

Qumulo On-Premises Administrator Guide



Copyright © 2024 Qumulo, Inc.

Table of Contents

Getting Started

Creating and Managing Directory Quotas	4
Supported Configurations and Known Limits.....	5

System Upgrades

Feature Log	6
Upgrade Modes.....	27
Instant Software and Platform Upgrades.....	31
Replication Version Requirements and Upgrade Recommendations.....	34

Authentication

Installing VPN Keys.....	35
Configuring SAML Single Sign-On (SSO)	38
Configuring Search Trusted Domains.....	39

Authorization

Managing Cross-Protocol Permissions.....	40
--	----

External Services

Using Access Tokens.....	41
Connecting a Kubernetes Cluster	42

Network Configuration

Required Networking Ports.....	43
--------------------------------	----

Network Multitenancy

Partitioning a Cluster into Tenants.....	44
Configuring Management Protocols	49
Configuring File System Protocols.....	52

Web UI

Setting the Web UI Login Banner	58
Setting the Web UI Inactivity Timeout.....	59

qq CLI

Enabling Autocomplete for the qq CLI	60
--	----

Metadata

Managing User-Defined Metadata.....	61
-------------------------------------	----

Snapshots

How Snapshots Work.....	62
Managing Snapshots.....	63
Locking and Unlocking Snapshots.....	64
Recovering Files by Using Snapshots	65

Encryption and Data Security

Managing Encryption at Rest.....	66
Generating and Storing ECDSA Keys.....	71
Managing Security Keys.....	72

Node Addition and Replacement

How Drive and Node Failure Protection Works	73
Adding Nodes to an Existing Cluster.....	79
Increasing the Node Fault Tolerance Level during Node-Add Operations.....	81
Adding Nodes with Adaptive Data Protection	82
Replacing Nodes by Performing a Transparent Platform Refresh.....	85
Improving Performance by Migrating a Cluster to a Different License Class.....	91
Understanding Offline Nodes and Checking for Free Space	92

Data Replication

Creating and Managing a Continuous Replication Relationship.....	94
Shift-To Amazon S3.....	95
Shift-From Amazon S3.....	96

File System Changes

How File System Change Notifications Work	97
Watching for Changes with SMB2 CHANGE_NOTIFY	98
Watching for Changes with REST.....	99

NFS

Creating and Managing an NFS Export	100
Enabling and Using NFSv4.1.....	101
Managing File Access Permissions with ACLs.....	102
Host Access Rules for NFS Exports	103

NFSv4.1 with Kerberos

How NFSv4.1 Works with Kerberos.....	104
Prerequisites for Joining to Active Directory.....	105
Configuring Active Directory	106
Performing Additional Cluster Configuration.....	107
Using Kerberos Permissions.....	108
Configuring a Linux Client.....	109
Configuring Cross-Domain Active Directory Trusts.....	110
Troubleshooting NFSv4.1 with Kerberos.....	111

SMB

Creating and Managing an SMB Share	112
--	-----

S3 API

Configuring and Using the S3 API	113
Creating and Managing S3 Access Keys	114
Creating and Managing S3 Buckets.....	115
Managing Access to S3 Buckets.....	116
Managing Access Policies for S3 Buckets	117
Managing Multipart S3 Uploads.....	118
Supported Functionality and Limits.....	119

Monitoring and Metrics

Enabling Cloud-Based Monitoring and Remote Support	120
Connecting to Cloud-Based Monitoring and S3 with a Custom Proxy.....	126
Restoring the Default Values for Cloud-Based and Nexus Monitoring	128
OpenMetrics API Specification.....	130

Getting Started

Creating and Managing Directory Quotas in Qumulo Core

This section explains how to create, modify, and delete directory quotas by using the Qumulo Core Web UI and how to use the Cluster Alerts for Qumulo script to manage cluster quota notifications.

Supported Configurations and Known Limits for Qumulo Core

This section provides an overview of supported configurations and known limits for Qumulo Core.

System Upgrades

Qumulo Core Feature Log

This section lists the most important features from each release.

For information about upgrade types for each release, see [Qumulo Core Upgrade Mode Reference \(page 27\)](#).

Qumulo Core 7.0.1.1

[Downloads and Release Notes](#)

Important

This release resolves a scenario in Qumulo Core 7.0.1 where a node in a Qumulo cluster could go offline if a tree delete operation was performed on a directory while an SMB client was watching it for changes.

- Configured rolling reboots to restart **N-1** nodes at a time, where **N** is the number of nodes configured for the cluster's fault tolerance
- Further improved NFSv3 random small-write performance
- Added support for Active Directory Global Catalog for lookups
- Added support for **DeleteBucketPolicy** and **PutBucketPolicy** S3 API actions
- Configured RBAC privileges to override bucket policies which deny a user access to API actions through the S3 API
- Improved REST API and **qq** CLI operations for working with S3 buckets
- Fixed a bug with custom LDAP schemas

Qumulo Core 7.0.0.1 (Quarterly)

[Downloads and Release Notes](#)

Important

To install Qumulo Core on HPE Alletra 4110 platforms, you must use the 7.0.0.1 release.

- Added support for NFSv4.1 cross-connection write combining
- Made significant improvements in SMB random small-write performance
- Added partial support for new S3 API actions
- Configured Qumulo Core to take daily snapshots at regular intervals throughout the day

- Made changes to the REST API and the `qq` CLI to accommodate improvements in snapshot and at-rest encryption functionality
- Revised permissions for SMB `copychunk` requests
- Resolved a minor issue with AD authentication, an issue with `.snapshot` directories in Windows Command Prompt, and an issue with SMB copy operations and long share names and volume labels

Qumulo Core 6.3.2

Downloads and Release Notes

- Added support for additional custom metadata operations to the S3 API
- Configured Qumulo Core to replicate user metadata
- Added support for counting metadata updates for a file
- Configured OpenMetrics API metrics for protocol operations to track file system REST API operations
- Fixed an issue with cluster authentication for Kerberos client principals with names that contain the `@` character

Qumulo Core 6.3.1.1

Downloads and Release Notes

Important

This release resolves an issue with memory utilization during secure credential handling.

- Added support for storing and retrieving custom metadata and tags by using the S3 API and Qumulo REST API and made supporting changes to the API and `qq` CLI
- Added the optional `data_revision` field for all REST resources that return file or stream attributes
- Resolved an issue with a third-party GSS authentication library

Qumulo Core 6.3.0.1 (Quarterly)

Downloads and Release Notes

Important

- This release resolves an issue with memory utilization during secure credential handling.
 - This release resolves an issue in Qumulo 6.2.1.1 that interrupted the functioning of cross-realm Kerberos authentication when the system returns CNAMEs for PTR records while performing reverse DNS lookups.
-
- Configured nodes to remain powered on after Transparent Platform Refresh operations
 - Changed the behavior of static IP addresses when a cluster's networking configuration changes or nodes are replaced
 - Improved the Web UI by making accessibility improvements, renaming headings on the **Dashboard** page, and added physical drive bay locations to the **Node Details** page for HPE Alletra 4110 and Quiver 1U All-NVMe Gen1 node types
 - Improved compatibility between Qumulo Core and the Auth0 SSO provider, increased the replication speed for large numbers of deleted files, updated the **smartpqi** driver for the HPE Apollo 4200 Gen10 and HPE Apollo 4200 Gen10 Plus node types, and resolved an issue with misreported temperatures for AMD CPUs

Qumulo Core 6.2.2.2

[Downloads and Release Notes](#)

Important

- This release resolves an issue with memory utilization during secure credential handling.
 - This release resolves an issue in Qumulo 6.2.1.1 that interrupted the functioning of cross-realm Kerberos authentication when the system returns CNAMEs for PTR records while performing reverse DNS lookups.
 - This release resolves an issue with caching NFS hostnames from the NFS allowed list.
-
- Added the ability to store master keys for at-rest encryption in an external Key Management Server (KMS) and updated the corresponding REST APIs and **qq** CLI commands
 - Added the ability to dismiss errors and warnings on the **Cluster > Active Directory** page in the Web UI by clicking **X**
 - Configured the drive LEDs for the front (external) storage bays on Quiver 1U All-NVMe Gen

1 platform to light up red if a drive fails

- Ensured that certain SMB `qq` CLI commands always output valid JSON

Qumulo Core 6.2.1.2

[Downloads and Release Notes](#)

