## CN REST - Story #8756

## Ensure replica auditor is effective

2019-01-12 20:25 - Chris Jones

| | | | |
|---|---|---|---|
| **Status:** | New | **Start date:** | 2018-05-01 |
| **Priority:** | Normal | **Due date:** | |
| **Assignee:** | Chris Jones | **% Done:** | 0% |
| **Category:** | d1_replication_auditor | **Estimated time:** | 0.00 hour |
| **Target version:** | | | |
| **Story Points:** | | | |

**Description**

The replication auditor service is currently configured to audit all objects every 90 days. As documented in #8582, the auditor is not working correctly. While the errors being thrown that are described in that ticket seem to be limited to pids with certain characters in them, I think the whole auditor process is not keeping up with our content.

Looking at the number of objects on each member node that haven't been audited in the last 90 days, auditing is well behind (if we consider it working at all):

```
SELECT sm.authoritive_member_node, count(smr.guid) AS count
    FROM systemmetadata sm INNER JOIN smreplicationstatus smr
    ON sm.guid = smr.guid
    WHERE
        smr.member_node != 'urn:node:CN' AND
        sm.date_uploaded < (SELECT CURRENT_DATE - interval '90 days') AND
        smr.date_verified < (SELECT CURRENT_DATE - interval '90 days')
    GROUP BY sm.authoritive_member_node
    ORDER BY count DESC;
```

```
 authoritive_member_node | count
-------------------------+--------
 urn:node:ARCTIC         | 771872
 urn:node:PANGAEA        | 507456
 urn:node:LTER           | 416339
 urn:node:DRYAD          | 374439
 urn:node:CDL            | 242115
 urn:node:PISCO          | 235791
 urn:node:KNB            |  86075
 urn:node:TDAR           |  75639
 urn:node:NCEI           |  50974
 urn:node:USGS_SDC       |  40290
 urn:node:TERN           |  31671
 urn:node:ESS_DIVE       |  28830
 urn:node:NMEPSCOR       |  16042
 urn:node:GOA            |   9266
 urn:node:IARC           |   7677
 urn:node:NRDC           |   6673
 urn:node:TFRI           |   6478
 urn:node:PPBIO          |   3464
 urn:node:ORNLDAAC       |   3328
 urn:node:FEMC           |   2430
 urn:node:EDI            |   2098
 urn:node:GRIIDC         |   2065
 urn:node:mnTestKNB      |   2010
 urn:node:SANPARKS       |   2008
 urn:node:ONEShare       |   1874
 urn:node:R2R            |   1787
 urn:node:USGSCSAS       |   1151
 urn:node:EDACGSTORE     |   1075
 urn:node:US_MPC         |   1032
 urn:node:RW            |    970
 urn:node:KUBI           |    516
 urn:node:NEON           |    487
 urn:node:LTER_EUROPE    |    343
 urn:node:IOE            |    279
 urn:node:RGD            |    273
 urn:node:ESA            |    272
 urn:node:NKN            |    218
 urn:node:OTS_NDC        |    126
 urn:node:BCODMO         |    115
 urn:node:SEAD           |     90
 urn:node:mnTestNKN      |     50
 urn:node:EDORA          |     28
 urn:node:ONEShare.pem   |     22
 urn:node:CLOEBIRD       |     17
 urn:node:mnTestBCODMO   |     11
 urn:node:USANPN         |     10
 urn:node:mnTestTDAR     |     10
 urn:node:MyMemberNode   |      1
```

The table above represents the number of un-audited objects (in the last 90 days), but I get the feeling that the auditor isn't able to audit any of the content it is charged to audit given 1) the frequency, 2) the number of threads allotted, and 3) the configured batch count (seems way too low). ~~Note that this query excludes replicated content - this is just the original objects~~ (After looking at my query again, I think the join is including all replicas - the total is 2,935,787, which is greater than the total objects in the system (2,751,136), so this query needs to be refined).

We need to evaluate the true effectiveness of the auditor. Some strategies may include: 1) looking to see if we may be in an infinite loop on processing a few pids due to the issues in #8582, 2) seeing if we can increase the batch size by increasing the total threads allocated in the executor, and 3) decide if we need to offload the process from the CNs and distribute the workload across a cluster of workers that can do the auditing faster. Needs some thought and discussion.

**Subtasks:**

| | |
|---|---|
| Story # 8582: Replica Auditing service is throwing errors | **New** |
| Story # 8757: Fix getChecksum() in MNAuditTask to use dynamic checksum algorithms | **New** |
| Task # 8776: Set valid replica status to completed | **New** |
| Task # 8777: Configure CN to audit objects greater than 1GB | **New** |
| Task # 8778: Ensure SystemMetadata replica auditing updates are saved and broadcast | **New** |

## History

### #1 - 2019-01-14 21:30 - Chris Jones

*- Description updated*

### #2 - 2019-01-14 22:59 - Chris Jones

*- Description updated*

### #3 - 2019-01-15 21:00 - Chris Jones

Adding some notes on evaluating audit timing:

```
cjones@cn-ucsb-1$ cat /var/log/dataone/replicate/*splunk* | \
cut -d" " -f2,3 | \
tr " " "T" |
tr "," "." > \
audit-times.txt
```

```
import numpy as np
import pandas as pd
from pandas import Series

# Load audit timestamps and sort them
audit_times = np.loadtxt('/Users/cjones/audit-times.txt', dtype = 'datetime64')
audit_times = np.sort(audit_times)
# And create a pandas.Series object
audit_series = pd.Series(audit_times)
```

```
# Create a pandas.DataFrame with the following columns:
# - time: the original sorted times
# - shifted-time: the time column shifted up one cell
# - time-lag: the difference in the times (t1 minus t0)
# -time-lag-ms: the time-lag objects converted to millis as float64s
series = pd.concat([audit_series, audit_series.shift(-1)], axis = 1)
series.columns = ['time', 'shifted-time']
series['time-lag'] = series['shifted-time'] - series['time']
# series['time-lag-ms'] = series['time-lag'].astype('timedelta64[ms]')

# Show summary stats of the lag times
series['time-lag'].describe()

# Filter out the one 2 1/2 day outlier
series['time-lag'][series['time-lag'] < np.timedelta64(24, 'h')].describe()

# count                     363884
# mean     0 days 00:00:08.526884
# std      0 days 00:02:31.333184
# min             0 days 00:00:00
# 25%      0 days 00:00:00.200000
# 50%      0 days 00:00:00.560000
# 75%      0 days 00:00:01.426000
# max      0 days 01:18:42.808000
# Name: time-lag, dtype: object
```