# Release Notes

**HGST Active Archive System SA-7000**

**September 2015**

**1ET0035**

**Revision 1.1**

**Long Live Data ™ | www.hgst.com**

# Copyright

**The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law: HGST a Western Digital company PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer or express or implied warranties in certain transactions, therefore, this statement may not apply to you.**

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HGST may make improvements or changes in any products or programs described in this publication at any time.

It is possible that this publication may contain reference to, or information about, HGST products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that HGST intends to announce such HGST products, programming, or services in your country.

Technical information about this product is available by contacting your local HGST representative or on the Internet at: www.hgst.com/support

HGST may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents.



HGST, a Western Digital company
3403 Yerba Buena Road
San Jose, CA 95135
Produced in the United States

Long Live Data™ is a trademark of HGST, Inc. and its affiliates in the United States and/or other countries.

HGST trademarks are authorized for use in countries and jurisdictions in which HGST has the right to use, market and advertise the brands.

Other product names are trademarks or registered trademarks of their respective owners.

One MB is equal to one million bytes, one GB is equal to one billion bytes, one TB equals 1,000GB (one trillion bytes) and one PB equals 1,000TB when referring to storage capacity. Usable capacity will vary from the raw capacity due to object storage methodologies and other factors.

References in this publication to HGST products, programs or services do not imply that HGST intends to make these available in all countries in which HGST operates.

Product information is provided for information purposes only and does not constitute a warranty.

Information is true as of the date of publication and is subject to change. Actual results may vary. This publication is for general guidance only. Photographs may show design models.

## Contents

# 1 Document Summary

**Topics:**



This document provides important information regarding functionality and known limitations of the HGST Active Archive System. For more information and a glossary of terms, see the *HGST Active Archive System Administration Guide*.

## 1.1 Intended Audience

This document is intended for storage administrators, IT infrastructure experts, and solutions architects deploying the Active Archive System.

## 1.2 Introduction

The Active Archive System is a unit that is vertically integrated with object storage software, networking, servers and storage in an industry standard 42U rack.

The Active Archive System is comprised of the following major components, all of which have a number of replaceable units:

- Storage Interconnect
- Controller Nodes
- Storage Nodes
- Storage Interconnect
- Power Distribution Units (PDUs)
- Storage Enclosure Basic Storage Arrays

---

**Note:** In addition to the major components, the system includes the rack, cables, rack panels, hardware, labels, power cords, and sleds.

---

## 1.3 Related Documents

For more information about the Active Archive System, please consult the following documents:

- The *HGST Active Archive System Administration Guide* explains how to use the Active Archive System interfaces for executing system management, monitoring, and analytics tasks.
- The *HGST Active Archive System API Guide* provides a reference for the Active Archive System S3 API.
- The *HGST Active Archive System FRU Replacement Guide* provides procedures for replacing hardware components of the Active Archive System.
- The *HGST Active Archive System Installation Guide* provides instructions for the installation of the Active Archive System in the data center, and its initial bringup.
- The *HGST Active Archive System Release Notes* provide important information about changes, new features, and known limitations.
- The *HGST Active Archive System Site Requirements Document* contains data center requirements for the Active Archive System.
- The *HGST Active Archive System Troubleshooting Guide* provides help for issues you might encounter.
- The *HGST Active Archive System Upgrade Guide* provides instructions for software and firmware updates, and system expansion.

For the latest or online version of any of these documents, visit http://www.hgst.com/support.

# 2 Version 1

**Topics:**



Release Date: June 2015

## 2.1 Summary

### 2.1.1 Highlights

- Raw capacity: 4.7PB per rack
- Client performance: 3.5GB/s per system
- Data durability of 15 9s
- S3 protocol support
- Single geo clusters
- SNMP support
- Telemetry collection support
- "Phone Home" policy support

### 2.1.2 Scalability

The Active Archive System has been tested in two-rack clusters.

### 2.1.3 Compatibility

The Active Archive System web interface is compatible with the following tools and web browsers:

- Adobe Flash Player 13.0.0.214 or lower
- Google Chrome (all versions except 34.0.1847.116-1)
- Microsoft Internet Explorer
- Mozilla Firefox

### 2.1.4 Updates

Updating the firmware on the SAS card in a Storage Node causes the Storage Enclosure Basic to go offline. If you are updating the firmware on the Storage Nodes, do the updates one node at a time, and wait for the disk safety to return to 5 before proceeding to the next node. For more information, see the *HGST Active Archive System Upgrade Guide*.

### 2.1.5 General Capabilities

| Component | Limitation |
|---|---|
| Maximum size of the SSDs used for MetaStores | 240 GB |
| Maximum superblock size | 256 MiB |
| Maximum number of 10 GbE uplinks per Controller Node (links to the customer application) | 2 |
| Maximum number of objects per name space | This is limited by the size of the SSD that stores the metadata for that name space. Each metadata entry per superblock requires at least 200 bytes. The actual size per metadata entry depends upon the following: |

| Component | Limitation |
|---|---|
| | • The object name length<br>• The size of your storage pool<br>• The custom metadata stored per object.<br>10% of the free space on an SSD is reserved to prevent performance degradation as a result of the write amplification process. |
| Maximum number of name spaces per Active Archive System | 5,000,000 |
| Maximum object size for S3 (non-multipart) | The Active Archive System supports objects with up to 65,536 superblocks. This results in a maximum object size of 4 TiB. |
| Maximum object size S3 multipart | • Maximum part size: identical to the maximum object size S3 (non-multipart)<br>• Maximum number of parts: 10,000<br>• Maximum object size: 16 TiB **(*)**<br>• Theoretical maximum object size: 160,000 TiB **(*)**<br>• Minimum object size: 0 MiB<br>• Minimum part size: 5 MiB |
| Maximum number of users that can have rights assigned on a name space | 64 |
| Required number of MetaStore nodes on distinct physical machines | 3 |
| Maximum number of files per blockstore | 14 million<br>When using a name space with small file support enabled, the maximum can be increased to 60 million, provided that sufficient MetaStores are provisioned. |
| Maximum size of custom object metadata | 1 MiB |
| Maximum object name length | 255 KiB (limited by the HTTP request size) |
| Maximum number of files per cachecluster when using readcacheclusters | 50 million files per cache cluster |
| Maximum number of parallel streams per client daemon | 100 |
| Maximum recommended number of client daemons per controller | 4 |
| Minimum average file size when using readcacheclusters | With an SSD of 480 GB , the minimum average object size should be larger than 8.4 KiB |

**Note: (*)**

The maximum object size to upload via S3, using multipart, is set to 16 TiB, but in theory it is possible to upload objects of 160,000 TiB when using S3 multipart. However, 160,000 TiB is larger than the capacity supported by a five rack Active Archive System cluster, which is 5 x 6 x 588

x 8TB (60.48PB). This number is calculated as follows:

- 5 racks

- 6 Storage Nodes per rack
- 588 disks per Storage Node
- Disk capacity of 8TB

The recommended way to upload a 16 TiB object is through multipart upload. For example 1,000 parts of 16 GiB or 10,000 parts of 1.6 GiB.