Important

- This release resolves an issue in Qumulo 6.2.1.1 that interrupted the functioning of cross-realm Kerberos authentication when the system returns CNAMEs for PTR records while performing reverse DNS lookups.
 - This release resolves an issue in Qumulo Core 6.2.1 that affected the persistence of cluster network settings configured by using the Web UI.
 - This release removes the flag `--dns-config-id` that was added in Qumulo Core 6.2.1. This flag prevents the normal functioning of the `qq fs_list_locks` command.
 - This release resolves an issue that we have identified in Qumulo Core 6.1.0 (and higher), where a potential session impersonation exploit was possible when a client accesses a Qumulo cluster by using the SMB protocol.
- Configured Qumulo Core 6.0.0 (and higher) to have replication compatibility with all quarterly and non-quarterly releases up to 8 quarters in the future (previously, compatibility was for 2 quarters)
 - Improved the functionality of S3 `ListObjects`, `ListObjectsV2`, and `PutObject` API actions
 - Added support for the SMB `ATTR_OFFLINE` extended file attribute
 - Replaced deprecated REST API and `qq` CLI command pertaining to authentication
 - Deprecated and replaced REST API command pertaining to the UID light and PSU information

Qumulo Core 6.2.0.1 (Quarterly)

[Downloads and Release Notes](#)

Important

- The removal of the deprecated `/v1/smb/shares` REST API endpoints in this release can affect certain third-party backup or migration workflows.
- This release resolves an issue with Qumulo Core's ability to route return packets back to clients when the following conditions are true for a cluster:
 - The cluster has a single tenant
 - VLAN networks are configured
 - Packets are destined for specific VLANs without a configured gateway

After upgrading to Qumulo Core 6.2.0.1, you can continue to use the default gateway to route VLAN-specific packets.



- Added the ability to specify hostnames in host access rules for NFS exports
- Added the privilege `FILE_READ_ACCESS` to Qumulo Core for roles that require read-only access
- Improved the snapshot deletion process
- Added information about drive bays and disk serial numbers for Quiver 1U All-NVMe Gen1 and virtual machine nodes
- Disabled the automatic addition of DNS records to an AD domain when you join the domain

Qumulo Core 6.1.2.2

[Downloads and Release Notes](#)



Important

- This release resolves:
 - An issue that affects parsing Kerberos principal names that contain special characters in Active Directory Kerberos authentication in Qumulo Core 6.1.2. This issue doesn't prevent access to SMB shares by using NTLM-based authentication.
 - An issue that affects upgrading Qumulo Core on the Quiver 1U Hybrid Gen2 platform in Qumulo Core 6.1.1.
 - An issue that affects getting a useful core dump in Qumulo Core 6.1.1.
- We've identified a known issue that affects event processing in the Cloud-Based Monitoring (CBM) service Qumulo Core 6.1.0.3. This issue is resolved in Qumulo Core 6.2.0.1.

- Added Adaptive Data Protection to the REST API and  CLI and made performance improvements
- Added Transparent Platform Refresh to the REST API and  CLI
- Added S3 configuration settings to the Web UI
- Improved hard link creation

Qumulo Core 6.1.1

[Downloads and Release Notes](#)

- Added support for new S3 API actions, snapshot locking for replication target relationships, and lock configuration to the Web UI
- Enabled the Search Trusted Domains Active Directory configuration option by default
- Upgraded the host and container images
- Revised the REST API for snapshots
- Added  CLI commands for lock key functionality
- Added a  CLI command for viewing blocked upgrade status
- Updated the OpenMetrics API Specification
- Made general improvements to Qumulo Core

Qumulo Core 6.1.0.3 (Quarterly)

[Downloads and Release Notes](#)

⚠ Important

This release resolves an issue where, under certain circumstances, while using Adaptive Data Protection in Qumulo Core 6.1.0 (and lower), it becomes impossible to add nodes to a cluster during normal cluster expansion or Transparent Platform Refresh operations, until you upgrade the cluster to version 6.1.0.3 (or higher).

- Configured Qumulo Core to use recursive change notifications by default
- Enabled SMB multichannel by default
- Added snapshot locking functionality by using the `qq` CLI
- Made usability changes to event logging and the Web UI
- Removed beta multitenancy REST APIs and `qq` CLI commands

Qumulo Core 6.0.2

[Downloads and Release Notes](#)

⚠ Important

The removal of the deprecated `/v1/smb/shares` REST API endpoints in this release can affect certain third-party backup or migration workflows.

- Added support for streaming file change notifications
- Improved the S3 API
- Made improvements to the Qumulo Core allocation algorithm
- Added audit logging and REST API and `qq` CLI commands for streaming file change notifications
- Removed v2 Python bindings for NFS export and SMB share APIs
- Added new fields to the OpenMetrics API
- Fixed issues with creating SMB shares, transparent platform refresh, vSphere clusters, upgrades in the Web UI, making SMB2 copies by using ODX, and creating snapshot policies

Qumulo Core 6.0.1.1

[Downloads and Release Notes](#)

- Added new SMB2 `CHANGE_NOTIFY` functionality
 - Full support for recursive watch-tree notifications

- Alternate Data Streams
- Numerous performance improvements
- Elimination of missed events
- Added read combining for NFSv4.1
- Improved the S3 API
 - S3 API action traversal rights
 - S3 traffic in the Web UI and in **qq** CLI output
 - Presigned URL functionality
 - HTTP identity transfer encoding for the **GetObject** API action
- Added operational information to audit logging in JSON format
- Made numerous updates to the multitenancy and S3 **qq** CLI commands and REST APIs
- Fixed an issue with Network Interface Card (NIC) addressing on 240TB and 480TB General Purpose models of the HPE Apollo 4200 Gen10 Plus node

Qumulo Core 6.0.0.2 (Quarterly)

Downloads and Release Notes

Important

- This release resolves an issue with transparent platform refresh that we discovered in Qumulo Core 6.0.0 and 6.0.0.1.
- This release restores the status of the `/v1/smb/shares` REST API endpoints (previously removed in Qumulo Core 5.3.4) to deprecated. We strongly recommend changing over to `/v2/smb/shares` endpoints as soon as possible.
- Added Transparent Platform Refresh for node replacement
- Enabled concurrent node reboots for rolling upgrades
- Added write combining for NFSv4.1
- Introduced optional HTTP mode for the S3 API
- Made SSL certificate changes for the S3 API without a cluster restart
- Added audit syslog in JSON format
- Made corresponding changes and additions to the REST API and **qq** CLI
- Fixed issues with the Web UI, directory access by non-root users on Linux, and changed

caching for some OpenMetrics API metrics

Qumulo Core 5.3.4

[Downloads and Release Notes](#)

Important

The removal of the deprecated `/v1/smb/shares` REST API endpoints in this release can affect certain third-party backup or migration workflows.

- Added data protection configuration, multitenancy, tenant-aware SMB and NFS, and automatic clean-up of stale multipart S3 uploads
- Removed all REST API endpoints under `/v1/smb/shares`
- Improved and added REST APIs and `qq` CLI commands for restriper, multitenancy, and tenant-aware SMB and NFS operations
- Improved the OpenMetrics APIs
- Fixed an issue with SMB2 attributes for symlinks and a potential security issue with REST API session tokens

Qumulo Core 5.3.3.1

[Downloads and Release Notes](#)

Note

This release resolves an issue that we discovered in Qumulo Core 5.3.3.

- Put out the initial release of the S3 Protocol Server
- Added support for NFSv4.1 with KRB5i
- Added the ability to disable access tokens
- Removed `/v1/session/kv` REST API endpoints
- Fixed a bug with listing trustees for a role and a bug with LDAP connections

Qumulo Core 5.3.2

[Downloads and Release Notes](#)

- Added distributed multi-AZ clustering to Qumulo clusters on AWS
- Added SMB2 `CHANGE_NOTIFY` parity with Windows
- New Qumulo OpenMetrics API metrics

- Added expiration time for access tokens
- Improved the REST API

Qumulo Core 5.3.1

Downloads and Release Notes

- Added support for watching for changes in file attributes and directory entries with SMB2 `CHANGE_NOTIFY` filters
- Updated all firmware for all variants of Mellanox CX5 and CX6 and Broadcom NICs
- Added metrics to the Qumulo OpenMetrics API
- Added two-factor authentication for qq-CLI-initiated SAML SSO workflows
- Improved REST API and `qq` CLI
- Added a missing library dependency that allows SAML SSO to work on cloud platforms

Qumulo Core 5.3.0 (Quarterly)

Downloads and Release Notes

- Added interactive SAML SSO login and SSO for Active Directory users who manage clusters
- Identified a known issue with SAML SSO on cloud platforms
- Improved SMB `CHANGE_NOTIFY` requests
- Released the Qumulo OpenMetrics API
- Released the Access Tokens API
- Improved the Web UI
- Added REST APIs for node identification
- Improved `qq` CLI commands

Qumulo Core 5.2.5.1

Downloads and Release Notes

Note

This release resolves an issue that we discovered in Qumulo Core 5.2.5.

