稽核日誌檔格式
StorageGRID software

NetApp
July 23, 2026

目錄

稽核日誌檔格式

了解 StorageGRID 中稽核記錄檔的格式

${post_edited_translations.segment}

每個稽核訊息都包含以下內容：

- 以 ISO 8601 格式表示觸發稽核訊息 (ATIM) 的事件的協調世界時 (UTC)，後面接著一個空格：

  YYYY-MM-DDTHH:MM:SS.UUUUUU，其中 *UUUUUU* 單位為微秒。

- 稽核訊息本身括在方括號內，並以 `AUDT`開頭。

以下範例顯示稽核日誌檔中的三個稽核訊息（為了便於閱讀而新增了換行符號）。這些訊息是在租戶建立 S3 儲存貯體並向該儲存貯體新增兩個物件時產生的。

```
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):84396067221084560 22]]

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]]
```

在預設格式中，稽核日誌檔中的稽核訊息並不易於讀取或解讀。您可以使用 "audit-explain 工具" 來取得稽核日誌中稽核訊息的簡化摘要。您可以使用 "audit-sum 工具" 來摘要記錄了多少次寫入、讀取和刪除作業，以及這些作業花費了多少時間。

使用 audit-explain 工具翻譯 StorageGRID 稽核訊息

您可以使用 `audit-explain`工具，將稽核日誌中的稽核訊息轉換為易於閱讀的格式。

開始之前

- 您有"特定存取權限"。
- 您必須擁有該 `Passwords.txt`檔案。
- 您必須知道主要管理節點的 IP 位址。

關於此任務

此 `audit-explain`工具可在主要管理節點上使用，提供稽核日誌中稽核訊息的簡化摘要。

> 該 `audit-explain`工具主要供技術支援在疑難排解作業中使用。處理 `audit-explain`查詢可能會使用大量 CPU 資源，這可能會影響 StorageGRID 的運作。

此範例顯示 `audit-explain` 工具的典型輸出。這四個 "SPUT" 稽核訊息是在帳戶 ID 為 92484777680322627870 的 S3 租戶使用 S3 PUT 要求建立名為 "bucket1" 的儲存貯體，並向該儲存貯體新增三個物件時產生的。

```
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
```

此 `audit-explain` 工具可以執行下列操作：

- 處理純文字或壓縮的稽核日誌。例如：

  ```
  audit-explain audit.log

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

- 同時處理多個檔案。例如：

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

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

- 接受來自管道的輸入，這可讓您使用 `grep` 命令或其他方式來篩選和預先處理輸入。例如：

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

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

由於稽核日誌可能非常大且剖析速度很慢，您可以藉由篩選想要查看的部分，並對這些部分（而非整個檔案）執行 `audit-explain` 來節省時間。

此 `audit-explain`工具不接受壓縮檔案作為管道輸入。若要處理壓縮檔案，請提供其檔案名稱作為命令列引數，或使用 `zcat`工具先解壓縮檔案。例如：

```
zcat audit.log.gz | audit-explain
```

使用 `help (-h)` 選項來查看可用的選項。例如：

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

步驟

1. ${post_edited_translations.segment}
   a. 輸入以下命令： `ssh admin@primary_Admin_Node_IP`
   b. 請輸入 `Passwords.txt`檔案中列出的密碼。
   c. 輸入以下命令以切換至根目錄： `su -`
   d. 請輸入 `Passwords.txt`檔案中列出的密碼。

      當您以根目錄身分登入時，提示符號會從 `$`變為 `#`。

2. 輸入下列命令，其中 `/var/local/audit/export/audit.log` 代表您要分析之檔案的名稱與位置：

   ```
   $ audit-explain /var/local/audit/export/audit.log
   ```
   neoplasms_of_the_central_nervous_system.json (1).zip (1.1 KB)

I am trying to load this JSON file into a pandas dataframe. The file contains a list of dictionaries, where each dictionary represents a neoplasm of the central nervous system.

I have tried the following code:

However, I get the following error:

I have checked the JSON file and it seems to be valid. What am I doing wrong?

The JSON file is attached.

The file you attached is not a valid JSON file. It contains multiple JSON objects, one after another, but they are not enclosed in a list (i.e., there are no and at the beginning and end of the file, and the objects are not separated by commas). This format is often called "JSON Lines" or "newline-delimited JSON" (ndjson).

Pandas has a built-in way to read this format directly using with the argument.

Here is how you can load it:

If you want to read it using the module first (though is preferred and faster), you would have to read it line by line:

+ 此 `audit-explain` 工具會列印指定檔案中所有訊息的人類可讀解讀。

+

為了縮短行長並提高可讀性，預設不顯示時間戳記。如果您想要查看時間戳記，請使用 timestamp (`-t`) 選項。

使用 audit-sum 工具彙總 StorageGRID 稽核訊息

您可以使用 `audit-sum`工具來統計寫入、讀取、head 及刪除稽核訊息的數量，並查看每種操作類型的最小值、最大值和平均時間（或大小）。

開始之前

- 您有"特定存取權限"。
- 您擁有 `Passwords.txt` 檔案。
- ${post_edited_translations.segment}

關於此任務

主要管理節點上提供的 `audit-sum` 工具可摘要說明已記錄的寫入、讀取和刪除作業數量，以及這些作業所花費的時間。

此 `audit-sum`工具主要供技術支援在進行疑難排解作業時使用。處理 `audit-sum`查詢可能會消耗大量的 CPU 運算能力，這可能會影響 StorageGRID 作業。

此範例顯示 `audit-sum` 工具的典型輸出。此範例顯示協定作業所花費的時間。