---

#### 2.1.5 Storage Daemon Configuration

To avoid out-of-memory issues caused by an congested repair queue, you can set a maximum amount of memory for the repair queue by setting the `repair_queue_max_mem_size` parameter in the configuration file for the storage daemon. The default value of this parameter is 500,000,000 (500 MB). For more information, see *Tuning the Active Archive System* in the *HGST Active Archive System Administration Guide*. [DSS-1960]

## 2.2 Open Items

### 2.2.1 Cloud Management Center (CMC)

| Issue | Details |
|---|---|
| Swap usage is incorrectly reported when doing multiple uploads of large objects. [DSS-1296] | When performing multiple uploads of large objects, the event `swap usage is over 50%` is triggered. However, the swap usage is actually at 100% at this point.<br>Workaround: Restart the daemons. |
| The CMC displayes partitions with less than 1 GB of used space as having 0 GB used space. [AMIF-920] | On the detail page of a disk, on the monitoring tab, partitions that use less than 1 GB are shown as having 0 GB of used space. |
| There is a confusing message in job log. [AMIF-3209] | In some of the job logs in the CMC, you might spot messages like the following at the end of the job, even though the job is marked as done with no errors:<br>`Failed to parse params {<job_details>}, Error: malformed string`<br>This is actually a coding warning instead of an error and does not influence the job itself or your the performance of the Active Archive System. |
| Events and jobs in the CMC are not using the same time zone set on the machine. [AMIF-3413] | The timezone definitions used are not recent and may be out of sync if your country recently changed time zones. |
| Enabling S3 through the CMC adds an empty domain in the configuration files of the client daemons. [AMIF-3855] | When you enable S3 on any client daemon with the **Enable S3 bucket operations** check box selected but without defining an S3 domain name, the domain parameter in the configuration files of the client daemons remains empty. As a result, no bucket operations can be performed.<br>Workaround: Enable S3 again through the CMC, but do not leave the domain field empty. |
| The CMC is temporarily unavailable once a week. [AMIF-4309] | Due to an old issue with mod-python, Apache is restarted on a weekly basis. During the restart the CMC is unavailable. |

| Issue | Details |
|---|---|
| An exception is generated after deleting an unmanaged disk from the disk detail page. [AMIF-4636] | When deleting a disk with status **Unmanaged**, an exception is generated, even though the operation succeeds. The exception is:<br>`faultCode:Fault faultString:'error' faultDetail:'<Fault 8002: 'disk with guid 5b4a0b42-b8ab-4889-b675-692803c65995 not found.'>'` |
| The **Storage Pool Used Capacity** graph includes decommissioned Storage Nodes. [AMIF-4638] | The CMC dashboard displays a graph of the **Storage Pool Used Capacity** in which the total available capacity shown includes decommissioned Storage Nodes. Decommissioned Storage Nodes should not be included in the display. |
| The **Swap Memory** image is incorrectly scaled (off by 1024). [AMIF-4644] | The scale of the **Swap Memory** image on an individual Controller or Storage Node pane is expressed in MB instead of GB. |
| The CMC displays different logical and physical sizes for `mdX` disks. [AMIF-4759] | For RAID devices, the logical size displayed by the CMC is the usable size, not the physical hardware size of the device. |
| The CMC fails to export decommissioned disk details to PDF. [AMIF-4770] | The version of Adobe Flash Player installed on your computer may be incompatible with the CMC requirement.<br>Workaround: Use Adobe Flash Player 13.0.0.214 or lower in order to export the details of decommissioned disk to a PDF file. |
| Only one member of a MetaStore is marked as **FULL**, instead of all members. [AMIF-5006] | When a MetaStore exceeds the threshold for the number of keys, it is marked as **FULL**. The CMC only marks one member of the MetaStore as **FULL**, whereas all three members of the MetaStore should have the status **FULL**. This has no functional impact. |
| The CMC does not show the System ID and current OS version. [HGST-185] | A user with multiple systems cannot tell which system's CMC they are logged into. |
| The CMC does not respond to **Export details as pdf** for decommissioned drives. [HGST-193] | In the screen showing the list of decommissioned drives, clicking **Export details as PDF** has no response. Nothing happens on the screen or the monitor.<br>Workaround: Use Adobe Flash Player 13.0.0.214 or lower on the machine that you use to log into the CMC. |
| The CMC does not retain port numbers when disabling/enabling client daemons. [HGST-211] | If you disable the client daemon ports, and then enable them again, the CMC does not save the port numbers.<br>Workaround: Write down the port numbers before disabling them. The port numbers are displayed in the CMC under **Dashboard** > **Administration** > **HGST Object Storage Management** > **Interfaces** > **S3** |
| The CMC reports that a reboot the Management Node fails, even though it succeeds. [HGST-601] | The CMC reports that a reboot the Management Node fails, with the following output.<br>`1969-12-31 16:00:01 \| Job status reset to ERROR during workflowengine initialization`<br>Normally, when a node is rebooted through the CMC or API, the Active Archive System waits for the reboot to complete and then tests that certain services are started in order to determine that the node is available again. When rebooting the Management Node, this workflow is the same, except that the workflow is hosted on the Management |

| Issue | Details |
|---|---|
| | Node, so when it reboots, the services stop as well. Therefore, the CMC reports that the reboot failed. |
| The CMC identifies the incorrect slot for failed SSDs on Controller Nodes. [HGST-582] | The image in the *decommissioned disk details* for SSDs is mislabeled: when it highlights slot 9, the decommissioned SSD is actually located in slot 5; when it highlights slot 10, the decommissioned SSD is actually located in slot 6. Disregard the image and refer to the *HGST Active Archive System FRU Replacement Guide* for the correct instructions. |

