Operating System Release Notes

Version 6.1.0

2021-02-04

Supports:

Azure FXT Edge Filer (all models)

Avere FXT (all models)

Avere vFXT for Azure

Table of Contents

What's new in version 6.1.0

This release includes updates and bug fixes, including the following resolved issues.

Resolved issues

Filesystem

5927379	Upgraded the OpenSSL library to version 1.1.1.
8719438	Fixed a race condition in the file removal path that could cause the filesystem service to restart.

General

5728834	Reduced the amount of time it takes to update software on Avere vFXT for Azure clusters by eliminating interim rebalancing steps.
8888189	Fixed an issue in the FXT 6000 series FreeBSD NIC driver that prevented sent packets from being captured during packet captures.
9021237	Fixed an issue that made the "create a new cluster" web UI unusable on unconfigured nodes.

NFS

7956790	Updated implementation of the XML-RPC method `corefiler.setMountExports` (introduced in version 6.0.0; see item 6213403 below). The method now synchronously validates that an export exists on the core filer by sending it a mount command. The `corefiler.setExportFilters` and `corefiler.getExportFilters` methods have been removed. If you have core filers configured with non-default export filters, they continue to work but the filters can't be modified. Consider removing any affected core filers and re-creating them by using the `corefiler.create` method with the option `useSetMountExports` set to **yes**.
8074492	The XML-RPC methods for setting anonymous user ID (anonuid) and anonymous group ID (anongid) values in export rules have changed. These changes affect the behavior of the methods `nfs.addRule`, `nfs.modifyRule`, and `nfs.listRule`. Here is an overview of the changes: • The anongid value can be explicitly set and viewed. Before this change it was automatically set to the anonuid value. • Values for anonuid and anongid are now supplied and stored as strings instead of as integers. A wider range of values is allowed.

- Previous behavior is preserved for backward compatibility.

Setting anongid:

The `nfs.addRule` and `nfs.modifyRule` methods now accept separate anonuid and anongid values.

- If anongid is specified, you must also specify anonuid.
- If only anonuid is specified, its value is used for both anonuid and anongid.

The `nfs.addRule` parameter `settings_struct` and the `nfs.modifyRule` parameter `setting_struct_array` now include anonuid and anongid values passed as strings. The positional parameter anonuid, which was passed as an integer, is available only for backward compatibility and cannot be used if anonuid or anongid are specified.

Valid values:

Previously, anonuid and anongid could only be set to integer values of -2 or 65534 (nobody), -1 or 65535 (no access), and 0 (root without superuser privileges) could be specified. The new system preserves these values but adds more flexibility.

- The string values -2, -1, 0, 65534, and 65535 still are mapped as in the past.
- Other values from 1 to 4294967295 are supported for both anonuid and anongid. They are mapped with no special treatment.

The XML-RPC help files for these methods have more complete information about the new use cases.

SMB

8654249	Fixed a defect where an internal event could cause a service restart and core file from the SMB file system or SMB name server services.

What's new in version 6.0.0

This release includes bug fixes, improved support for Azure products including Azure HPC Cache, and security improvements.

Upgrading to version 6.0.0

Customers can upgrade directly to directly to this OS version if their systems are using Avere OS version 5.3.1.2 or later.

If your system is using 4.8 software, you must update to the most recent 5.3. version before you can install this release. Contact Microsoft Service and Support if you need assistance.

You cannot downgrade to an earlier OS version after installing version 6.0.0.

Resolved issues

Cloud computing

6435492	Increased cloud file system cache size to 5GB for smaller vFXT instances and to 17GB for E32 series and other large memory vFXT instances. This change prevents some issues related to memory consumption.
6446744	Fixed an issue that could cause TLS connections to be opened too frequently to cloud back-end storage, resulting in peer resets.
7081961	Fixed a problem with space calculations for caches located in the cloud. In some situations, the problem could cause the cache to stop performing operations.
7824839	Enabled Server Name Indication (SNI) in TLS connections to back-end object storage. This change prevents connection failures when communicating with services that require SNI information from systems that support SNI. This change affects Azure and other object store providers.

