größe) für jeden Operationstyp anzuzeigen.

Bevor Sie beginnen

- Du hast "spezifische Zugriffsberechtigungen".
- Sie haben die `Passwords.txt` Datei.
- Sie kennen die IP-Adresse des primären Admin-Knotens.

Über diese Aufgabe

Das `audit-sum` Tool, das auf dem primären Admin-Knoten verfügbar ist, fasst zusammen, wie viele Schreib-, Lese- und Löschvorgänge protokolliert wurden und wie lange diese Vorgänge gedauert haben.

> Das `audit-sum` Tool ist primär für den Einsatz durch den technischen Support bei der Fehlerbehebung vorgesehen. Die Verarbeitung `audit-sum` von Abfragen kann viel CPU-Leistung beanspruchen, was StorageGRID Vorgänge beeinträchtigen kann.

Dieses Beispiel zeigt die typische Ausgabe des `audit-sum` Tools. Dieses Beispiel zeigt, wie lange Protokolloperationen dauerten.

```
  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
```

Das `audit-sum`Tool liefert Anzahl und Zeiten für die folgenden S3- und ILM-Audit-Meldungen in einem Revisionsprotokoll.

> Prüfcodes werden aus dem Produkt und der Dokumentation entfernt, sobald Funktionen veraltet sind. Falls ein Prüfcode auftritt, der hier nicht aufgeführt ist, bieten die vorherigen Versionen dieses Themas Informationen zu älteren StorageGRID Releases. Zum Beispiel "StorageGRID 11.8 Verwendung des audit sum Tools".

Code	Beschreibung	Siehe
IDEL	ILM Initiated Delete: Protokolliert, wenn ILM den Prozess zum Löschen eines Objekts startet.	"${post_edited_translations.segment}"
SDEL	S3 DELETE: Protokolliert eine erfolgreiche Transaktion zum Löschen eines Objekts oder Buckets.	"SDEL: S3-DELETE"

Code	Beschreibung	Siehe
SGET	S3 GET: Protokolliert eine erfolgreiche Transaktion zum Abrufen eines Objekts oder zum Auflisten der Objekte in einem Bucket.	"SGET: S3 GET"
SHEA	S3 HEAD: Protokolliert eine erfolgreiche Transaktion, um die Existenz eines Objekts oder Buckets zu überprüfen.	"SHEA: S3 HEAD"
${post_edited_translations.segment}	S3 PUT: Protokolliert eine erfolgreiche Transaktion zum Erstellen eines neuen Objekts oder Buckets.	"SPUT: S3 PUT"

Das `audit-sum`Tool kann Folgendes ausführen:

- Einfache oder komprimierte Revisionsprotokolle werden verarbeitet. Beispielsweise:

```
audit-sum audit.log

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

- Mehrere Dateien gleichzeitig verarbeiten. Zum Beispiel:

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

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

- Eingaben aus einer Pipe akzeptieren, was das Filtern und Vorverarbeiten der Eingabe mithilfe des `grep` Befehls oder anderer Methoden ermöglicht. Beispielsweise:

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

grep bucket1 audit.log | audit-sum

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

> Dieses Tool akzeptiert keine komprimierten Dateien als Eingabe über eine Pipe. Um komprimierte Dateien zu verarbeiten, können deren Dateinamen als Befehlszeilenargumente angegeben oder das `zcat` Tool verwendet werden, um die Dateien zuerst zu dekomprimieren. Beispiel:
>
> ```
> audit-sum audit.log.gz
>
> zcat audit.log.gz | audit-sum
> ```

Mithilfe von Befehlszeilenoptionen können Sie Operationen an Buckets getrennt von Operationen an Objekten zusammenfassen oder Nachrichtenübersichten nach Bucket-Name, Zeitraum oder Zieltyp gruppieren. Standardmäßig werden die minimale, maximale und durchschnittliche Operationszeit angezeigt, aber mit der `size (-s)` Option kann stattdessen die Objektgröße betrachtet werden.

Die `help (-h)` Option zeigt die verfügbaren Optionen an. Zum Beispiel:

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

Schritte

1. ${post_edited_translations.segment}

   a. Geben Sie den folgenden Befehl ein: `ssh admin@primary_Admin_Node_IP`

   b. Geben Sie das in der `Passwords.txt` Datei aufgeführte Passwort ein.

   c. Geben Sie den folgenden Befehl ein, um zu root zu wechseln: `su -`

   d. Geben Sie das in der `Passwords.txt` Datei aufgeführte Passwort ein.

   Wenn Sie als Root angemeldet sind, ändert sich die Eingabeaufforderung von `$` zu `#`.

2. Wenn Sie alle Nachrichten im Zusammenhang mit Schreib-, Lese-, Head- und Löschvorgängen analysieren möchten, gehen Sie wie folgt vor:

   a. Geben Sie den folgenden Befehl ein, wobei `/var/local/audit/export/audit.log` den Namen und den Speicherort der Datei oder Dateien bezeichnet, die Sie analysieren möchten:

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

   Dieses Beispiel zeigt die typische Ausgabe des `audit-sum` Tools. Dieses Beispiel zeigt, wie lange Protokolloperationen dauerten.

   ```
     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 diesem Beispiel sind SGET (S3 GET)-Operationen mit durchschnittlich 1,13 Sekunden am langsamsten, aber sowohl SGET- als auch SPUT (S3 PUT)-Operationen zeigen lange Worst-Case-Zeiten von etwa 1.770 Sekunden.