```
  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
```

此 `audit-sum` 工具提供稽核日誌中下列 S3 和 ILM 稽核訊息的計數與時間。

稽核代碼會隨著功能被取代而從產品和說明文件中移除。如果您遇到此處未列出的稽核代碼，請檢查此主題的先前版本以取得較舊的 StorageGRID 版本。例如，"${post_edited_translations.segment}"。

程式碼	說明	請參閱
IDEL	${post_edited_translations.segment}	"IDEL：ILM 發起的刪除"
SDEL	S3 DELETE：記錄成功刪除物件或儲存貯體的交易。	"SDEL：S3 DELETE"
SGET	S3 GET：記錄成功擷取物件或列出儲存貯體中物件的交易。	"SGET：S3 GET"
SHEA	S3 HEAD：記錄成功的交易，以檢查物件或儲存貯體是否存在。	"SHEA：S3 HEAD"
SPUT	S3 PUT：記錄建立新物件或儲存桶的成功交易。	"SPUT：S3 PUT"

此 `audit-sum`工具可以執行下列操作：

- 處理純文字或壓縮的稽核日誌。例如：

  ```
  audit-sum audit.log

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

- 同時處理多個檔案。例如：

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

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

- 接受來自管道的輸入，這可讓您使用 `grep` 命令或其他方式來篩選和預先處理輸入。例如：

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

  grep bucket1 audit.log | audit-sum
  ```
  neoplasms_of_the_central_nervous_system.json (1).zip (1.1 KB)

I am trying to load this JSON file into a pandas dataframe. The file contains a list of dictionaries, where each dictionary represents a neoplasm of the central nervous system.

I have tried the following code:

However, I get the following error:

I have checked the JSON file and it seems to be valid. What am I doing wrong?

The JSON file is attached.

The file you attached is not a valid JSON file. It contains multiple JSON objects, one after another, but they are not enclosed in a list (i.e., there are no and at the beginning and end of the file, and the objects are not separated by commas). This format is often called "JSON Lines" or "newline-delimited JSON" (ndjson).

Pandas has a built-in way to read this format directly using with the argument.

Here is how you can load it:

If you want to read it using the module first (though is preferred and faster), you would have to read it line by line:

+ `grep SPUT audit.log | grep bucket1 | audit-sum`
neoplasms_of_the_central_nervous_system.json (1).zip (1.1 KB)

I am trying to load this JSON file into a pandas dataframe. The file contains a list of dictionaries, where each dictionary represents a neoplasm of the central nervous system.

I have tried the following code:

However, I get the following error:

I have checked the JSON file and it seems to be valid. What am I doing wrong?

The JSON file is attached.

The file you attached is not a valid JSON file. It contains multiple JSON objects, one after another, but they are not enclosed in a list (i.e., there are no and at the beginning and end of the file, and the objects are not separated by commas). This format is often called "JSON Lines" or "newline-delimited JSON" (ndjson).

Pandas has a built-in way to read this format directly using with the argument.

Here is how you can load it:

If you want to read it using the module first (though is preferred and faster), you would have to read it line by line:

此工具不接受壓縮檔案作為管道輸入。若要處理壓縮檔案，請提供其檔案名稱作為命令列引數，或使用 `zcat` 工具先解壓縮檔案。例如：