### 2.2.2 Arakoon

| Issue | Details |
|---|---|
| The *preferred master* remains master after executing `dropMaster`. [ARA-97] | When you have selected a preferred master in an Arakoon cluster and you then execute a `dropMaster`, the selected preferred master remains the master of the cluster. |
| The event message for OBS-ARAKOON-0008 is misleading. [AMIF-4730] | The event message for event OBS-ARAKOON-0008 may indicate a wrong machine name. However, the node name and corresponding MetaStore are correct. |
| Arakoon does not start correctly and without errors. [AMIF-4848] | When Arakoon starts, it verifies that its transaction log is not corrupt. If the transaction log is very large, the verification may time out and Arakoon may start even though the transaction log is corrupt. As a result, Arakoon is not be able to recover upon launch. |

### 2.2.3 Pound

| Issue | Details |
|---|---|
| The `Pound` application may produce unexpected messages in its log. [AMIF-2832] | The following message might be logged multiple times in the `Pound` log file:<br>`localhost pound: NULL get_thr_arg`<br>This is a `Pound`-specific issue, but has no functional impact on the Active Archive System. |
| The `Pound` application is unnecessarily restarted when adding or removing storage LANs. [AMIF-3843] | When adding or removing storage LANs, `Pound` is restarted but its configuration is not updated. Since its configuration has not been updated, the restart is unnecessary. |
| The `Pound` application is listed as **ACTIVE** on the old Management Node after a failover. [AMIF-4416] | After a failover it is possible that on the old Management Node, the `Pound` application is still listed as **ACTIVE**, due to a lag in the update of the `Pound` application in the model. |

### 2.2.4 PUT, GET, and REPAIR Operations

| Issue | Details |
|---|---|
| Saving the `metadatastore` object sometimes fails with a `list index out of range` error. [AMIF-4392] | When a `metadatastore` object is saved, the save operation may fail with the error `list index out of range`. The application server log is then flooded with the following line:<br>`DEBUG:osis.client.xmlrpc:PUT disk` |

| Issue | Details |
|---|---|
| | This may occur during a failover or when extending a MetaStore. The issue is caused when finding disk objects returns an empty list. |
| Name space deletion removes data from the blockstores in a serial fashion. [DSS-115] | Deleting the data of a name space removes the data from the blockstores in a serial fashion (in other words, one blockstore at a time). When a system has run full, it takes up to the n-th blockstore before the system can ingest data again. |
| The system logs unclear error messages when creating an already existing directory, or a directory with a missing parent directory. [DSS-549] | Creating a directory that already exists results in a `HTTP 405 Method not allowed` error.<br>Creating a directory when its parent directory does not exist results in a `HTTP 409 Conflict`. |
| Pulling out a switch does not immediately result in operations taking the bandwidth of the remaining switch. [DSS-601] | When one of the back end switches fails, all existing PUT operations fail (until they time out). New PUT operations that are started after the failure succeed, but could also time out. Only when all blockstores have been contacted is there no further connection attempts to the failed switch, and hence no more failed PUT operations. Depending upon the size of the objects and superblocks, it can take up to 15 minutes before the full bandwidth capacity of the remaining NIC is used. After restarting the client daemons, it takes up to 3 minutes to achieve a steady state situation.<br>Workaround: Adjust the timeout values on the client daemons. Restart the client daemons. |
| The **Upload** button on the client daemon web-interface does not work when authentication is enabled. [DSS-611] | The **Upload** button does a redirect to a location on which the logged-in user does not have sufficient rights. There is no API available that allows to provide these rights to the user. Therefore, the system cannot allow the operation when authenitcation is enabled. |
| When two back-end switches fail shortly after one another, PUT and GET operations hang. [DSS-747] | As stated in the previous known issue, it can take a while before enough connections are migrated from one network to another. If the failed switch is restored, it takes a while before all connections are migrated back and well balanced over both switches. If the second failure happens too soon after the first, not enough connections are migrated back, and ongoing PUT/GET operations hang, because there are not enough available connections.<br>Workaround: Adjust the timeout values on the client daemons. Restart the client daemons. |
| Inconsistent output of REST directory entries listing. [DSS-869] | When requesting directory entries in REST, using the various application types, the JSON output is inconsistent, but the XML output is okay. |
| You cannot upload files when the `IF_NONE_MATCH HTTP` header is defined together with user and password. [DSS-1037] | Due to a bug in the `libcurl` library, when trying to PUT a file while defining a user, a password, and an `IF_NONE_MATCH` HTTP header, you get a failure instead of success.<br>The curl initially writes (PUT) a file of 0 bytes. Then the curl makes a second PUT request on the same connection to write the actual contents of the file. Since the file already exists because of the first PUT, the second PUT fails. |

| Issue | Details |
|---|---|
| The client daemon returns an empty `HTTP 200` response instead of an error if not enough blockstores are online. [DSS-1108] | If two or more storage daemons are stopped in your environment and you try to read a file that was stored on those storage daemons, you should get an error. Instead, the S3 interface returns an `HTTP 200` response with a correct header, but with either empty content or an `incompleteread()` message. |
| Rebalancing for small file support does not work. [DSS-1146] | With the introduction of small file support, rebalancing has been disabled because the spreading of the full-copy was not done in a homogenous way and a rebalance could put too much strain on a single disk.<br>Workaround: See KB article BSP038. |
| The S3 interface does not support object names with a leading forward slash. [DSS-1436] | The S3 interface does not support object names that start with a forward slash (`/`). It also fails on authentications with a leading forward slash. |
| Many GET operations make the log file grow too fast. [DSS-1443] | Many GET operations make the BitSpread log file grow too fast. This typically occurs with low bandwidth setups. You see a lot of lines in the log file appear with the following content:<br>`wr_sb_duration >= 10.0s` |
| A failed PUT operation writes metadata to Arakoon anyway, but cannot be deleted afterwards. [DSS-1474] | When a PUT operation fails, there is still metadata written to Arakoon. It is impossible to remove the orphaned metadata. |
| PUT object with presigned URL fails if special headers are included. [DSS-1752] | Putting an object with a presigned URL and an `x-amz`, `x-amz-acl`, or `x-amz-meta` header fails. Without these `x-amz*` headers, the PUT with a presigned URL succeeds. |
| Outstanding objects and incomplete multipart uploads are not deleted. [DSS-1882] | When you delete an S3 bucket, you do not delete possible outstanding objects or upload parts of an incomplete multipart upload. You can no longer see these objects, but they remain on disk. |