Filesystem

5126091	Increased the default chunk size to 32 to speed up file truncation in cloud environments and improve performance.
5143125	Corrected an issue in the directory manager rebalance process during node reformat. Before the fix, this issue could cause a node rebalance operation to get stuck.
5186590	Improved cache population speed by fixing a slowdown in the cache filesystem consistency check during read-ahead.
5239722	Fixed an issue where earlier errors could cause the core filer bandwidth control to interact with the data flush process and cause problems.
5371778	Improved the cloud file system version database recovery during failover.
5407127	Improved the synchronization of metadata files during recovery.
5409620	Fixed a lock contention issue that could cause high CPU usage.
5447895	Improved read speed for cold directories.
5460210	Fixed an in-memory inconsistency in marshaling parameters.
5534498	Fixed a crash related to a race condition when recycling an internal state.
5581410	A default setting has been changed in the built-in cache policy Full Caching. The maximum writeback delay in this policy now defaults to one hour. Previously, the maximum writeback delay was set to 10 minutes by default. The value can be customized by using the **Core Filer** > **Manage Cache Policies** settings page for an FXT or Avere vFTX for Azure cluster. This change only affects core filers that are created with Avere OS 5.3.3.1 or later software. If a cluster is upgraded to 5.3.3.1, core filers that were created before the

	update keep the original 10-minute default writeback delay, but any new core filers configured will have the 1-hour default delay.
5645667	Fixed an issue to free space on the HA partner node after deleting a file.
5651782	Fixed a race condition that could cause memory leaks related to a secondary policy in a file delete path. This change also suppresses a related alert.
5720169	Added code to correctly detect Azure response header error for a missing key vault encryption key, and to return the appropriate alert.
5733946	Fixed a crash seen during a data dump to debug leaked memory buffers.
5736836	Fixed an internal race condition related to cache directory metadata that could cause a restart.
5865574	Disabled Bloom filters for directory segments with names that can cause a rare type of hash collision. This change impacts only segments with the matching names, not the whole directory. This fix prevents a possible system restart.
5877594	Added age information to data dump entries for failed metadata operations.
5905389, 6031250	Improved accuracy in directory flushing and fixed a bug that could cause a crash due to a concurrent access issue during a data dump of buffer pools.
5906986	Improved client operation failover handling the event of a cluster data manager outage.
5986301	Fixed an issue in metadata file synchronization during recovery that could cause a service restart.
5990301	Fixed an issue with OpenSSL memory allocation initialization that could result in a crash.
5992267	Improved performance when removing files.
6034136	Improved failover performance during recovery.
6254713	Fixed a problem that could cause the link count to be off by one if a directory was renamed to the same name as an existing directory that was empty. (The existing empty directory would be overwritten but still appeared in the count.)
6344590	Changed internal operation handling to improve down time during HA failover.
6446516	Fixed an issue that could cause a data flush operation to an Isilon core filer to stop progressing.
6562456	Improved failover timing during the recovery process.
7046266	Fixed an XML parsing error in responses that could cause a service to restart.
7119939	Fixed a bug in read-only caching code that could cause nodes to restart.
7216091	Fixed a problem that could cause a crash when a core filer forwards a read request to another core filer.
7261813	Fixed an issue with certificate verification to allow failover to CRL if OCSP is not available.
7715919	Improved ability to diagnose cluster communication failures.

7741219	Fixed an issue in handling metafile synchronization during a service restart.
7851695	Fixed a problem related to the file deletion path that could cause a service restart due to a race.
7856062	Changed code to more reliably update the cluster data manager's configuration database.
7970316	Fixed a problem related to recycling internal directory-related state, which could cause a crash.
8126540	Additional work was done to fix issues with group IDs outside the first 16 in the groups list. This issue affected customers who use extended groups. The symptom of this problem was that a file's group ID (GID) could not be changed if the GID was not one of the first 16 groups listed for the account.
8361454	Updated code to handle non-UTF8 characters when flushing directory log records. Before this change, some characters could cause an error that blocked the flush.
8632234	Fixed an issue that could cause a cluster OS upgrade to stall when updating from version 4.8 or older.