- Allowed clusters joined to Active Directory to use SAML 2.0 SSO
- Added support for displaying NFSv4.1 quotas

- Created a more balanced usage of free space on clusters after the addition of nodes
- Improved the Web UI
- Added a flag to a `qq` CLI command
- Fixed bugs related to quotas and Alternative Data Streams

Qumulo Core 5.2.4

[Downloads and Release Notes](#)

The majority of functionality and stability improvements in this Qumulo Core release are internal.

Qumulo Core 5.2.3

[Downloads and Release Notes](#)

The majority of functionality and stability improvements in this Qumulo Core release are internal.

Qumulo Core 5.2.2

[Downloads and Release Notes](#)

- Made improvements to the Web UI on the NFS Exports and Role Management pages
- Adjusted the REST API endpoint `/v1/smb/sessions` and the corresponding `qq` CLI command
- Fixed an issue that occurs during NFSv4.1 failover
- Fixed an issue that prevents a cluster from forming due to an incorrect HDD-SSD ratio.

Qumulo Core 5.2.1

[Downloads and Release Notes](#)

- Added over-the-wire encryption to NFSv4.1 with Kerberos
- Improved performance for SMB Find requests
- Added an optional Web UI login banner
- Removed `/v1/upgrade/` endpoints
- Fixed a bug on the NFS Exports page.

Qumulo Core 5.2.0.2 (Quarterly)

[Downloads and Release Notes](#)

- Enabled local trustees for NFSv4.1 with Kerberos
- Added support for Amazon EC2 M6i instances and Amazon EBS gp3 volumes

- Removed and replaced REST API endpoints related to shutting down and stopping clusters and nodes
- Fixed an SMB client bug
- Resolved a file system operations latency issue with continuous replication jobs and on clusters with high file creation load
- Resolved an issue that caused a node to enter a reboot loop under certain conditions.

Qumulo Core 5.1.5

Downloads and Release Notes

- Enabled creating mounts by using NFSv4.1 with Kerberos
- Improved NLM and NFSv4.1 locking
- Updated Mellanox NIC firmware
- Made improvements to the Web UI on the APIs & Tools, Add Nodes, and Active Directory pages
- Added NFS fields to the REST API and `qq` CLI
- Deprecated the `/v2/upgrade/status` REST API
- Updated security permissions for home directories
- Prohibited the use of the `switch-user` command
- Added support for user principal (UPN) names

Qumulo Core 5.1.4.1

Downloads and Release Notes

- Improved workload read performance for streaming
- Deprecated the `/v1/session/kv/` REST API endpoints
- Fixed an issue on the FTP Settings page
- Fixed a bug where unexpected behavior from an NFS client can cause issues with memory usage on a Qumulo cluster.

Qumulo Core 5.1.3

Downloads and Release Notes

- Added NFSv4.1 byte-range locking support for clusters that have NFSv4.1 enabled
- Added custom naming functionality for snapshots

- Made improvements to the Web UI
- Improved node fault tolerance for node-adds
- Updated the time synchronization for cloud clusters
- Added a REST API to calculate node-add capacity
- Fixed bugs about adding S3 privileges, making **OPEN** requests over NFSv4.1, and reassigning floating IP addresses.

Qumulo Core 5.1.2

Downloads and Release Notes

- Improved user experience for NFSv4.1 ACL editor
- Made improvements to the Web UI
- Updated time synchronization APIs for QaaS Azure clusters
- Fixed bugs with malformed **RPCBIND** requests and replication relationships.

Qumulo Core 5.1.1

Downloads and Release Notes

- Improved time management on QaaS Azure clusters
- Improved resiliency against data center accidents
- Made improvements to the Web UI
- Updated the SDK to Python 3.8
- Disabled time management APIs on QaaS Azure clusters
- Fixed SMB and Web UI form bugs.

Qumulo Core 5.1.0.1 (Quarterly)

Downloads and Release Notes

- Improved write performance for workloads on clusters with high-churn, small-file datasets
- Added a new Active Directory capability
- Made a number of pages in the Web UI more clear
- Improved SMB and NFS share, export management, and status APIs
- Added inactivity timeout to the Web UI
- Identified a regression in 5.0.5 and fixed a bug that occurs for rare NFS client applications.

Qumulo Core 5.0.6

Downloads and Release Notes

- Improved read performance for previously read large files
- Improved write performance for certain platforms and workloads
- Improved cluster resilience on Azure
- Added support for HPE Broadcom NICs and updated the NIC and SKU reference
- Made improvements to the Web UI
- Fixed bugs pertaining to snapshots of directories with SMB-illegal characters and localized Active Directory installations.

Qumulo Core 5.0.5

Downloads and Release Notes

- Increased the tolerance of uncertainty for NTP
- Mapped Isilon-style and Samba-style NFS SIDs to Qumulo SIDs
- Made changes to the Python SDK
- Made an XSS security fix
- Resolved two bugs

Qumulo Core 5.0.4

Downloads and Release Notes

- Enabled advanced data integrity for new clusters
- Added a check to confirm whether an upgrade can begin
- Added an initial release of the S3 protocol
- Deprecated reboot `qq` CLI commands
- Added access time (`atime`) API resources and `qq` CLI commands.

Qumulo Core 5.0.3

Downloads and Release Notes

- Added Rolling Reboot for platform upgrades (including new APIs and CLI commands)
- Updated the Web UI for both software and platform upgrades
- Made improvements to write performance over NFSv3
- Improved the **Add Nodes** and **Snapshots** pages in the Web UI

Qumulo Core 5.0.2

Downloads and Release Notes

- Changed all HPE firmware updates to be performed using the Qumulo installer
- Improved cluster resilience against boot drive failures
- Added the ability to set timestamps through NFSv4.1
- Added the `access_time` field to the attributes REST API resource.

Qumulo Core 5.0.1

Downloads and Release Notes

- Enforced the Qumulo-to-Qumulo replication compatibility guarantee to 3 consecutive quarterly releases

✓ Tip

For example, 5.0.1 can't replicate with versions lower than 4.3.0 or higher than 5.2.0.

- Made improvements to Shift-to-S3
- Removed unconfigured node APIs

Qumulo Core 5.0.0.1 (Quarterly)

Downloads and Release Notes

- Changed the Qumulo-to-Qumulo replication compatibility guarantee from 2 to 3 consecutive quarterly releases
- Enforced Kerberos ticket expiration
- Enabled NLM persistence by default
- Added sorting and filtering to the Snapshot Policies page
- Improved speed of deleted snapshot clean-up
- Removed requirements for QaaS clusters
- Added support for Python 3.8
- Improved the API and CLI version and user-querying command
- Resolved an issue that causes an unpredictable node failure if two NFSv4.1 clients have the same hostname.

Qumulo Core 4.3.4

[Downloads and Release Notes](#)

⚠ Important

This release of Qumulo Core resolves a critical security vulnerability. We strongly recommend upgrading your Qumulo cluster to 4.3.4 to resolve this issue.

- Added work estimation to Shift-from-S3 relationships
- Added a feature that improves cluster resiliency for unresponsive devices
- Changed the kiosk timeout window to 15 minutes

Qumulo Core 4.3.3

[Downloads and Release Notes](#)

- Prevented authentication requests with PAC buffers of unknown type from failing
- Improved Saved Snapshots page in the Web UI
- Added a mechanism for monitoring status of front-end network on platforms with separate interfaces.