### 2.2.5 S3 REST Interface

| Issue | Details |
|---|---|
| The `GET /` command on the S3 service returns an `HTTP 500` error. [DSS-1262] | The `GET /` command, when pointing a browser at an S3 end-point, results in an `HTTP 500` error. |
| The `PUT` command fails when issued through `s3cmd` with an extra forward slash (`/`) in the path name. [DSS-1263] | If you try something similar to the following, you receive an `HTTP 403: SignatureDoesNotMatch` error:<br>`./s3cmd put ../../testfile.txt s3://hbuck//testfile.txt` |
| Copying an object with an invalid option in `x-amz-metadata-directive` fails, with an error response that is different from AWS S3. [DSS-1271] | If you copy an object using the S3 interface and a value other than `COPY` or `REPLACE` is set in the `x-amz-metadata-directive` header, the error message sent in the HTTP response by the AWS S3 Interface is different from the message returned by the Active Archive System S3 interface. |
| The Active Archive System responds differently from AWS S3 when creating buckets with illegal characters. [DSS-1281] | If you create (a) bucket(s) that contain any of the following characters, the Active Archive System returns an `HTTP 501 (Not` |

| Issue | Details |
|---|---|
| | `Implemented)` instead of the expected `HTTP 400 (Bad Request)`:<br>`#,{,},\,<,>,[,], \|,`,^,"` |
| The authentication error in the client daemon log file does not explain why the authentication failed. [DSS-1286] | The output in S3 does not return the provided signature, but a `CDATA` field instead. |
| Service GET cannot handle query parameters like `delim` and `prefix`, which makes `Webdrive` fail. [DSS-1287] | When using a `Webdrive` client, listing the buckets does not work.<br>Workaround: Specify a bucket. |
| `GetObjectACL` returns an `HTTP 200 "OK"` for a non-existing object. [DSS-1288] | It should return an `HTTP 404 "Not Found"`. |
| The Active Archive System returns content type headers that are different from AWS S3. [DSS-1299] | The Active Archive System does not store the `content-type` information in its metadata. |
| The Active Archive System reports an `HTTP 501 (not implemented)` error when given a bad header or query param *and* the bucket does not exist. [DSS-1300] | This should return a "parameter problem" error, similar to AWS S3. |
| The Active Archive System returns an `InvalidBucketName` error for an S3 request with the wrong domain name. [DSS-1310] | When requests are received with the wrong domain name, the Active Archive System returns an `HTTP 400 (bad request)` error, whereas it should return an `HTTP 403 (forbidden).` |
| The Active Archive System behaves differently from AWS S3 for GET requests with multiple byte ranges. [DSS-1393] | When GET requests are received in which the `Range` parameter has multiple byte ranges in the header, the Active Archive System returns the whole object instead of the requested range, and concatenates the response in a non-intuitive way. |
| Forcibly creating a bucket on a syncstore with a special name generates an inconsistent HTTP error responses. [DSS-1398] | When buckets are forcibly created in a syncstore with an ID that is is not a well formed 32-character hexadecimal string, the Active Archive System returns inconsistent error responses. For example:<br>• For a syncstore with `aaaabbbbccccddddeeeeffff00001111`, the creation returns `404 Syncstore Not Found`.<br>• For a syncstore with `spaghetti_arrabiata` returns `400 Bad Request`, whereas it should be `400 Bad Syncstore ID`. |
| The Active Archive System behaves differently from AWS S3 for a cancelled multipart upload. [DSS-1498] | When canceling a multipart upload and then trying to upload other parts of the multipart, there is a difference in the behavior between AWS S3 and the Active Archive System.<br>AWS S3: socket timeout<br>Active Archive System: S3ResponseError: `404 Not Found`.<br>`<?xml version="1.0" encoding="UTF-8"?>`<br>`<Error>`<br>`<Code>NoSuchUpload</Code>`<br>`<Message>The specified multipart upload does not exist.` |

| Issue | Details |
| --- | --- |
| | <pre>The upload ID might be invalid, or the multipart<br>upload might have been aborted or completed.<br></Message><br><RequestId></RequestId><br><br><Resource>testfile7</Resource><br></Error></pre> |
| Sorting order for multipart uploads in progress is not based on creation time. [DSS-1521] | When listing multipart upload in progress, the sorting order is by object name. If the object name has multiple multipart uploads associated with it, they should be ordered by creation time, but they are not. |
| A socket timeout occurs when updating a multipart object. [DSS-1543] | When you update an object or multipart object with a regular PUT request *without* update permissions, you can get a socket timeout instead of an `access denied` response. |
| S3 requests fail when using unsupported time zones. [DSS-1561] | When using unsupported time zones (for example, Iran Standard Time - IRST), an S3 request fails with an `HTTP 500 Internal Server` error.<br><br>Supported time zones are: ADT, AFT, AKDT, AKST, ALMST, ALMT, AMST, AMT, ANAST, ANAT, ARST, ART, AST, AZOST, AZOT, AZST, AZT, BDST, BDT, BNT, BOT, BRST, BRT, BST, BTT, CAST, CCT, CDT, USA, CEST, CET, CETDST, CHADT, CHAST, CHUT, CKT, CLST, CLT, COT, CST, CXT, DAVT, DDUT, EASST, EAST, EAT, EDT, EEST, EET, EETDST, EGST, EGT, EST, FET, FJST, FJT, FKST, FKT, FNST, FNT, GALT, GAMT, GEST, GET, GFT, GILT, GMT, GYT, HKT, HST, ICT, IDT, IOT, IRKST, IRKT, IST, JST, KDT, KGST, KGT, KOST, KRAST, KRAT, KST, LHST, LINT, MAGST, MAGT, MART, MAWT, MDT, MEST, MET, METDST, MHT, MMT, MSD, MSK, MST, MUST, MUT, MVT, MYT, NDT, NFT, NOVST, NOVT, NPT, NST, NUT, NZDT, NZST, OMSST, OMST, PDT, PET, PETST, PETT, PGT, PHOT, PHT, PKST, PKT, PMDT, PMST, PONT, PST, PWT, PYST, PYT, RET, SAST, SCT, SGT, TAHT, TFT, TJT, TKT, TMT, TOT, TVT, ULAST, ULAT, UTC, UYST, UYT, UZST, UZT, VET, VLAST, VLAT, VOLT, VUT, WAKT, WAST, WAT, WET, WETDST, WFT, WGST, WGT, YAKST, YAKT, YEKST, and YEKT.<br><br>For definitions of time zone abbreviations, see http://docs.aws.amazon.com/redshift/latest/dg/time-zone-abbrevs.html. |
| Incomplete multipart objects are not deleted when their bucket is deleted. [DSS-1566] | When deleting a bucket which contains incomplete multipart uploads, the incomplete parts are not removed from the blockstore directories, thereby decreasing overall capacity of the system. |
| "GET /?versioning" response cannot be parsed due to missing new line character. [DSS-1568] | The response of the request `"GET /?versioning"` cannot be parsed due to a missing new line character after the `<?xml version ="1.0" encoding="UTF-8"?>` element. |
| The Active Archive System responds differently from AWS S3 for some time zone formats. [DSS-1569] | The Active Archive System may return different responses (`internal server error`, `forbidden`) to some `x-amz-date` time zone formats, compared to AWS S3. |

