Release Notes: IBM Aspera High-Speed Transfer Server, High-Speed Transfer Endpoint, and Desktop Client, 4.4.1

Product Released: September 8, 2022
Release Notes Updated: September 8, 2022

This release of IBM Aspera High-Speed Transfer Server (HSTS), High-Speed Transfer Endpoint (HSTE), and Desktop Client provides the new features, fixes, and other changes listed below. In particular, the Breaking Changes section provides important information about modifications to the product that might require you to adjust your workflow, configuration, or usage. Additional sections cover system requirements and known problems.

NEW FEATURES

Support for exclude filters added to the `aclean` utility. (Core#119)

Ascp4 can now use a native library `libpvdr_cloud` for transfers to Object Storage without having to use `asperatrapd`. (Core#166)

Ascp4 supports the `--move-after-transfer` command line parameter. (Core#170)

The Node API `/asyncs/summary` response contains additional fields to have specific counters for files. (Core#97)

The Node API supports authenticating node users using JWT bearer tokens without requiring having a password for the node user. (Core#50)

The Node API `/ops/transfers` request payload accepts `preserve-uid` and `preserve-gid` options to use in the transfer. (Core#48)

Ascp using `stdio:/// sources` can transfer growing files (Core#309)

Ascp4 supports the `--precalculate-job-size` command line parameter. (Core#168)

Ascp4 supports the `--partial-file-suffix` command line parameter. (Core#169)

Changed the default value for the `pvcl_max_memory` configuration setting to 128 MB (Core#124)

Add the capability to use the asynchronous job queue to the Node API `/files/search` API endpoint. (Core#54)

New configuration parameter `async_inactivity_timeout_sec` for async transfers in continuous mode to control the idle timeout for reporting transfers as finished. (Core#182)

Watchfolder can synchronize an empty directory. (Core#26)

Transfers that use `stdio-tar` can preserve the timestamps of the source files. (Core#138)

BREAKING CHANGES

If you are upgrading from a previous release, the following changes in this release might require you to adjust your workflow, configuration, or usage.

Core#118 - When access keys are defined for a node user, requests to `/files/upload_setup` and `/files/download_setup` using node user credentials are denied.

Core#331 - The default minimum rate for jobs that are submitted through the async API is now 0.

Core#37 - Using the Node API `/files/upload_setup` and `/files/download_setup` require having the corresponding authorization option in `aspera.conf` set to token.

Core#22 - Enforce an upper limit of up to 99 threads for the `local-fs-threads` and `remote-fs-threads` parameters in async.

OTHER CHANGES

Core#246 - Improved performance in asperanoded when monitoring ongoing transfers.

Core#392 - The background task files_cleanup is disabled by default in asperanoded.

Core#484 - The internal async db schema version has been updated to version 21.

Core#220 - The file name for the Redis database on macOS X is different than in other platforms.

ISSUES FIXED IN THIS RELEASE

Core#295 - Async with continuous mode and kvstore activity logging enabled might crash when reporting activity if an error is returned from Redis.

Core#32 - Files with storage type Glacier Flexible Retrieval are not added to the list of files returned by Node API.

Core#16 - Using async with mirror mode enabled an in pull mode might result in POTENTIALCONFLICT state for files that are modified on the destination.

Core#318 - Transfers that use ascp might report its status as partially completed when using --symbolic-links copy.

Core#196 - Using the Node API /files/page endpoint with a count parameter of 1000 or higher against local storage might result in an error response.

Core#423 - Using ascp with the --no-read command line parameter results in a Disk read error.

Core#249 - A potential deadlock in ascp might prevent the process from terminating after the transfer is completed.

Core#360 - Transfers that use WebSockets fail to authenticate access keys.

Core#390 - The asperarund service stops automatically when the asperanoded service is stopped in Linux.

Core#394 - Async might be unable to restart the job if during the previous run a Disk write error occurs while transferring files.

Core#209 - The init.d scripts used in Linux versions that don't have systemd have %DAEMON_USER% and %DAEMON_GROUP% values instead of asperadaemon and aspadmins.

Core#189 - Ascp4 fails to start when using an S3 or Azure cloud storage source or destination.

SYSTEM REQUIREMENTS

Linux 64-bit: RHEL 7-8. CentOS 7-8. Ubuntu 20.04 LTS. Ubuntu 18.04 LTS. Ubuntu 16.04 LTS. SUSE Linux Enterprise Server (SLES) 12. Debian 8+. Fedora 19+. Kernel 3.10 or higher and Glibc 2.17+.

Windows: Windows Server (64-bit) 2012, 2016, and 2019. For client use only, you might also use Windows 10, 11 (64-bit).

macOS: 10.15 (Catalina), macOS 11.0 and 11.1 (Big Sur).

PowerLinux: RHEL 7-8. CentOS 7-8. Ubuntu 20.04 LTS. Ubuntu 18.04 LTS. Ubuntu 16.04.2 LTS. Your OS version must support little-endian (LE) ordering, and it must run on IBM Power hardware that supports LE ordering. Kernel: Linux 4.4.0-116-generic. Architecture: ppc64-le.

zLinux: Linux on z Systems s390, 64-bit. RHEL 7-8. SUSE Linux Enterprise Server (SLES) 12.

AIX: 7.1, 7.2, 7.3.

PRODUCT SUPPORT

For online support, go to the IBM Aspera Support site at https://www.ibm.com/mysupport/. To open a support case, log in with your IBMid or set up a new IBMid account.