   b. Um die 10 langsamsten Abrufvorgänge anzuzeigen, kann der Befehl grep verwendet werden, um nur SGET-Nachrichten auszuwählen, und die Option long output (`-l` kann hinzugefügt werden, um Objektpfade einzuschließen:

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

   Die Ergebnisse umfassen den Typ (Objekt oder Bucket) und den Pfad, was es ermöglicht, das Revisionsprotokoll nach anderen Meldungen zu durchsuchen, die sich auf diese bestimmten Objekte

beziehen.

```
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
```

+ Anhand dieser Beispielausgabe lässt sich erkennen, dass die drei langsamsten S3 GET-Anfragen Objekte mit einer Größe von jeweils etwa 5 GB betrafen, die deutlich größer sind als die übrigen Objekte. Die große Größe erklärt die langen Abrufzeiten im ungünstigsten Fall.

3. Wenn Sie ermitteln möchten, welche Größen von Objekten in Ihr Grid eingelesen und daraus abgerufen werden, verwenden Sie die Option Größe (`-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 diesem Beispiel liegt die durchschnittliche Objektgröße für SPUT unter 2,5 MB, aber die durchschnittliche Größe für SGET ist deutlich größer. Die Anzahl der SPUT-Nachrichten ist wesentlich höher als die Anzahl der SGET-Nachrichten, was darauf hindeutet, dass die meisten Objekte nie abgerufen werden.

4. Wenn festgestellt werden soll, ob die Abrufe gestern langsam waren:

   a. Den Befehl im entsprechenden Revisionsprotokoll mit der Option group-by-time (`-gt` ausführen, gefolgt vom Zeitraum (z. B. 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
```

Diese Ergebnisse zeigen, dass der S3 GET-Traffic zwischen 06:00 und 07:00 sprunghaft anstieg. Sowohl die maximale als auch die durchschnittliche Zeit waren in diesem Zeitraum deutlich höher und stiegen nicht allmählich mit zunehmender Anzahl an. Diese Kennzahlen deuten darauf hin, dass die Kapazität überschritten wurde, möglicherweise im Netzwerk oder in der Fähigkeit des Grids, Anfragen zu verarbeiten.

b. Um zu ermitteln, welche Größe die gestern stündlich abgerufenen Objekte hatten, die Option (`-s`) zum Befehl hinzufügen:

```
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
```

Diese Ergebnisse deuten darauf hin, dass einige sehr große Abrufe stattfanden, als das gesamte Abrufaufkommen seinen Höhepunkt erreichte.

c. Um weitere Details anzuzeigen, kann die Funktion "audit-explain Tool" verwendet werden, um alle SGET Operationen während dieser Stunde zu überprüfen:

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

Falls die Ausgabe des grep-Befehls voraussichtlich aus vielen Zeilen besteht, kann der `less` Befehl hinzugefügt werden, um den Inhalt der Revisionsprotokoll-Datei seitenweise (bildschirmweise) anzuzeigen.

5. Wenn Sie feststellen möchten, ob SPUT-Operationen auf Buckets langsamer sind als SPUT-Operationen auf Objekten:

a. Beginnen Sie mit der Option `-go`, die Meldungen für Objekt- und Bucket-Operationen separat gruppiert:

```
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
```

Die Ergebnisse zeigen, dass SPUT-Operationen für Buckets andere Leistungsmerkmale aufweisen als SPUT-Operationen für Objekte.

b. Um festzustellen, welche Buckets die langsamsten SPUT-Operationen aufweisen, kann die `-gb` Option verwendet werden, die Nachrichten nach Bucket gruppiert:

```
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. Um zu ermitteln, welche Buckets die größte SPUT-Objektgröße aufweisen, sollten sowohl die `-gb` als auch die `-s` Optionen verwendet werden:

```
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
```