| Issue | Details |
|---|---|
| An S3 request without a date header returns `HTTP 500 Internal Server`. [DSS-1819] | An S3 request without a date header returns an `HTTP 500 Internal Server` error. This is the wrong error code. The correct behavior should be `HTTP 403 Forbidden` to be compliant with AWS S3. |
| An S3 request with `Expires` parameter is not supported. [DSS-1822] | An S3 request which contains the `Expires` parameter returns an `HTTP 501 Not Implemented` response because it is not implemented in the Active Archive System. |
| An S3 request with the `TE` parameter is not supported. [DSS-1827] | An S3 request which contains the `TE` (Transfer Encoding) parameter returns an `HTTP 501 Not Implemented` response because it is not implemented in the Active Archive System. |
| Uploading of a multipart object succeeds if its size is greater than `multipart_object_max_size`. [DSS-1833] | If you upload a multipart object whose size is greater than the defined `multipart_object_max_size`, the upload succeeds, whereas it should fail. |
| The latest versions of Cyberduck cannot be used to create buckets. [DSS-1915] | Some versions of Cyberduck report interoperability errors when connecting to an S3 client daemon with SSL enabled. An identified issue is that it is not possible to create buckets when using Cyberduck 4.6.3. |
| Uploading of a multipart object with the latest versions of Cyberduck raises an error. [DSS-1916] | When executing a multipart upload using Cyberduck 4.6, you get an error: `Upload failed due to a \| mismatch between MD5 hash {0} of uploaded data and ETag {1} returned by the server` even though the upload succeeded. |
| The Active Archive System S3 interface supports v2 signatures only. [HGST-573] | The Active Archive System S3 interface only works with the older V2 signature for GETs and PUTs. This means that you cannot use S3 clients such as s3cmd version 1.5, which uses a V4 signature. Attempts to use a signature newer than V2 result in exceptions that are logged in the client daemon log, such as:<br>S3 GET<br>`Apr 27 17:09:38.6116 info [11307] connection from inet:192.168.201.51:55576 to inet:192.168.3.11:7071, fd 567: accepted`<br>`Apr 27 17:09:38.6117 info [11307] starting protocol`<br>`Apr 27 17:09:38.6146 info [11307] [a235ebeefad944f0be8e91d8bdf7892a] Authentication succeeded for user 'hive'`<br>`Apr 27 17:09:38.6147 info [11307] [a235ebeefad944f0be8e91d8bdf7892a] user 'hive' successfully authenticated`<br>`Apr 27 17:09:38.6148 error [11307] [a235ebeefad944f0be8e91d8bdf7892a] action: 'GET' is disabled or not allowed on the service`<br>`Apr 27 17:09:38.6151 info [11307] connection from inet:192.168.201.51:55576 to inet:192.168.3.11:7071, fd 567: closed`<br>`Apr 27 17:09:38.6152 error [11307] exception 'S3Errors.S3Failure(_)' ends session` |

| Issue | Details |
|---|---|
| | S3 PUT<br>`Apr 27 17:10:03.6841 error [11312] [c862c2326653406eb78abe1bc17d98d9] Invalid x-amz-date header: 20150428T001003Z`<br>`Apr 27 17:10:03.6845 info [11312] connection from inet:192.168.201.51:55580 to inet:192.168.3.11:7071, fd 567: closed`<br>`Apr 27 17:10:03.6845 error [11312] exception 'S3Errors.S3Failure(_)' ends session` |

### 2.2.6 Failover

| Issue | Details |
|---|---|
| A machine reboot event is incorrectly shown after a Management Node failover. [AMIF-4067] | When executing a failover of a Management Node, there is an event generated that the new Management Node is rebooted, even though this new node has not been rebooted. |
| Upgrade patch history is incomplete when the Management Node is replaced and then failed over to another node. [AMIF-4173] | After executing a failover to a new Management Node, the upgrade patch history is lost when checking the **About** section in the CMC. This is because after a failover, there are no patches available on the new Management Node.<br>This has no impact on the functionality of the Active Archive System. |
| During a failover, too many messages appear when installing the PostgreSQL and Apache packages. [AMIF-4365] | A failover is typically executed in a screen session. When the failover installs the PostgreSQL and Apache packages, an overload of messages appear. |
| During a failover, the `pound` application might be restarted on the old node before it gets started on the new node. [AMIF-4402] | During a failover, `pound` is stopped on the old Management Node. It may occur that the monitoring agent detects this action and that it initiate a restart of `pound` on this old node. If this happens, the failover will fail because `pound` cannot be started on the new node.<br>Workaround: Disable TLS and restart the failover. After completion of the failover, you can enable TLS again. |
| Apache is still running on the old Management Node after failover. [AMIF-4838] | If you execute a failover while the old Management Node is running, Apache is still running on the old Management Node even when the failover completes successfully. |
| Failover does not succeed when nodes are not reachable through the management network. [AMIF-5064] | When you have nodes which are not reachable over the management LAN, a failover does not succeed, because the management services cannot be failed over to another Controller Node in this scenario. An unsuccessful failover means that the old Management Node remains as the Management Node. |
| Custom SNMP polling community string is lost after failover. [AMIF-5066] | When you perform a failover, the community string for custom SNMP polling is reset to the default value. |
| Failover may not succeed when executed immediately after adding new nodes. [AMIF-5084] | When executing a failover immediately after adding a new node, the failover may not succeed. This may be caused by the fact that the model was not yet backed up to Arakoon. |

