Run Element storage health checks prior to upgrading storage

HCI

dbag-personal, Dave Bagwell
June 22, 2020

NetApp®

Table of Contents

Run Element storage health checks prior to upgrading storage

You must run health checks prior to upgrading Element storage to ensure all storage nodes in your cluster are ready for the next Element storage upgrade.

What you'll need

- You have updated to the latest management services bundle (2.10.27 or later).

    > ❗ You must upgrade to the latest management services bundle before upgrading your Element software to version 12.

- You are running management node 11.3 or later.
- Your cluster version is running NetApp Element software 11.3 or later.

Health check options

You can run health checks using NetApp Hybrid Cloud Control (HCC) UI, HCC API, or the HealthTools suite:

- Use NetApp Hybrid Cloud Control to run Element storage health checks prior to upgrading storage (Preferred method)
- Use API to run Element storage health checks prior to upgrading storage
- Use HealthTools to run Element storage health checks prior to upgrading storage

You can also find out more about storage health checks that are run by the service:

- Storage health checks made by the service

Use NetApp Hybrid Cloud Control to run Element storage health checks prior to upgrading storage

Using NetApp Hybrid Cloud Control (HCC), you can verify that a storage cluster is ready to be upgraded.

Steps

1. Open a web browser and browse to the IP address of the management node:

    ```
    https://<ManagementNodeIP>
    ```

2. Log in to NetApp Hybrid Cloud Control by providing the storage cluster administrator credentials.

3. Click **Upgrade** near the top right of the interface.

4. On the **Upgrades** page, select the **Storage** tab.

5. Click the health check for the cluster you want to check for upgrade readiness.

6. On the **Storage Health Check** page, click **Run Health Check**.

7. If there are issues, do the following:

   a. Go to the specific KB article listed for each issue or perform the specified remedy.

   b. If a KB is specified, complete the process described in the relevant KB article.

   c. After you have resolved cluster issues, click **Re-Run Health Check**.

After the health check completes without errors, the storage cluster is ready to upgrade. See storage node upgrade instructions to proceed.

Use API to run Element storage health checks prior to upgrading storage

You can use REST API to verify that a storage cluster is ready to be upgraded. The health check verifies that there are no obstacles to upgrading, such as pending nodes, disk space issues, and cluster faults.

Steps

1. Locate the storage cluster ID:

   a. Open the management node REST API UI on the management node:

   ```
   https://[management node IP]/inventory/1
   ```

   b. Click **Authorize** and complete the following:

      i. Enter the cluster user name and password.

      ii. Enter the client ID as `mnode-client` if the value is not already populated.

      iii. Click **Authorize** to begin a session.

      iv. Close the authorization window.

   c. From the REST API UI, click **GET /installations**.

   d. Click **Try it out**.

   e. Click **Execute**.

   f. From the response, copy the installation asset ID (`"id"`).

   g. From the REST API UI, click **GET /installations/{id}**.

   h. Click **Try it out**.

i. Paste the installation asset ID into the **id** field.

j. Click **Execute**.

k. From the response, copy and save the storage cluster ID (`"id"`) of the cluster you intend to check for upgrade readiness.

2. Run health checks on the storage cluster:

   a. Open the storage REST API UI on the management node:

   ```
   https://[management node IP]/storage/1
   ```

   b. Click **Authorize** and complete the following:

      i. Enter the cluster user name and password.

      ii. Enter the client ID as `mnode-client` if the value is not already populated.

      iii. Click **Authorize** to begin a session.

   c. Click **POST /health-checks**.

   d. Click **Try it out**.

   e. Enter the storage cluster ID in the parameter field.

   f. Click **Execute** to run a health check on the specified storage cluster.