General

5096015, 5146122	Fixed a problem handling OS download URLs that include URL query parameters.
5133739	Fixed a GUI issue with XML-RPC response handling that incorrectly caused error messages to appear in the GUI, even though the XML-RPC command succeeded.
5153110	Improved internal logging for core filer creation jobs.
5254123	Fixed an issue that prevented the cluster TLS certificate from being regenerated when it was nearing expiration. After upgrading, you might see a new cluster TLS certificate presented by the control panel interface if the previous certificate was expired or nearing expiration.
5352452	Updated the help for the `node.allowToJoin` XML-RPC method to note that an activity tracker is returned if the allow-join action did not immediately complete.
5533584, 6622809	Updated TLS certificate generation for the cluster webserver. Changes include: • Added the Extended Key Usage and Subject Alternative Name (SAN) extensions • Updated the cluster web server certificate to include the internal cluster IP addresses as SANs
5580761	Fixed a problem that could show a node's state as "pending" when it was actually "up". The issue was related to a workflow in which all of the node's client IPs were homed away from the local node.
5877092	Added code to automatically detect Azure networking proxy settings and use them when connecting to Azure APIs.

6185556, 8058284, 8437737	Fixed an issue that caused support information gathering to be slow or to get stuck.
6213953	Addressed an issue that prevented the VLAN configuration from being set when using the `cluster.create` XML-RPC method.
6518120	Fixed a labeling issue that could cause confusion when conditions were included in the alert history.
6600899, 6600988	Updated the presentation of the End User License Agreement, Terms of Use, and Microsoft Privacy Statement in the cluster control panel.
5695188	Fixed a bug that prevented changing the verification time for read-only cache policies.
5729545	Fixed an issue in XML-RPC methods to ensure that an activity tracker UUID is returned in almost all cases of long-running operations. Before this change, XML-RPCs would sometimes return "success" instead of an activity UUID due to a race condition, even when the operation was not complete.
5926431	This change restored the previous default encoding value “double” for long integers in XML-RPC methods. In a previous release, the default XML-RPC marshalling (encoding) value for long integers was changed to i8 instead of double. This change could possibly require updates in client-side software, since methods that previously returned elements tagged as double would now return i8-tagged elements. With this release, the default long-integer encoding value has returned to the historic usage, “double.”
5954828	Eliminated a bug that could cause a corrupt SNMP configuration file to be created on the cluster, causing the snmpd server to restart over and over.
5955606	Fixed a bug that could cause errors when updating the directory services configuration.
5991352	Updated code to prevent a deadlock in the process responsible for refreshing cloud service credentials.
7035118	Fixed a problem in the `cluster.addVlan` XML-RPC method that was related to setting a VLAN’s MTU.
7061943	Some sleep calls in the HyperV drivers have been switched into busy wait loops to avoid kernel crashes due to illegal sleeping in the kernel while holding a mutex.
7098786	Fixed a problem that prevented the cluster from switching software images during an upgrade.
7112400	Fixed a bug that caused the cluster management process to crash repeatedly when removing a node management IP address range.
8035682, 8050881	Improved failover for sending secure email notifications from the cluster. Now, if TLS negotiation fails, connections are restarted without encryption if the SMTP server supports sending in plaintext. Emails are sent in plaintext until the next email notification job starts.

	The FXT cluster will not send a password across an unencrypted connection, so this fallback works only when authentication is not required. To avoid TLS negotiation failures, make sure that your SMTP server uses cipher suites that are compatible with the Azure FXT Edge Filer operating system. This change also fixes a system restart that was associated with a problem in the email utility.
8087992	The memory available to the main software process in FXT 6000 series nodes has been increased by 25GB.This change significantly reduces swap space usage.
8324059	Fixed an issue that prevented cluster nodes from getting the required TLS certificates when upgrading OS software. This problem affected clusters that included nodes that had used software older than 5.3.1.2 and nodes that had never run software before 5.3.1.2.
8325030	Fixed an incompatibility issue that could prevent OS upgrade in a cluster with a mixture of FXT 6000 series hardware nodes and older models.
8657279	Fixed a bug that could cause the management service (mgmtd) to crash at startup when network connectivity issues prevented it from communicating with other nodes. When network link aggregation was in use, the bug in item 8675331 could cause this crash.
8675331	Fixed a bug that caused the management service (mgmtd) to write an erroneous temporary network configuration at process startup that caused network link aggregates to be torn down. On some hardware platforms, this teardown process could take a long period of time, causing a loss of network connectivity and leading to a mgmtd crash (item 8657279) and/or a kernel core (item 6014464).
8690658	The network manager process has been changed to disable the link aggregate interface instead of destroying it when a link aggregate configuration is removed. The aggregate interface is left unused. This change mitigates other issues related to destroying a link aggregate interface.