Qumulo Core 4.3.2

[Downloads and Release Notes](#)

Fixed a bug in Qumulo Core's handling of cross-protocol permissions

Qumulo Core 4.3.1

[Downloads and Release Notes](#)

- Improved write performance
- Improved speed and duration of terminating outstanding requests to AD Controllers
- Added file system permission checks for SMB shares and NFS exports
- Enabled mounting NFS over UDP with floating IP addresses

Qumulo Core 4.3.0 (Quarterly)

[Downloads and Release Notes](#)

- Added Network File System Version 4.1 (NFSv4.1)
- Improved performance for SMB offloaded data transfer (ODX) and SMB directory query
- Allowed AWS Sidecar to handle a new class of EBS failures

- Fixed bugs in Shift-From-S3 and the Web UI

Qumulo Core 4.2.6

[Downloads and Release Notes](#)

- Made an enhancement to Web UI for Qumulo on Azure
- Adjusted a REST resource
- Made an important change to export names that end with a slash (/) or are longer than 64 KiB

Qumulo Core 4.2.5

[Downloads and Release Notes](#)

Added Web UI management support for Qumulo Shift-from-S3 operations

Qumulo Core 4.2.4

[Downloads and Release Notes](#)

- Enhanced tree-delete operations
- Made performance improvement for C-72T, C-168T, C-192T, and C-432T platforms
- Enhanced Shift-to-S3
- Made performance improvements
- Made several important bug fixes

Qumulo Core 4.2.3

[Downloads and Release Notes](#)

- Added Qumulo Shift-from-S3
- Web UI Update for SMB Shares
- Enabled DNS configuration for cloud deployments in the Web UI

Qumulo Core 4.2.2

[Downloads and Release Notes](#)

- Improved the appearance of spinners in the Web UI
- Made a failover procedure bug fix

Qumulo Core 4.2.1

[Downloads and Release Notes](#)

- Made improvements for security identifiers

Important

We removed this feature for compatibility reasons. For more information, see [the release notes](#) on Qumulo Nexus.

- Upgraded the Linux kernel to 5.4
- Improved multistream throughput for 4U nodes
- Updated **qq** CLI commands for unconfigured node cluster creation
- Fixed a bug

Qumulo Core 4.2.0 (Quarterly)

[Downloads and Release Notes](#)

- Enhanced snapshot performance
- Enhanced Active Directory connections
- Made an important bug fix

Qumulo Core 4.1.5

[Downloads and Release Notes](#)

- Added details to the **Node Add** page in the Web UI
- Updated **qq** CLI upgrade commands for unconfigured nodes
- Made improvements to snapshot cleanup
- Made an important bug fix

Qumulo Core 4.1.4

[Downloads and Release Notes](#)

- Updated user privileges for the quota **qq** CLI commands
- Enhanced autocomplete functionality in the Web UI
- Made several important bug fixes

Qumulo Core 4.1.3

[Downloads and Release Notes](#)

The majority of functionality and stability improvements in this Qumulo Core release are internal.

Qumulo Core 4.1.2

[Downloads and Release Notes](#)

- Enhanced security of **qq** CLI commands
- Made several important bug fixes

Qumulo Core 4.1.1

[Downloads and Release Notes](#)

- Added the ability to repeat completed jobs for Shift
- Enhanced the security flag for SMB connections
- Improved directory-caching performance
- Updated the HPE Field Verification Tool

Qumulo Core 4.1.0.1 (Quarterly)

[Downloads and Release Notes](#)

- Enhanced the upgrade experience
- Improved the Qumulo Sidecar for AWS
- Made several important bug fixes

Qumulo Core 4.0.6

[Downloads and Release Notes](#)

Made a platform upgrade for cloud clusters

Qumulo Core 4.0.5

[Downloads and Release Notes](#)

- Updated the SMB API and CLI
- Made several important bug fixes

Qumulo Core 4.0.4

[Downloads and Release Notes](#)

- Updated the firmware for C-72T, C-168T, C-192T, C-432T, K-144T, K-168T, and K-432T platforms
- Enhanced **qq** CLI commands
- Fixed an important bug for mixed-node clusters

Qumulo Core 4.0.3

[Downloads and Release Notes](#)

- Enhanced the web server security
- Fixed an important bug fix for mixed-node clusters

Qumulo Core 4.0.2


[Downloads and Release Notes](#)

- Improved the performance of SSD clusters
- Made minor performance enhancements
- Fixed an important bug in the Web UI

Qumulo Core 4.0.1.1

[Downloads and Release Notes](#)

Fixed important bugs in:

- Cloud-Based Monitoring
- SMB file enumeration
- Compound API and  CLI request errors

Qumulo Core 4.0.0.2 (Quarterly)

[Downloads and Release Notes](#)

- Deprecated the Python 2.7 SDK
- Made several important bug fixes

Qumulo Core 3.3.5

[Downloads and Release Notes](#)

- Updated mixed-node compatibility
- Made a performance improvement for Qumulo Shift
- Improved the Field Verification Tool for HPE Apollo 4200 Gen10 clusters
- Adjusted audit logging functionality
- Made a few important bug fixes

Qumulo Core 3.3.4

[Downloads and Release Notes](#)

- Added support for the HPE ProLiant DL325 Gen10 Plus platform
- Made general performance enhancements for hybrid clusters

Qumulo Core 3.3.3

[Downloads and Release Notes](#)

- Added Instant Software Upgrade
- Made Qumulo Shift performance improvements and Web UI enhancements
- Updated the HPE Service Pack Pro (SPP) Update for HPE Apollo 4200 Gen10 clusters

Qumulo Core Upgrade Mode Reference

This section provides a reference for Qumulo Core upgrade modes from version 3.3.3 onwards.

- An [instant software upgrade \(page 31\)](#) requires restarting only the container on your nodes and has a downtime of less than 30 seconds without disruption to the operation of the cluster.
- A [platform upgrade \(page 32\)](#) requires either a complete reboot (rebooting all nodes in your cluster at the same time) or a rolling reboot (rebooting the nodes in your cluster one at a time).
- A [quarterly upgrade](#) aggregates all improvements and fixes since the last quarterly upgrade. The version number of a quarterly upgrade ends in **.0**.

⚠ Important

Although the *upgrade types* for on-premises upgrades and cloud upgrades are most often the same, they do occasionally diverge. For example, for Qumulo Core 5.3.1, a cloud deployment allows an instant upgrades, an on-premises deployment requires a platform upgrade.

For downloads, release notes, and upgrade paths for on-premises and cloud releases of Qumulo Core, see [Qumulo Nexus](#) 🔒.

Qumulo Core Upgrade Modes

For information about the most important features from each release, click the Qumulo Core version.

Version	On-Premises Upgrade Type	Cloud Upgrade Type
7.0.1.1 (page 6)		Instant
7.0.0.1 (Quarterly) (page 6)		Instant
6.3.2 (page 7)		Instant
6.3.1.1 (page 7)		Instant
6.3.0.1 (Quarterly) (page 7)		Platform
6.2.2.2 (page 8)		Instant
6.2.1.2 (page 9)		Instant

Version	On-Premises Upgrade Type	Cloud Upgrade Type
6.2.0.1 (Quarterly) (page 9)	Instant	
6.1.2.2 (page 10)	Platform	
6.1.1 (page 11)	Platform	
6.1.0.3 (Quarterly) (page 11)	Instant	
6.0.2 (page 12)	Instant	
6.0.1.1 (page 12)	Platform	
6.0.0.2 (Quarterly) (page 13)	Instant	
5.3.4 (page 14)	Instant	
5.3.3.1 (page 14)	Instant	
5.3.2 (page 14)	Instant	
5.3.1 (page 15)	Platform	Instant
5.3.0 (Quarterly) (page 15)	Instant	
5.2.5.1 (page 15)	Instant	
5.2.4 (page 16)	Instant	
5.2.3 (page 16)	Instant	
5.2.2 (page 16)	Instant	
5.2.1 (page 16)	Instant	
5.2.0.2 (Quarterly) (page 16)	Instant	
5.1.5 (page 17)	Platform	
5.1.4.1 (page 17)	Instant	
5.1.3 (page 17)	Instant	
5.1.2 (page 18)	Instant	
5.1.1 (page 18)	Platform	

Version	On-Premises Upgrade Type	Cloud Upgrade Type
5.1.0.1 (Quarterly) (page 18)		Instant
5.0.6 (page 19)		Instant
5.0.5 (page 19)		Instant
5.0.4 (page 19)		Instant
5.0.3 (page 19)		Instant
5.0.2 (page 20)		Instant
5.0.1 (page 20)		Instant
5.0.0.1 (Quarterly) (page 20)		Instant
4.3.4 (page 21)		Instant
4.3.3 (page 21)		Instant
4.3.2 (page 21)		Instant
4.3.1 (page 21)		Instant
4.3.0 (Quarterly) (page 21)		Instant
4.2.6 (page 22)		Instant
4.2.5 (page 22)		Instant
4.2.4 (page 22)		Platform
4.2.3 (page 22)		Instant
4.2.2 (page 22)		Instant
4.2.1 (page 22)		Platform
4.2.0 (Quarterly) (page 23)		Instant
4.1.5 (page 23)		Instant
4.1.4 (page 23)		Instant
4.1.3 (page 23)		Instant