   The response should indicate state as `initializing`:

   ```
   {
     "_links": {
       "collection": "https://10.117.149.231/storage/1/health-checks",
       "log": "https://10.117.149.231/storage/1/health-checks/358f073f-896e-4751-ab7b-
   ccbb5f61f9fc/log",
       "self": "https://10.117.149.231/storage/1/health-checks/358f073f-896e-4751-
   ab7b-ccbb5f61f9fc"
     },
     "config": {},
     "dateCompleted": null,
     "dateCreated": "2020-02-21T22:11:15.476937+00:00",
     "healthCheckId": "358f073f-896e-4751-ab7b-ccbb5f61f9fc",
     "state": "initializing",
     "status": null,
     "storageId": "c6d124b2-396a-4417-8a47-df10d647f4ab",
     "taskId": "73f4df64-bda5-42c1-9074-b4e7843dbb77"
   }
   ```

   g. Copy the `healthCheckID` that is part of response.

3. Verify the results of the health checks:

   a. Click **GET /health-checks/{healthCheckId}**.

   b. Click **Try it out**.

   c. Enter the health check ID in the parameter field.

   d. Click **Execute**.

   e. Scroll to the bottom of the response body.

4. If the `message` return indicates that there were problems regarding cluster health, do the following:

   a. Go to the specific KB article listed for each issue or perform the specified remedy.

   b. If a KB is specified, complete the process described in the relevant KB article.

   c. After you have resolved cluster issues, run **GET /health-checks/{healthCheckId}** again.

If all health checks are successful, the return is similar to the following example:

```
"message": "All checks completed successfully.",
"percent": 100,
"timestamp": "2020-03-06T00:03:16.321621Z"
```

Use HealthTools to run Element storage health checks prior to upgrading storage

You can verify that the storage cluster is ready to be upgraded by using the `sfupgradecheck` command. This command verifies information such as pending nodes, disk space, and cluster faults.

If your management node is at a dark site, the upgrade readiness check needs the `metadata.json` file you downloaded during [HealthTools upgrades](#) to run successfully.

About this task

This procedure describes how to address upgrade checks that yield one of the following results:

- Running the `sfupgradecheck` command runs successfully. Your cluster is upgrade ready.

- Checks within the `sfupgradecheck` tool fail with an error message. Your cluster is not upgrade ready and additional steps are required.

- Your upgrade check fails with an error message that HealthTools is out-of-date.

- Your upgrade check fails because your management node is on a dark site.

Steps

1. Run the `sfupgradecheck` command:

```
sfupgradecheck -u <cluster-user-name> MVIP
```

ℹ️ For passwords that contain special characters, add a backslash (\) before each special character. For example, `mypass!@1` should be entered as `mypass\!\@`.

Sample input command with sample output in which no errors appear and you are ready to upgrade:

```
sfupgradecheck -u admin 10.117.78.244
```

```
check_pending_nodes:
Test Description: Verify no pending nodes in cluster
More information:
https://kb.netapp.com/support/s/article/ka11A0000008ltOQAQ/pendingnodes
check_cluster_faults:
Test Description: Report any cluster faults
check_root_disk_space:
Test Description: Verify node root directory has at least 12 GBs of available disk
space
Passed node IDs: 1, 2, 3
More information: https://kb.netapp.com/support/s/article/ka11A0000008ltTQAQ/
SolidFire-Disk-space-error
check_mnode_connectivity:
Test Description: Verify storage nodes can communicate with management node
Passed node IDs: 1, 2, 3
More information:
https://kb.netapp.com/support/s/article/ka11A0000008ltYQAQ/mNodeconnectivity
check_files:
Test Description: Verify options file exists
Passed node IDs: 1, 2, 3
check_cores:
Test Description: Verify no core or dump files exists
Passed node IDs: 1, 2, 3
check_upload_speed:
Test Description: Measure the upload speed between the storage node and the
management node
Node ID: 1 Upload speed: 90063.90 KBs/sec
Node ID: 3 Upload speed: 106511.44 KBs/sec
Node ID: 2 Upload speed: 85038.75 KBs/sec
```

2. If there are errors, additional actions are required. See the following sub-sections for details.

Your cluster is not upgrade ready

If you see an error message related to one of the health checks, follow these steps:

1. Review the `sfupgradecheck` error message.

    Sample response:

    ```
    The following tests failed:
    check_root_disk_space:
    Test Description: Verify node root directory has at least 12 GBs of available disk space
    Severity: ERROR
    Failed node IDs: 2
    Remedy: Remove unneeded files from root drive
    More information: https://kb.netapp.com/support/s/article/ka11A00000008ltTQAQ/SolidFire-
    Disk-space-error
    check_pending_nodes:
    Test Description: Verify no pending nodes in cluster
    More information: https://kb.netapp.com/support/s/article/ka11A00000008ltOQAQ/pendingnodes
    check_cluster_faults:
    Test Description: Report any cluster faults
    check_root_disk_space:
    Test Description: Verify node root directory has at least 12 GBs of available disk space
    Passed node IDs: 1, 3
    More information: https://kb.netapp.com/support/s/article/ka11A00000008ltTQAQ/SolidFire-
    Disk-space-error
    check_mnode_connectivity:
    Test Description: Verify storage nodes can communicate with management node
    Passed node IDs: 1, 2, 3
    More information:
    https://kb.netapp.com/support/s/article/ka11A00000008ltYQAQ/mNodeconnectivity
    check_files:
    Test Description: Verify options file exists
    Passed node IDs: 1, 2, 3
    check_cores:
    Test Description: Verify no core or dump files exists
    Passed node IDs: 1, 2, 3
    check_upload_speed:
    Test Description: Measure the upload speed between the storage node and the management
    node
    Node ID: 1 Upload speed: 86518.82 KBs/sec
    Node ID: 3 Upload speed: 84112.79 KBs/sec
    Node ID: 2 Upload speed: 93498.94 KBs/sec
    ```

In this example, node 1 is low on disk space. You can find more information in the knowledge base
(KB) article listed in the error message.

HealthTools is out of date

If you see an error message indicating that HealthTools is not the latest version, follow these instructions:

1. Review the error message and note that the upgrade check fails.

    Sample response:

    ```
    sfupgradecheck failed: HealthTools is out of date:
    installed version: 2018.02.01.200
    latest version: 2020.03.01.09.
    The latest version of the HealthTools can be downloaded from:
    https://mysupport.netapp.com/NOW/cgi-bin/software/
    Or rerun with the -n option
    ```

2. Follow the instructions described in the response.

Your management node is on a dark site

1. Review the message and note that the upgrade check fails:

    Sample response:

    ```
    sfupgradecheck failed: Unable to verify latest available version of healthtools.
    ```

2. Download a JSON file from the NetApp Support Site on a computer that is not the management node and rename it to `metadata.json`.

3. Run the following command:

    ```
    sfupgradecheck -l --metadata=<path-to-metadata-json>
    ```

4. For details, see additional HealthTools upgrades information for dark sites.

5. Verify that the HealthTools suite is up-to-date by running the following command:

    ```
    sfupgradecheck -u <cluster-user-name> -p <cluster-password> MVIP
    ```

Storage health checks made by the service

Storage health checks make the following checks per cluster.

Check Name	Node/Cluster	Description
check_async_results	Cluster	Verifies that the number of asynchronous results in the database is below a threshold number.
check_cluster_faults	Cluster	Verifies that there are no upgrade blocking cluster faults (as defined in Element source).
check_upload_speed	Node	Measures the upload speed between the storage node and the management node.
connection_speed_check	Node	Verifies that nodes have connectivity to the management node serving upgrade packages and estimates connection speed.
check_cores	Node	Checks for kernel crash dump and core files on the node. The check fails for any crashes in a recent time period (threshold 7 days).
check_root_disk_space	Node	Verifies the root file system has sufficient free space to perform an upgrade.
check_var_log_disk_space	Node	Verifies that `/var/log` free space meets some percentage free threshold. If it does not, the check will rotate and purge older logs in order to fall under threshold. The check fails if it is unsuccessful at creating sufficient free space.
check_pending_nodes	Cluster	Verifies that there are no pending nodes on the cluster.

Find more information

- NetApp HCI Documentation Center
- NetApp HCI Resources Page