NFS

5095958	Addressed a defect that prevented netgroup, username, and username override file downloads over http or https from succeeding.
5221782	Changed ONCRPC timeouts for new Azure HPC Cache clusters to compensate for latency. In general, the cache-to-cache protocol timeout is 220 seconds, and the NFS protocol timeout to NAS core filers is 100 seconds. For new Azure HPC Cache systems, a connection multiplier of 4 is now applied to the NFS protocol for NAS storage targets.
5411717	Updated the XML-RPC method `corefiler.modify` so that it now can be used to change NAS core filer IP addresses without changing DNS names. Before this change, the `corefiler.modify` method would not start the transition if the old and new DNS name were the same.

5416783 An alert was added to detect address changes in a NAS core filer's DNS entry. The alert is raised if the addresses returned do not match the addresses in use.

The operating system does not automatically switch to using new addresses. To start using the new addresses, use `corefiler.modify` or follow the instructions in the alert. An administrator also can resolve the alert by restoring the previous addresses.

5865648 Changed procedures to prevent new filesystem operations or automatic retries from being sent to a node if the cluster has determined that the node is down. In normal operation, filesystem operations have a default retry period of 50 seconds regardless of node state.

5865680 Re-enabled connectivity alerts for ONC-RPC sessions. If the destination node is known to be down for an intra-cluster session, the connectivity alert is not raised. This change also fixed an internal race condition that could cause unnecessary updates when there were many sessions.

5865686 Fixed a problem in which NAS core filer connectivity problems could cause nodes to enter a pending state. If the connectivity problems alternated between nodes, it was possible for the client-facing addresses to fail over to healthy nodes.

These checks are related to the deprecated 2-node HA feature. If the 2-node HA feature is enabled, then this work item is disabled. If you use this feature, contact Microsoft Service and Support for help moving away from the deprecated configuration.

6008216 Corrected a problem that could cause communication issues with a cluster node. The root cause was a defect that allowed an IP address from an NFS client performing NLM lock operations to be reassigned as a cluster address.

6056374 Replaced an internal utility used to download netgroup, username, and username map URIs after a FreeBSD upgrade caused it to stop working. The new utility supports both http and https.

6176065 Addressed a defect where a kernel network buffer allocation failure was handled incorrectly for cache-to-cache operations. The defect resulted in a file system service restart with associated core file.

6200334 Fixed a problem where a core filer removal could cause the File System Service process to restart with an associated core file. The defect was triggered by a mount protocol response from the core filer that caused a retry while the core filer was being removed.

6213403 An alternative workflow has been added for NAS core filer exports. In this optional workflow, the "showmount" remote procedure call for the mount protocol is not sent to NAS core filers. Instead, the cluster is configured with the specific exports to access using the XML-RPC method `corefiler.setMountExports`. The cluster periodically probes only the exports that are specified.

To use this workflow, specify the `useSetMountExports` option when creating the core filer with the XML-RPC method `corefiler.create`. Work is in progress to add this option to other configuration interfaces.

Note: Two related changes are being withdrawn:

- A previous approach to this solution, work item 5153606, introduced new XML-RPC methods named `corefiler.setExportFilters` and `corefiler.getExportFilters()`. These methods have been deprecated.
- A related change, 7244835, gave the XML-RPC method `corefiler.setMountExports` to add multiple exports at once, but this change will be reverted with the next iteration of this feature.