Version	On-Premises Upgrade Type	Cloud Upgrade Type
4.1.2 (page 24)	Instant	
4.1.1 (page 24)	Instant	
4.1.0.1 (Quarterly) (page 24)	Instant	
4.0.6 (page 24)	Instant	Platform
4.0.5 (page 24)	Instant	
4.0.4 (page 24)	Instant	
4.0.3 (page 25)	Instant	
4.0.2 (page 25)	Instant	
4.0.1.1 (page 25)	Instant	
4.0.0.2 (Quarterly) (page 25)	Instant	
3.3.5 (page 25)	Instant	
3.3.4 (page 25)	Instant	
3.3.3 (page 26)	Instant	

Performing Qumulo Core Instant Software Upgrades and Platform Upgrades

This section explains the difference between Qumulo Core Instant Software Upgrades and Platform Upgrades. For more information, see [Performing Qumulo Core Upgrades by Using the qq CLI](#) on Qumulo Care.

⚠ Important

- If you perform multiple upgrades back to back, you might encounter one or more platform upgrades in one of the incremental releases; you must install these upgrades before you continue. Before performing back to back upgrades, [contact the Qumulo Care team](#) for guidance.
- If you don't see a rolling reboot option for a platform upgrade, refresh the page in your browser.

For downloads, release notes, and upgrade paths for on-premises and cloud releases of Qumulo Core, see [Qumulo Nexus](#) 🔒.

Understanding the Differences Between Upgrade Modes

For information about which upgrade modes different Qumulo Core releases use, see [Qumulo Core Upgrade Mode Reference \(page 27\)](#).

Instant Software Upgrade

The more common, faster instant software upgrade requires restarting only the container on your nodes and has a downtime of less than 30 seconds without disruption to the operation of the cluster.

Because in Qumulo Core 3.3.2 (and higher), the Qumulo file and data protection systems are separate from the host in charge of running the operating system and the services specific to each hardware or cloud platform, and because these services run in a lightweight container (by using Ubuntu-native [systemd-nspawn](#) containerization) in the user space, it is possible to move quickly from one version of Qumulo Core to another by loading a new container and pointing the runtime environment at updated software.

Note

- Under certain conditions, an end-to-end instant software upgrade might take a little longer while Qumulo Core performs background tasks. This doesn't impact user experience.
- Instant software upgrades don't impact existing support for the qq CLI or REST API commands.
- A direct upgrade to Qumulo Core 3.3.3 *isn't* an instant software upgrade (it only establishes the framework for this functionality). Upgrading from Qumulo Core 3.3.2 to 3.3.3 is the first official, minimally disruptive instant software upgrade. Any subsequent upgrade, regardless of release, is an instant software upgrade unless we specify otherwise.

Platform Upgrade

The infrequent, somewhat slower platform upgrade requires either a *complete reboot* (rebooting all nodes in your cluster at the same time with SMB and NFS client interruption) or—in Qumulo Core 5.0.3 (and higher)—a *rolling reboot* (rebooting the nodes in your cluster one at a time with impact only to SMB clients).

The reboot process differentiates platform upgrades from instant software upgrades, where your nodes maintain the Linux operating system (and certain services that Qumulo Core relies on) during the upgrade process and where the underlying host changes less frequently than the container and the file and data protection systems.

Important

- If you perform multiple upgrades back to back, you might encounter one or more platform upgrades in one of the incremental releases; you must install these upgrades before you continue. Before performing back to back upgrades, [contact the Qumulo Care team](#) for guidance.
- If you don't see a rolling reboot option for a platform upgrade, refresh the page in your browser.
- Upgrading past a platform upgrade still requires a node reboot, even if you don't install the exact build. For example, if your cloud cluster runs Qumulo Core 4.0.0, installing Qumulo Core 4.1.0 triggers a platform upgrade, because the installation includes all changes contained in Qumulo Core 4.0.6 that is a platform upgrade.

Understanding the Upgrade Phases

Every Qumulo Core upgrade has two phases, *preparation* and *commit*.

1. **Preparation:** Qumulo Core stages a new image in an alternate boot drive partition while the current image continues to run. This phase is responsible only for the background work (unpacking and writing the platform image and upgrade firmware, and so on). When the preparation phase is complete, we continue to the commit phase.
2. **Commit:** Qumulo Core does one of the following:
 - **Instant Software Upgrade:** Stops the existing container and starts a new one.
 - **Platform Upgrade:** Initiates a reboot and selectively upgrades the operating system image.

To determine what phase an upgrade is in, use the `qq upgrade_status` command while your cluster is performing an upgrade. For more information, see [Performing Qumulo Core Upgrades by Using the qq CLI](#) on Qumulo Care.

The following is example output from the command.

```
{
  "install_path": "/upgrade/qinstall.qimg",
  "state": "UPGRADE_PREPARING",
  "details": "",
  "error_message": "",
  "error_state": "UPGRADE_ERROR_NO_ERROR",
  "is_blocked": false,
  "blocked_reason": ""
}
```

Replication Version Requirements and Upgrade Recommendations for Qumulo Core

This section explains the relationship between the version of Qumulo Core that a cluster runs and data replication between it and other clusters. It also gives recommendations for upgrading Qumulo Core in relation to data replication tasks.

Authentication

Installing VPN Keys on a Qumulo Cluster

This section explains how to install VPN keys obtained from the Qumulo Care team on your Qumulo cluster, over a network. You can install the VPN keys by using the `qq` CLI from a machine on the same network as your cluster or from one of your nodes.

! Caution

Follow these steps only if a member of the Qumulo Care team instructs you to do so. Performing these steps incorrectly can cause network performance, connectivity, and data integrity issues. It can also expose your cluster to unauthorized access. For help with this task, [contact the Qumulo Care team](#).

Prerequisites

Before you begin, make sure that you have done the following.

- Obtain a `.zip` file with VPN keys from Qumulo Care
- Whitelist the following domains in your firewall rules:
 - `ep.qumulo.com`
 - `api.missionq.qumulo.com`
 - `monitor.qumulo.com`
 - `api.nexus.qumulo.com`
- Permit outbound HTTPS traffic on port 443

i Note

If your firewall performs stateful packet inspection (also known as *SPI* or *deep-packet inspection*), you must allow OpenVPN (SSL VPN) explicitly, rather than only open port 443.

To Install VPN Keys from a Networked Machine

1. Copy the `.zip` file from Qumulo Care to a computer on the same network as your cluster, and decompress the file.
2. Install the `qq` CLI on the same computer. For more information, see [QQ CLI: Get Started](#) on Qumulo Care.
3. To log in to your cluster, use the `qq` CLI and specify your cluster's IP address. For example:

```
qq --host 203.0.113.0 login
```

Note

Your user must have `PRIVILEGE_SUPPORT_WRITE` and `PRIVILEGE_SUPPORT_READ`.

4. To install the VPN keys on your cluster, specify your cluster's IP address and the path to the directory that contains the VPN keys. For example:

```
qq --host 203.0.113.0 install_vpn_keys /my/path
```

5. To verify that the VPN keys installed correctly, use the `get_vpn_keys` command. For example:

```
qq --host 203.0.113.0 get_vpn_keys
```

6. Remove any local copies of the VPN key files.

To Install VPN Keys from a Node

Note

On macOS and Linux, you can use the `scp` and `ssh` tools. On Windows Server 2022, Windows Server 2019, and Windows 10 (build 1809 and higher), we recommend installing OpenSSH.