### 2.2.7 Monitoring

| Issue | Details |
|---|---|
| Certificate expiration dates for monitoring are converted from the local time zone and not from GMT. [AMIF-3259] | The certificate expiration time is converted based upon the local time of the CMC machine, instead of based upon GMT. Because of this, there might be a time span of a few hours where monitoring/events differ from reality. |
| The monitoring agent repeatedly logs the `task python:PID blocked for more than 120 seconds` error in `kern.log`. [AMIF-3534] | During the run of the tlog collapse policy, the monitoring agent repeatedly logs the previous named error in the kernel log. This error has no further impact on the system, nor do any events trigger because of this. |
| A MetaStore is not marked as degraded when one of its nodes is unavailable. [AMIF-4078] | When a node of a MetaStore is unavailable, the MetaStore should be marked as degraded but it is not. |
| Failure of machine agent is not auto-corrected. [AMIF-4170] | On some rare occasions, it is possible that checking the availability of an agent fails. This may result in an agent who seems to be connected, but in reality is offline. As a result, this can cause several agent scripts to fail as there is no agent available to execute these scripts. In normal situations, the monitoring should detect that an agent is unavailable and automatically restart it.<br>Workaround: Restart the application server on the node whose agent is unavailable. |
| An I/O error on blockstore blocks the check of other disks. [AMIF-4426] | When the monitoring agent checks the blockstores of a node and there is an I/O error on one of the blockstores/disks, the monitoring agent does not check the remaining blockstores/disks. As a result, the other blockstores may remain unmonitored until the disk is decommissioned. |
| Deleting a bucket containing a huge number of objects causes `MetaStore lagging` events. [AMIF-4481] | When you delete a bucket containing millions of objects, the Active Archive System might raise `MetaStore lagging` events.<br>Workaround: Monitor the event log to make sure these events eventually disappear. If they disappear, all is fine, but if they do not, contact HGST support. |
| The event message `blockstore path is offline` is raised for manually offlined blockstores. [AMIF-4609] | When setting blockstores manually to offline in the CMC, events are raised for those blockstores indicating that they are offline (event OBS-DSS-BLOCKSTORE-0033). |
| A deleted unmanaged disk is added again with the next monitoring cycle. [AMIF-4637] | When deleting an unmanaged disk from the CMC, the monitoring cycle starts and adds the disk again as unmanaged disk. This is repeated until the disk is either physically removed or repurposed. |
| The monitoring agent raises an event for halted MetaStore during Arakoon recovery. [AMIF-4710] | After an ungraceful shutdown of a Controller Node, an Arakoon recovery is automatically initiated, where the Arakoon is temporarily halted. For this reason the monitoring agent raises an event and tries to restart it, but the restart fails, resulting in another event (OBS-ARAKOON-0018). This last event can be confusing since it may give the impression that the recovery is not initiated.<br>In this scenario, you can safely ignore the event. |
| The failed login attempt log provides insufficient details. [AMIF-4772] | When you have a failed login to the CMC, an event is generated. The event does not log the source IP address from which the login has been attempted. |

| Issue | Details |
|---|---|
| The monitoring agent still uses the old hostname of a node. [AMIF-4843] | When renaming a node's hostname, the monitoring agent still uses the old hostname in events.<br>Workaround: Restart the monitoring agent. |
| The monitoring agent does not restart when its process is "defunct". [AMIF-4879] | In some situations, the monitoring agent process might become defunct. In that case you can no longer restart the monitoring agent. Consult KB article BDY069 |
| The monitoring agent enters an unknown state when the `kipmi0` process does not respond. [AMIF-5022] | When the `kipmi0` process does not respond, the monitoring agent also stops responding and gets into an unknown state.<br>Workaround: Restart the monitoring agent manually or wait for it to be reset when the monitor policy runs. |
| The monitoring agent does not detect `ixgbe module verification failed` messages. [AMIF-5027] | The monitoring agent does not detect the following error message in the kernel logs and as such does not raise an event: `ixgbe: module verification failed: signature and/or required key is missing - tainting kernel.` This error may be safely ignored. |
| The thresholds for monitoring free space on tlog partitions are wrong. [AMIF-5035] | The monitoring agents checks the free space thresholds on tlog partitions in such a way that the critical threshold is always reached first. As a result, a MetaStore is automatically set to **FULL** before you are warned about reaching error thresholds. |
| Machine rebooted events are raised when restarting the monitoring agent. [AMIF-5080] | When you manually restart the monitoring agent on a node, you may get events indicating that the node has rebooted. This has no functional impact. |
| The event message for OBS-STORAGE-0004 is updated unexpectedly. [AMIF-5107] | When the event OBS-STORAGE-0004 is raised, the event message contains environment statistics, showing the number of:<br>• degraded disks<br>• decommissioned disks<br>• offline disks<br>• healthy disks<br>When this event occurs again within the deduplication period, the event message is overwritten and no longer shows the environment statistics. |
| Deleted objects are included when calculating `object_name_length_stats`. [DSS-1727] | When calculating `object_name_length_stats` information, deleted objects are also taken into account, but they should be omitted. |
| When a monitoring agent failure occurs on a Storage Node, the Storage Node generates the following event:<br>`Errors detected on machine HGST-Alpha02-DC01-R02-SN06: Error retrieving block store information for [547]`<br>[HGST-269] | This event (OBS-APPLICATION-0050, ERROR level) is sometimes generated with moderate traffic on the system, but the problem it reports resolves on its own. No action is needed. To verify that the problem will resolve on its own, compare the monitoring agent logs from where the event was generated with the Apache logs from the Management Node as follows:<br>1. Log into the machine that generated the event. In the example below, the machine is `HGST-Alpha02-DC01-R02-SN06`.<br>2. Open the monitoring agent log in the directory `/opt/qbase3/var/log/monitoringagent/` that corresponds to the time of the event. You will see a trace similar to below. This particular |