6213517 Fixed an error that caused the XML-RPC method `nfs.listExports` to repeat export discovery even when configured to not retry.

6214221 A change was made to allow NAS core filer exports to use root squash under specific circumstances. Before this change, root squash had to be disabled on NAS core filers for access from both the cluster and vserver IP addresses.

Now, you can configure the core filer export to use root squash (with the user ID set to nobody) if your cache meets these criteria:

- The local directories setting is disabled.
- Cluster and vserver addresses must be included in the core filer export rule.
- Root squash must be configured the same way in the front-end export rule (which controls client access through the cache) and the core filer export rule.

Root squash is disabled in the default client access policy (export policy) for new vservers. Edit these policies to change the value.

6529372 Decreased the execution time for the XML-RPC methods `nfs.modify`, `vserver.addJunction`, `vserver.modifyJunction`, and `vserver.removeJunction`.

6529373 Reduced the amount of time needed to create a NAS core filer. This change modified the XML- RPC `corefiler.create` method so that it resolves NAS core filer DNS addresses 20 to 30 seconds sooner.

6529374 Improved the amount of time it takes to show exports for a newly created core filer that does not use the local directories feature in its cache policy.

This change affects the amount of time between the start of the XML-RPC method `corefiler.create` and the completion of to the `nfs.listExports`.

6529386 Namespace changes now complete more quickly. Specifically, the XML-RPC methods `vserver.addJunction`, `vserver.modifyJunction`, and `vserver.removeJunction` were updated to take less time.

7087032 Fixed a filesystem process restart and core file due to a memory allocation error that caused a failed ONCRPC portmap lookup.

7260838 Lowered the default limits for portmap lookup threads in cloud-based clusters. In hardware clusters, one thread is created per portmap lookup, and the default limit on the number of these threads is 500. Azure HPC Cache systems and Avere vFXT for Azure clusters created with this software have a default limit of 100 portmap lookup threads.

SMB

5132383	Fixed an issue related to failed SMB2 signing that could cause an SMB filesystem core file if a client issued a large number of SMB tree requests.
5132513	Addressed an error related to SMB access through a path involving SMB symbolic links. In some situations where a symbolic link could not be evaluated, the wrong error code was returned (NT_STATUS_OBJECT_NAME_NOT_FOUND or NT_STATUS_OBJECT_PATH_NOT_FOUND were sometimes returned instead of NT_STATUS_ACCESS_DENIED).
5923685	Updated code to prevent SMB guest access from being enabled by automatic fallback settings after internal failures. If SMB guest access is not disabled, the cluster can have SMB filesystem process restarts, with associated core files.
5996748	Fixed a memory leak that could occur when an SMB share disconnected and could lead to a core file.
5997035	Removed a defect that could cause the SMB logon service process to enter a bad state after an unrelated core file was generated in a different SMB logon service process. This situation could cause symptoms that included high CPU utilization and excessive log messages (which could lead to system instability if the log partition became full). After this change, the remaining SMB logon service process restarts gracefully.
6111684	Fixed two issues related to SMB share names: • Addressed a defect where share ACLs could persist after a share was deleted and then recreated with the same name. • Addressed a defect where any SMB XML-RPC method that resolved an AD SID to a name would fail due to a string processing difference between Python 2 and 3.
6214156	Updated the help file for the XML-RPC method `corefiler.create` to indicate that NetApp Clustered mode core filers require the filerClass attribute to be set only if SMB ACL junctions exist.
6447052	Fixed a defect where an SMB2 compound request could result in a restart and core file from the SMB filesystem service.
7227579	Fixed a defect that prevented the SMB process from restarting after certain types of memory allocation failures. In general, the operating system's SMB/CIFS process is configured to restart and write a core file if there are memory allocation failures.

Contact Microsoft Customer Service and Support

Microsoft Customer Service and Support can be reached by website, phone, or email.

Website: Use the links under **Support Information** on https://www.microsoft.com/en-us/avere/contact-us

Phone: 1-888-88-AVERE, Option 2 (Toll-Free)
1-412-894-2570, Option 2

Email: averesupport@microsoft.com