1. Copy the `.zip` file from Qumulo Care to a computer on the same network as your cluster, and decompress the file.
2. To copy the VPN key files to one of your nodes, use the `scp` command. For example:

```
scp /my-path/* admin@203.0.113.0:~/
```

3. To connect to the node to which you copied the VPN key files, use the `ssh` command. For example:

```
ssh admin@203.0.113.0
```

The `qq` CLI is available to the admin user. For example:

```
qq version
```

4. To install the VPN keys on your cluster, specify the path to the directory that contains the VPN keys. For example:

```
sudo qq install_vpn_keys /my/path/
```

5. To verify that the VPN keys installed correctly, use the `get_vpn_keys` command. For example:

```
sudo qq get_vpn_keys
```

To Register Cluster with Cloud-Based Monitoring

1. To retrieve your cluster ID, use the `node_state_get` command.
2. Send the output of the command to Qumulo Care.
3. Use the Web UI to enable Qumulo Care Remote Support.
4. Notify Qumulo Care when this process is complete.

Qumulo Care verifies your VPN functionality and then adds your cluster to Cloud-Based Monitoring.

Configuring SAML Single Sign-On (SSO) for Your Qumulo Cluster

This section explains how to integrate your Qumulo cluster with your organization's single sign-on (SSO) service by configuring Security Assertion Markup Language (SAML) 2.0 for Qumulo Core 5.2.5.1 (and higher).

Configuring the Search Trusted Domains Option in Active Directory for a Qumulo Cluster

This section explains how to restrict the scope of LDAP queries by using the Search Trusted Domains configuration option for a Qumulo cluster joined to an Active Directory (AD) domain.

Authorization

Managing Cross-Protocol Permissions (XPP) in Qumulo Core

This section explains how Cross-Protocol Permissions (XPP) work in Qumulo Core and how to enable, disable, and check the status of XPP by using the `qq` CLI.

External Services

Using Qumulo Core Access Tokens

This section explains how to create and use access tokens—by using the Qumulo REST API, Python SDK, and `qq` CLI—to authenticate external services to Qumulo Core.

Connecting Your Kubernetes Cluster to Your Qumulo Cluster by Using the Qumulo Container Storage Interface (CSI) Driver

This section introduces the Qumulo Container Storage Interface (CSI) driver and explains how you can connect your Kubernetes cluster to your Qumulo cluster by using the Qumulo CSI driver.

Network Configuration

Required Networking Ports for Qumulo Core

This section explains which inbound and outbound networking ports Qumulo Core requires.

Network Multitenancy

Partitioning a Qumulo Cluster into Tenants

This section explains how to enable, disable, and use network multitenancy in Qumulo Core.

In Qumulo Core 5.3.4 (and higher), *network multitenancy* lets you partition a single physical Qumulo cluster into multiple virtual *tenants*. You can define a tenant by using a name and one or more networks. When you assign a network to a tenant, Qumulo Core treats any client that connects from that network as part of that tenant. For more information, see [Connect to Multiple Networks in Qumulo Core](#) on Qumulo Care.

For each tenant, you can specify individual [management protocol access and configuration](#) (page 49) and [file system protocol access and configuration](#) (page 52).

Note

All tenants share the cluster's underlying file system, identity providers, role-based access control (RBAC) configuration, and other global settings.

Prerequisites

Important

If your cluster runs a version of Qumulo Core lower than 6.1.0.3, you must use the `qq multitenancy_enable` command to enable multitenancy for your cluster.

To manage network multitenancy and tenants, your user must have membership in a Qumulo role with the following privileges.

- `PRIVILEGE_NETWORK_READ` : Viewing networks
- `PRIVILEGE_NETWORK_WRITE` : Assigning networks to tenants
- `PRIVILEGE_TENANT_READ` : Viewing tenants
- `PRIVILEGE_TENANT_WRITE` : Enabling network multitenancy and creating, modifying, and deleting tenants

Creating, Configuring, and Unassigning Tenants by Using the qq CLI.

This section explains the lifecycle of working with tenants, including creating tenants, assigning networks to new and existing tenants, viewing tenant configuration, and unassigning tenants.

Step 1: Create a New Tenant

After you enable multitenancy for your cluster, you can create a new tenant.

Use the `qq multitenancy_create_tenant` command and specify a name for your tenant. For example:

```
$ qq multitenancy_create_tenant \  
  --name my_tenant
```

Your cluster creates a new tenant with no networks assigned to it. By default, Qumulo Core disables access to the tenant through all protocols.

Step 2: Assign Networks to a Tenant

To allow a tenant to apply its configuration to clients that connect to the cluster from specific networks, you must associate the tenant with one or more networks. To do this, you can:

- Create a new tenant with networks assigned to it.
- Assign networks to, and unassign networks from, an existing tenant.
- Move networks between tenants.

Creating a New Tenant with Assigned Networks

Use the `qq multitenancy_create_tenant` and specify the tenant name and network ID.

Important

The network must not be assigned to another tenant already.

In the following example, we specify a single network.

```
$ qq multitenancy_create_tenant \  
  --name my_tenant_name \  
  --network-id 2
```

Assigning a Single Network to an Existing Tenant

To assign a single network to a tenant, modify *the tenant that belongs to the network*. Use the `qq network_mod_network` command and specify the network and tenant ID.

```
$ qq network_mod_network \  
  --network-id 3 \  
  --tenant-id 2
```

Assigning Multiple Networks to an Existing Tenant

To assign multiple networks to a tenant, modify *the networks that belong to the tenant*. Use the `qq multitenancy_modify_tenant` command and specify the tenant and network ID.

⚠ Important

Any existing networks not specified after the `--network-id` flag become unassigned.

In the following example, we specify three networks.

```
$ qq multitenancy_modify_tenant \  
  --id 2 \  
  --network-id 2 3 4
```

Unassigning a Single Network from a Tenant

To unassign a single network from a tenant, clear *the tenant that belongs to the network*. Use the `qq network_mod_network` command and specify the network and the `--clear-tenant-id` flag.

ℹ Note

After you unassign a network from a tenant, you can assign it to another tenant.

```
$ qq network_mod_network \  
  --network-id 3 \  
  --clear-tenant-id
```

Unassigning All Networks from a Tenant

To unassign all networks from a tenant, clear *the networks that belong to the tenant*. Use the `qq multitenancy_modify_tenant` command and specify the tenant and network ID.

ℹ Note

- After you unassign a network from a tenant, you can assign it to another tenant.
- Don't specify any arguments for the `--network-id` flag.

```
$ qq multitenancy_modify_tenant \  
  --id 2 \  
  --network-id
```

Moving Networks between Tenants

Use the `qq network_mod_network` command and specify the network and target tenant.

```
$ qq network_mod_network \  
  --network-id 2 \  
  --tenant-id 1
```

Step 3: View Tenant Information

To determine a tenant's network assignments and enabled management and file system protocols, you can view the tenant information.

Viewing Information for a Single Tenant

To view the information for a single tenant, use the `qq multitenancy_get_tenants` command.

```
$ qq multitenancy_get_tenant \  
  --id 1
```

Viewing Information for All Tenants

- In the Web UI, log in to Qumulo Core and then click **Cluster > Network Multitenancy**.
- In the `qq` CLI, use the `multitenancy_list_tenants` command.

Determining the Tenant Assignment for Networks

- To view the information for a single network, use the `qq network_get_network` command.

```
$ qq network_get_network \  
  --network-id 2
```

- To view the information for all networks, use the `qq network_list_networks` command.

Step 4: Delete a Tenant

Important

- When you delete a tenant, Qumulo Core removes the tenant's entire configuration from your cluster, including NFS exports and SMB shares associated with the tenant.
- It isn't possible to delete the last tenant.

To delete a tenant, use the `qq multitenancy_delete_tenant` command and specify the tenant ID.

```
$ qq multitenancy_delete_tenant \  
--id 2
```

Known Network Multitenancy Limitations in Qumulo Core

Currently, Qumulo Core doesn't support:

- Creating or modifying tenants on cloud-based clusters
- Using one VLAN on multiple tenants
- Using a separate Active Directory, standalone LDAP, or user-defined mapping configuration for each tenant
- Using a separate DNS configuration for each tenant
- Scoping RBAC privileges to each tenant

Configuring Management Protocols on a Tenant

This section explains how to configure management protocols for each tenant on a Qumulo cluster.

After you [create tenants \(page 44\)](#) on your Qumulo cluster, you can manage access for clients that connect to the cluster from the tenant's network by enabling or disabling the management protocols for each tenant.

⚠ Important

- Access to a management protocol lets a client use the protocol to view and modify resources across the entire cluster, not only within the client's tenant.
- If you disable a management protocol, you can still access your cluster by using a physical or remote console.

Prerequisites

To configure management protocols, your user must have membership in a Qumulo role with the following privileges.