| Issue | Details |
|---|---|
| | issue will show the exception that references a `ProtocolError` with a `-1` return code (`/appserver/xmlrpc/: -1`).<br><br>For example:<br><br>`2015-02-22 18:17:03,369: utils.py: ERROR: ['Traceback (most recent call last):\n', '  File "/opt/qbase3/apps/monitoring_agent/monitoring/dssstoragedaemons.py", line 276, in _get_blockstore_info\n storagepool_data=storagepool_data)\n', '  File "/opt/qbase3/lib/python/site-packages/framework/utils/dss_utils.py", line 416, in max_files_for_blockstore\n return min(max_for_blockstore, _get_max_files_for_syncstores(_get_nr_syncstores()))\n', '  File "/opt/qbase3/lib/python/site-packages/framework/utils/dss_utils.py", line 449, in _get_nr_syncstores\n return len(api.metadatastore.find(metadatastore_type=\ str(q.enumerators.metadatastoretype.SYNCSTORE))[\'result\'])\n', '  File "/opt/qbase3/lib/python/site-packages/framework/utils/cloudapi_proxy.py", line 74, in __call__\n    raise ex\n', 'CloudApiConnectionException: \ <ProtocolError for cloudapi:acf25f54-a68e-11e4-b67c-90e2ba7c5214@10.1.12.154:80/appserver/xmlrpc/: -1 >\n']`<br><br>**3.** Next, open the Apache error log `/opt/qbase3/var/log/apache/error.log` on the Management Node. You will see a corresponding message in this log that Apache received a `SIGHUP` and was restarting.<br><br>For example:<br><br>`[Sun Feb 22 04:17:04 2015] [notice] SIGHUP received. Attempting to restart`<br>`[Sun Feb 22 04:17:04 2015] [notice] Apache/2.2.22 (Ubuntu) mod_ssl/2.2.22 OpenSSL/1.0.1 configured -- resuming normal operations`<br>`[Sun Feb 22 18:17:04 2015] [notice] SIGHUP received. Attempting to restart`<br>`[Sun Feb 22 18:17:04 2015] [notice] Apache/2.2.22 (Ubuntu) mod_ssl/2.2.22 OpenSSL/1.0.1 configured -- resuming normal operations`<br><br>**4.** The time of the Apache restart should be correlated to the monitoring agent failure. |
| There are errors retrieving information from the storage daemons. [HGST-408] | When you collect log files that cover a large time span, you may cause the logging partitions to reach full capacity, which in turn affects storage services.<br><br>Workaround: Collect logs for a single date at a time. If you need to collect logs for multiple dates, run the log collector multiple times, once for each date needed. |
| The collection and sending of telemetry data may stop on Storage Nodes. [427] | If you change the system date, time, or time zone, telemetry collection may stop on all or some Storage Nodes. It continues to run on Controller Nodes. |

| Issue | Details |
|---|---|
| | Workaround: Run the following command at the Linux prompt on all nodes where the time or time zone has been changed:<br>`sudo service cron restart` |

### 2.2.8 Non-Critical Third-Party Vulnerabilities

Vulnerabilities have been detected in some of the third-party packages used by the Active Archive System.

| Issue | Details |
|---|---|
| PostgreSQL vulnerability [AMIF-4486] | There are several issues that may lead to a possible denial of service or execution of arbitrary code. |
| Python Security Notice [AMIF-4500] | Python can be made to crash or run programs if it receives specially crafted network traffic. |
| initramfs-tools vulnerability [AMIF-4551] | Incorrect mount options. |
| Linux Kernel vulnerability [AMIF-4572] | There are several security issues, information leaks, and buffer underflow errors. |

### 2.2.9 Supportability

| Issue | Details |
|---|---|
| The workflow engine is not automatically restarted when it is stopped or killed. [AMIF-258] | The automatic restart of an application requires both Apache and the workflow engine to be running. If they are not running, they cannot be restarted automatically.<br>Workaround: Monitor these applications with an external tool. This way, downtime does not go unnoticed. |
| The root file system can fill up when a lot of core files are created within a period of 1 week. [AMIF-892] | Core files are compressed and those older than 7 days are removed, but if a component produces a lot of core files within this 7 day window, the root file system could fill up. Events are raised when core files are detected. |

### 2.2.10 Various

| Issue | Details |
|---|---|
| The Active Archive System tries changing the `admin` on a node marked **FAILED**. [AMIF-2985] | If you change the `admin` password of your environment while it has a node marked **FAILED**, the Active Archive System tries (and fails) to change the password on the failed node as well. The password change succeeds in your environment, but also displays a pop-up window that it failed on the failed machine.<br>Workaround: Run the health checker to verify that all nodes are healthy prior to changing the password. |
| After a power cycle, the Active Archive System does not automatically resume operations because PID files are lingering around. [AMIF-3645] | When a Controller Node (and more specifically the Management Node) is power-cycled (in other words, rebooted in an uncontrolled fashion), upon restart, some of the PID files (used to prevent starting multiple instances of the same process) are not cleaned up. This prevents the restart. |

| Issue | Details |
|---|---|
| | Workaround: Identify the process that refuses to start and to remove its PID file. Restart the application server manually with the following command:<br>`q.manage.applicationserver.restart()` |
| The log collector returns unsorted results when specifying a time window. [AMIF-3782] | When you specify a time window, the log collector returns unsorted results. You may also lose log data when you specify time keywords.<br>Export the results, and then sort it. |
| After changing the password of the `cloudapi` user, login failure events are raised. [AMIF-3989] | It is possible to change the password of the `cloudapi` user through the CMC. When you do so, many `Failed login attempt with username cloudapi` events are raised. This is because the application server keeps the old password in its cache.<br>Workaround: Restart the application server after updating the password. |
| Real-time statistics in OSMI do not show a timeout range for the collection process. [AMIF-4027] | When you want to collect real-time statistics in OSMI, you can only provide a timeout value in seconds in the range between `120` and `100,000`.<br>Workaround: Specify a value within `120-100,000` seconds. |
| The kernel log contains the message `Program lshw tried to access /dev/mem between ff000->101000`. [AMIF-4219] | The `lshw` command might try to read information from memory contents. When doing so, a warning `Program lshw tried to access /dev/mem between ff000->101000` is added to the kernel log.<br>This warning has no further impact on your system. |
| Cleanup of a machine shows errors because it cannot be removed from the initialization queue. [AMIF-4273] | The cleanup of a machine tries to remove the machine from the initialization queue. When the machine is not in this queue, this step fails as expected, but the cleanup continues and completes successfully. |
| Updating the network routes fails when there are nodes in a **Configured** status. [AMIF-4384] | When you have nodes in a **Configured**" status, the job to update the network routes fails. This has no impact on the running nodes; these are updated correctly with the new network routes.<br>Workaround: Clean up the configured machines with in the CMC:<br>1. Navigate to **Dashboard** > **Administration** > **Hardware** > **Servers** > **Unmanaged Devices** > **Failed**.<br>2. Select the nodes and click **Cleanup Devices** from the **Commands** pane. |
| Arakoon recovery may leave `.part` files. [AMIF-4534] | When an Arakoon recovery is interrupted between the creation of `.part` files and uploading them, the `.part` files remain on the file system, instead of being removed. |
| DNS update only works on first LAN. [AMIF-4537] | When you have multiple public LANs, the update of DNS will only work on the first configured LAN. If you update the DNS of another public LAN via the CMC, the job succeeds, but the update is not persisted on the system. |