```
audit-sum audit.log.gz
```

`zcat audit.log.gz | audit-sum` neoplasms_of_the_central_nervous_system.json (1).zip (1.1 KB)

I am trying to load this JSON file into a pandas dataframe. The file contains a list of dictionaries, where each dictionary represents a neoplasm of the central nervous system.

I have tried the following code:

However, I get the following error:

I have checked the JSON file and it seems to be valid. What am I doing wrong?

The JSON file is attached.

The file you attached is not a valid JSON file. It contains multiple JSON objects, one after another, but they are not enclosed in a list (i.e., there are no and at the beginning and end of the file, and the objects are not separated by commas). This format is often called "JSON Lines" or "newline-delimited JSON" (ndjson).

Pandas has a built-in way to read this format directly using with the argument.

Here is how you can load it:

If you want to read it using the module first (though is preferred and faster), you would have to read it line by line:

您可以使用命令列選項，將貯存庫上的作業與物件上的作業分開進行摘要，或是按貯存庫名稱、時間週期或目標類型來將訊息摘要分組。預設情況下，摘要會顯示最小、最大和平均作業時間，但您可以改用 `size (-s)` 選項來查看物件大小。

使用 `help (-h)` 選項來查看可用的選項。例如：

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

步驟

1. ${post_edited_translations.segment}
   a. 輸入以下命令： `ssh admin@primary_Admin_Node_IP`
   b. 請輸入 `Passwords.txt`檔案中列出的密碼。
   c. 輸入以下命令以切換至根目錄： `su -`
   d. 請輸入 `Passwords.txt`檔案中列出的密碼。

      當您以根目錄身分登入時，提示符號會從 `$`變為 `#`。
2. ${post_edited_translations.segment}
   a. 輸入下列命令，其中 `/var/local/audit/export/audit.log` 代表您要分析之檔案的名稱與位置：

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

此範例顯示 `audit-sum` 工具的典型輸出。此範例顯示協定作業所花費的時間。

```
  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
```

${post_edited_translations.segment}

b. 若要顯示速度最慢的 10 個擷取操作，請使用 grep 命令僅選擇 SGET 訊息，並新增 long 輸出選項(`-l`以包含物件路徑：

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

結果包括類型（物件或儲存桶）和路徑，可讓您在稽核日誌中搜尋與這些特定物件相關的其他訊息。

```
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
```

+ 從此範例輸出中，您可以看到三個最慢的 S3 GET 要求是針對大小約為 5 GB 的物件，這比其他物件大得多。較大的大小是導致最差情況下擷取時間緩慢的原因。

3. 如果您想確定擷取至網格及從網格擷取的物件大小，請使用 size 選項(`-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
```

在此範例中、SPUT 的平均物件大小小於 2.5 MB、但 SGET 的平均大小則大得多。SPUT 訊息的數量遠高於 SGET 訊息的數量、這表示大多數物件從未被擷取。

4. 如果您想確定昨天的檢索速度是否緩慢：

   a. 在相應的稽核日誌上發出命令，並使用 group-by-time 選項(`-gt`），後接時間段（例如，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
```

結果顯示，S3 GET 請求流量在 06:00 至 07:00 之間出現峰值。在此期間，最大請求時間和平均請求時間均顯著升高，且並非隨著請求數量的增加而逐漸增加。這些計量表明，容量可能已超出負荷，可能是網路方面，或是網格處理請求的能力方面。

b. 若要確定昨天每小時擷取的物件大小，請在命令中加入大小選項 (`-s`) ：

```
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
```

${post_edited_translations.segment}

c. 若要查看更多詳細資訊，請使用 "audit-explain 工具" 審查該小時內的所有 SGET 作業：

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

如果 grep 命令的輸出預期會有多行，請加入 `less` 命令，以一次一頁（一個畫面）的方式顯示稽核日誌檔案的內容。

5. ${post_edited_translations.segment}

a. 請先使用 `-go` 選項，此選項會分別將物件和儲存貯體作業的訊息進行分組：

```
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
```

${post_edited_translations.segment}

b. 若要確定哪些儲存桶的 SPUT 操作速度最慢，請使用 `-gb`選項，該選項可依儲存桶對訊息進行分組：

```
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. 若要確定哪些貯存庫具有最大的 SPUT 物件大小，請同時使用 `-gb` 和 `-s` 選項：

```
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
```