- `PRIVILEGE_TENANT_READ` : Viewing tenants
- `PRIVILEGE_TENANT_WRITE` : Modifying tenants

Enabling and Disabling REST API Access

The Qumulo REST API lets you manage clusters by using the `qq` CLI, Python bindings, and REST API calls.

- To enable REST API access, use the `qq multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-rest-api` flag.
- To disable access, use the `--disable-rest-api` flag.

In the following example, the cluster begins to accept REST API traffic on TCP port 8000 on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2  
  --enable-rest-api
```

Enabling and Disabling Web UI Access

The Qumulo Core Web UI lets you manage clusters from a browser.

Note

Enabling Web UI access doesn't require enabling REST API access.

- To enable REST API access, use the `qq multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-web-ui` flag.
- To disable access, use the `--disable-web-ui` flag.

In the following example, the cluster begins to serve the Web UI on TCP ports 80 and 433 on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2  
  --enable-web-ui
```

Enabling and Disabling SSH Access

SSH lets you view logs and use the `qq` CLI by using a client to connect to nodes in a cluster remotely.

Note

Enabling SSH access doesn't require enabling REST API access.

- To enable SSH access, use the `qq multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-ssh` flag.
- To disable access, use the `--disable-ssh` flag.

In the following example, the cluster begins to accept SSH traffic on TCP port 22 on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2  
  --enable-ssh
```

Enabling and Disabling Replication Access

Replication lets a cluster transfer data from a directory on one cluster to a directory on another cluster.

- To enable replication access, use the `qq multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-replication` flag.
- To disable access, use the `--disable-replication` flag.

In the following example, the cluster begins to accept replication traffic on TCP port 3712 on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2 \  
  --enable-replication
```

Configuring File System Protocols on a Tenant

This section describes how to configure file system protocols for each tenant on a Qumulo cluster.

After you [create tenants \(page 44\)](#) on your Qumulo cluster, you can manage access for clients that connect to the cluster from the tenant's network by enabling or disabling file system protocols, or configuring file system protocols specifically for each tenant.

Prerequisites

To configure file system protocols, your user must have membership in a Qumulo role with the following privileges.

- `PRIVILEGE_NFS_EXPORT_READ` : View NFS exports
- `PRIVILEGE_NFS_EXPORT_WRITE` : Create, modify, and delete NFS exports
- `PRIVILEGE_NFS_SETTINGS_READ` : View NFS settings
- `PRIVILEGE_NFS_SETTINGS_WRITE` : Modify NFS settings
- `PRIVILEGE_SMB_SHARE_READ` : View SMB shares and settings
- `PRIVILEGE_SMB_EXPORT_WRITE` : Create, modify, and delete SMB shares; modify SMB settings
- `PRIVILEGE_TENANT_READ` : View tenants
- `PRIVILEGE_TENANT_WRITE` : Modify tenants

Enabling NFS and Configuring Settings and Exports

This section explains how to enable the NFS protocol for a tenant, the difference between global settings and settings for each tenant, and how to configure NFS exports for a tenant.

Enabling the NFS Protocol for a Tenant

To let a tenant use NFS to access a specific set of exports, use the `qq` `multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-nfs` flag.

Note

Initially, the tenant has no exports to mount.

In the following example, the cluster begins to accept NFS traffic on TCP and UDP ports 111 and 2019, on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2 \  
  --enable-nfs
```

Configuring the NFS Protocol Globally or for a Tenant

You can use *global settings* that apply to all tenants. For example, to enable NFSv4.1 for all tenants, use the `qq nfs_modify_settings --enable-v4` command. For more information about NFS settings see [Enabling and Using NFSv4.1 on a Qumulo Cluster \(page 101\)](#) and [How NFSv4.1 works with Kerberos in Qumulo Core \(page 104\)](#).

To override this global setting, use the `qq multitenancy_nfs_modify_settings` command to disable NFSv4.1 on a specific tenant, specify the tenant ID, and use the `--disable-v4` flag. In the following example, despite the global setting, the specified tenant no longer accepts NFSv4.1 traffic on port 2049, on all networks assigned to the tenant.

```
$ qq multitenancy_nfs_modify_settings \  
  --tenant-id 2 \  
  --disable-v4
```

To revert to the global setting for a specific tenant, use the `qq multitenancy_nfs_delete_settings` command and specify the tenant ID. For example:

```
$ qq multitenancy_nfs_delete_settings \  
  --tenant-id 2
```

Configuring NFS Exports

To isolate clients within a network in specific directories in the file system, you can configure NFS exports for each tenant.

Note

Because each new tenant has no exports initially, to give clients within that tenant's network access to the cluster over NFS, you must add exports. For more information, see [Create an NFS Export on Qumulo Care](#).

To create an NFS export for a tenant, use the `qq nfs_add_export` command and specify the tenant ID, the export path, and the file system path and use the `--no-restrictions` flag. In the following example, a cluster lets clients access directory `/my-directory` from the export `/my-export` when the clients are in the network assigned to the tenant.

```
$ qq nfs_add_export \  
  --tenant-id 2 \  
  --export-path /my-export \  
  --fs-path /my-directory \  
  --no-restrictions
```

⚠ Important

When multiple tenants exist on a cluster, you must specify the tenant ID.

To move an NFS export from one tenant to another, use the `qq nfs_mod_export` command. In the following example, while the cluster denies access to clients in tenant 2, it lets clients in tenant 1 access the export `/my-export`.

```
$ qq nfs_mod_export \  
  --tenant-id 2 \  
  --export-path /my-export \  
  --new-tenant-id 1
```

Enabling SMB and Configuring Settings and Shares

This section explains how to enable the SMB protocol for a tenant, the difference between global settings and settings for each tenant, and how to configure SMB shares for a tenant.

Enabling the SMB Protocol for a Tenant

To let a tenant use SMB to access a specific set of shares, use the `qq multitenancy_modify_tenant` command, specify the tenant ID, and use the `--enable-smb` flag.

📌 Note

Initially, the tenant has no shares to mount.

In the following example, the cluster begins to accept SMB traffic on TCP port 445, on all networks assigned to the tenant.

```
$ qq multitenancy_modify_tenant \  
  --id 2 \  
  --enable-smb
```

Configuring the SMB Protocol Globally or for a Tenant

You can use *global settings* that apply to all tenants. For example, to require encryption for all tenants, use the `qq smb_modify_settings --encryption-mode required` command. For more information about SMB settings, see the following resources in Qumulo Care:

- [SMB3 Encryption with Qumulo Core](#)
- [SMB Host Restrictions](#)
- [Hide an SMB Share](#)
- [Snapshots: Snapshot Directory Mode](#)

To override this global setting, use the `qq multitenancy_smb_modify_settings` command to allow clients on only specific networks to require encryption. In the following example, despite the global setting, the specified tenant accepts unencrypted SMB requests on all networks assigned to the tenant.

```
$ qq multitenancy_smb_modify_settings \  
--tenant-id 2 \  
--encryption-mode none
```

To revert to the global setting for a specific tenant, use the `qq multitenancy_smb_delete_settings` command and specify the tenant ID. For example:

```
$ qq multitenancy_smb_delete_settings \  
--tenant-id 2
```

Configuring SMB Shares

To isolate clients within a network in specific directories in the file system, you can configure SMB shares for each tenant.

Note

Because each new tenant has no shares initially, to give clients within that tenant's network access to the cluster over SMB, you must add shares. For more information, see [Create an SMB Share on Qumulo Care](#).

To create an SMB share for a tenant, use the `qq smb_add_share` command and specify the tenant ID, the share path, and the file system path and use the `--all-access` flag. In the following example, a cluster lets clients access access directory `/my-directory` from the share `/my-share` when the clients are in the network assigned to the tenant.


```
$ qq smb_add_share \  
  --tenant-id 2 \  
  --name /my-share \  
  --fs-path /my-directory \  
  --all-access
```

⚠ Important

When multiple tenants exist on a cluster, you must specify the tenant ID.

To move an SMB share from one tenant to another, use the `qq smb_mode_share` command. In the following example, while the cluster denies access to clients in tenant 2, it lets clients in tenant 1 access the share `/my-share`.