| Issue | Details |
|---|---|
| Under load, hot swap may fail due to unmounting disk failure. [AMIF-4654] | When a system is under load, the hotswapping of a Controller Node or Storage Node disk might fail due to an unmount failure. This results in a disk with a failed initialization. |
| It is possible to repurpose a disk while it is being initialized. [AMIF-4564] | When you repurpose a disk, it is put into the list of unmanaged disks. This disk remains in the list of unmanaged disks until the system fully initializes it. While it is being initialized, it is possible for you to repurpose it again. The system should not allow this. |
| PDF export of disks may fail in Google Chrome browser. [AMIF-4611] | When using the Google Chrome browser, exporting a PDF of the decommissioned disk details fails. This issue has been identified in Google Chrome 34.0.1847.116-1 and may not be limited to this version. |
| The health check fails due to decommissioned storage daemons. [AMIF-4628] | During the health check, you are asked whether to start a storage daemon in case the health check finds a storage daemon that is not running. If you answer `No`, because you decommissioned the storage daemon, the health check fails, which is the incorrect action. |
| Kernel dmesg events with empty call trace are raised after power cycle. [AMIF-4646] | When restarting a Controller Node you may receive kernel dmesg events with an empty call trace. |
| Hotswapping of an SSD is not possible if it contains a cache daemon. [AMIF-4655] | It is not possible to hotswap an SSD which contains a cache daemon, because the daemon is not stopped. The cache daemon must be halted before you can hotswap the SSD. |
| The health check fails on TFTP. [AMIF-4662] | The health check might fail when checking TFTP. This is due to writing multiple PIDs in the PID-file. |
| The root password is saved in a Q-Shell cluster configuration file. [AMIF-4752] | When you create a Q-Shell cluster, its configuration file stores the root password.<br>Workaround: Remove the Q-Shell cluster after having executed the necessary actions on the cluster. |
| Decommissioning a disk when its agent is unavailable results in its blockstore being decommissioned. [AMIF-4771] | When you decommission a disk on a node when there is no agent available, its blockstore is decommissioned, even though the disk is not decommissioned. This is because local updates happen first, and these updates are not undone when the decommissioning job fails.<br>Workaround: Decommission the disk again when the agent is back up and running. |
| `Rscript` cannot be killed when the node its running on is unreachable. [AMIF-4871] | When a node becomes unreachable while an `Rscript` is being executed on this node, the `Rscript` hangs and cannot be killed. The `Rscript` cannot be re-executed and may impact the functionality of the system. For example, when retrieving aggregate graphics, they would omit the unreachable node's data. |
| During startup of Controller or Storage Nodes there is a recommendation to use a different driver. [AMIF-5040] | When starting a Storage Node or Controller Node the dmesg may show the following recommendation:<br>`ACPI: If an ACPI driver is available for this device, you should use it instead of the native driver.`<br>This message may be safely ignored. |

| Issue | Details |
|---|---|
| The original serial number of a disk is not restored when `initialize_new_disk` fails. [AMIF-5058] | When the initialization of a new disk fails, the serial number of the disk it replaced (the old/failed disk) should be restored back into the system model, but it is not. Instead, the system model now contains the ID of old/failed disk, but the serial number of the new disk. There is no impact on the functionality of the system. |
| A failed restart of a node sets the status of the node to **RUNNING**. [AMIF-5076] | When the restart of a node fails, the node status is still set to **RUNNING** instead of the expected **HALTED**. |
| It is possible to decommission more than one Controller Node at the same time. [AMIF-5216] | It is possible to decommission more than one Controller Node at the same time, leading to an unavailable system if these nodes contain Arakoon nodes of the same MetaStores, especially the system MetaStores.<br>Workaround: Do not decommission more than one Controller Node at a time. |
| Abandoning a blockstore with `force` flag fails. [DSS-1080] | If a Storage Node is hung and still responding to pings, but not to SSH connections, abandoning blockstores and storage daemons fails on that node. |
| It is possible to create a namespace on a full MetaStore. [DSS-1440] | When you mark a MetaStore as full in the Q-Shell, you can still create a name space (in the Q-Shell) with that MetaStore, if you explicitly specify the MetaStore. However, you are not able to add any data to the new name space. |
| When using a range that starts at the boundary of a superblock, a "Bug" message appears in the log file. [DSS-1570] | When using a range that starts at the boundary of a superblock, an error with a "Bug" message is added to the log file:<br>`"Bug: this should not happen: obtained an empty range...` |
| The Active Archive System ignores drives with a different serial number than expected. [HGST-89] | The Active Archive System does not automatically recognize a new drive as being the replacement for another drive.<br>Workaround: Decommission a drive manually or automatically before replacing it. Ensure that you are replacing the correct drive. For more information, see the *HGST Active Archive System FRU Replacement Guide*. |
| Enabling the model database backup policy fails because it reuses a cached client daemon connection that is no longer active. [HGST-144] | When the client daemon attempts to use a cached connection and it fails, the connection is marked in an internal registry which manages the connection list. Every 30 seconds, the registry is inspected and a new connection is attempted for failed connections. |
| The Active Archive System cannot retrieve SMART attribute statistics from Storage Enclosure Basic drives. [HGST-263] | The Active Archive System can only obtain a log of SMART events, rather than actual SMART attributes, such as:<br>`Raw_Read_Error_Rate`<br>`Spin_Up_Time`<br>`Start_Stop_Count`<br>`Reallocated_Sector_Ct`<br>`Seek_Error_Rate`<br>`Power_On_Hours`<br>`Spin_Retry_Count`<br>`Calibration_Retry_Count`<br>`Power_Cycle_Count` |

| Issue | Details |
|---|---|
| | Power-Off_Retract_Count<br>Load_Cycle_Count<br>Temperature_Celsius<br>Reallocated_Event_Count<br>Current_Pending_Sector<br>Offline_Uncorrectable<br>UDMA_CRC_Error_Count<br>Multi_Zone_Error_Rate |
| The Active Archive System does not securely erase Storage Enclosure Basic drives during decommissioning. [HGST-306] | To work around this problem, log into the Storage Node associated with the decommissioned drive and run the following command at the Linux prompt:<br>`sg_sanitize --crypto DEVICE` |
| During a software update, automatic recovery may fail. [HGST-597] | During a software update, you may observe the following event:<br>`Event Message: MetaStore instance metastore_name::node_name automatic recovery failed after 5 retries on machine [hostname]`<br>This is a benign issue that in which the monitoring agent attempts to restart the metastore while it is being restarted as part of the upgrade and can be ignored while the upgrade is in progress. |