```
$ qq smb_mod_share --tenant-id 2 \  
  --name foo \  
  --new-tenant-id 1
```

Using the Microsoft Management Console (MMC)

To create, modify, and delete SMB shares for each tenant, you can use the MMC Shared Folders snap-in. For more information, see [Manage File Shares with Shared Folders MMC Snap-in](#) on Qumulo Care.

⚠ Important

Because folder paths always start at the root of the file system, using the MMC on a Windows client on a tenant's network causes the MMC to modify that tenant's shares.

Global Configuration of S3 and FTP Protocols

Qumulo Core doesn't permit configuring file system protocols such as S3 and FTP for each tenant. You must enable, disable, and configure these protocols globally.

⚠ Important

Enabling these protocols makes them available on all networks.

For more information, see [S3 API \(page 113\)](#) in this guide and [FTP in Qumulo Core](#) on Qumulo Care.

Listing Client Connections

To determine tenant activity and client types connected to the tenant, use the `qq network_list_connections` command. For more information, see [Balance of Client Connections on your Qumulo Cluster](#) on Qumulo Care.

The following is example output from the command.

```
[
  {
    "connections": [
      {
        "type": "CONNECTION_TYPE_NFS",
        "ip_address": "255.0.0.1",
        "tenant_id": 1
      },
      {
        "type": "CONNECTION_TYPE_SMB",
        "ip_address": "192.168.0.1",
        "tenant_id": 2
      }
    ],
    "id": 1
  },
  ...
]
```

Web UI

Setting the Web UI Login Banner

This section explains how to set a login banner for the Qumulo Core Web UI.

Setting the Web UI Inactivity Timeout

This section explains how to set an inactivity timeout for the Qumulo Core Web UI.

qq CLI

Enabling Autocomplete for the qq CLI

This section explains how to enable automatic command completion for the qq CLI and for command aliases.

Metadata

Managing User-Defined Metadata in Qumulo Core

This section explains how to create, retrieve, list, and delete user-defined metadata in Qumulo Core by using the `qq` CLI.

Snapshots

How Snapshots Work in Qumulo Core

This section explains snapshots, their storage usage, and their locking functionality in Qumulo Core.

Managing Snapshots in Qumulo Core

This section explains how to create on-demand snapshots and snapshot policies, view and search for existing snapshots, and delete snapshots by using the Web UI. It also explains how to create snapshots on a schedule, create a snapshot with an expiration time, and modify a snapshot's expiration time.

Locking and Unlocking Snapshots in Qumulo Core

This section explains how to lock or unlock a snapshot by using a key located in the Qumulo file system key store and the `qq` CLI. In addition, it explains how to lock policy-created snapshots for local policies and for policies that are part of a replication target relationship.

Recovering Files by Using Snapshots

This section explains how to use snapshots to recover files.

Encryption and Data Security

Managing Encryption at Rest in Qumulo Core

This section explains how encryption at rest works in Qumulo Core, how to rotate master keys, how to configure a Key Management Server (KMS), and how to ensure that the master keys across your cluster are secured correctly by using the `qq` CLI.

⚠ Important

- Upgrading a Qumulo cluster from a version of Qumulo Core lower than 3.1.5 doesn't enable encryption automatically. You must rebuild your cluster to take advantage of this feature. When you [create a new cluster](#), Qumulo Core enables encryption automatically and distributes the master key to all nodes in the cluster.
- In case of replication processes, Qumulo Core maintains the encryption type after data transfers. Although source and target clusters don't require encryption for replication, we strongly recommend encrypting both source and target clusters.

How Encryption at Rest and Master Keys Work in Qumulo Core

In Qumulo Core 3.1.5 (and higher), in addition to encrypting data *in transit* (for example, to clients that use SMBv.3.1), software-based encryption also secures data *at rest* for on-premises clusters. Qumulo Core encrypts all data and metadata in the file system. Removing or reinserting drives and replication doesn't affect encryption at rest. For more information, see [Encryption Limitations \(page 70\)](#).

Qumulo Core uses a *master key* to protect the *data key* that encrypts the data on the cluster. The master key is stored either locally—on the boot drive of every node, in a file that only the `root` user can access—or on an external Key Management Server (KMS)—from where the system retrieves the master key upon startup. Both approaches help protect your data from potential threats such as a malicious actor's access to stolen or decommissioned disks.

Retrieving Information about a Qumulo Cluster's Encryption Configuration

This section explains how to retrieve the status or detailed information about an active encryption configuration for a Qumulo cluster and gives examples for a system that uses a locally stored master key and a system that uses a Key Management Server (KMS).

📘 Note

The `qq` CLI commands `encryption_get_key_store` and `encryption_get_status` require the `PRIVILEGE_ENCRYPTION_READ` privilege.

To View the Status of an Active Encryption Configuration

Use the `qq encryption_get_status` command.

The following is example JSON output for a locally stored master key.

```
{
  "last_key_rotation_time": "2022-11-20T12:15:25.683207795Z",
  "status": "Encrypted",
  "type": "Local"
}
```

The following is example JSON output for a master key stored in a KMS.

```
{
  "ca_cert_expiry": "2027-04-18T19:55:17Z",
  "client_cert_expiry": "2027-04-18T19:55:17Z",
  "last_key_rotation_time": "2023-09-05T20:15:40.06864014Z",
  "last_status_update_time": "2023-09-05T20:28:58.108120131Z",
  "status": "KMS Available",
  "type": "KMS"
}
```

To View Detailed Information for an Active Encryption Configuration

Use the `qq encryption_get_key_store` command.

The following is example JSON output for a locally stored master key.

```
{
  "config_details": {
    "status": "Encrypted"
  },
  "config_type": "Local"
}
```

The following is example JSON output for a master key stored in a KMS.

```
{
  "config_details": {
    "config_creation_time": "2024-02-28T20:01:25.683207795Z",
    "hostname": "kms-server.example.com",
    "key_id": "abcd-1234-efgh-5678",
    "port": 5696
  },
  "config_type": "KMS"
}
```

Configuring Qumulo Core to Use a Master Key Stored Locally or in a Key Management Server (KMS)

This section explains how to configure Qumulo Core to use a master key stored locally or in a Key Management Server (KMS) by using the `qq` CLI.

Note

- The `qq` CLI command `encryption_set_key_store` requires the `PRIVILEGE_ENCRYPTION_WRITE` privilege.
- To be able to configure an external KMS, the KMS must support Key Management Interoperability Protocol (KMIP) 1.0.

To Configure Qumulo Core to Use a Master Key Stored Locally

Important

- While the *master* key on your boot drive encrypts your *data* keys, the master key *itself* isn't encrypted.
- The boot drive contains the disk image, the installed build of Qumulo Core, and configuration files. In the unlikely event that your boot drive fails and requires replacement, remove the encrypted data keys associated with the master key from the boot drive by [rotating the master key \(page 69\)](#). When you complete the key rotation process, you can dispose of the failed boot drive securely.
- To avoid potential decryption, ensure that your data keys eventually *age out* by rotating the master key any time you replace a drive in your cluster.

1. To configure the system to use a local key store, use the `qq encryption_set_key_store local` command.
2. To confirm that the system is configured correctly, use the `qq encryption_get_status` command ([page 66](#)).

In the output, ensure that the `type` field is set to `Local`.

To Configure Qumulo Core to Use a Master Key Stored in a Key Management Server (KMS)

! Caution

- If the master key is deleted from the KMS, and all nodes in the cluster are rebooted, all data on the cluster becomes permanently unrecoverable.
- If you allow the certificates to expire, or the master key is deleted accidentally, you must create a new, valid configuration as soon as possible. To warn you of this scenario, the Web UI indicates if any of your certificates are about to expire, or if the configured master key becomes unavailable.

1. To configure the system to use a KMS, use `qq encryption_set_key_store kms` command and specify the path to the client certificate, private key, the server CA certificate, the key ID, and the KMS server hostname. For example:

```
qq encryption_set_key_store kms \  
  --client-cert path/to/client_cert.pem \  
  --client-private-key path/to/client_pk.pem \  
  --server-ca-cert /path/to/server_cert.pem \  
  --key-id abcd-1234-efgh-5678 \  
  --host-name kms-server.example.com
```

2. To confirm that the system is configured correctly, use the `qq encryption_get_key_store` command (page 67).

In the output, ensure that the `type` field is set to `KMS`.

Rotating the Master Key

This section explains how to rotate the master key and check the encryption status for your cluster by using the `qq` CLI and how to check the encryption status by using the Web UI.

! Caution

Qumulo Care team members can help you rotate your master keys (page 69). However, they don't have access to your encryption keys and can't retrieve them for you.

To Rotate Master Keys Stored Locally

1. Use the `qq rotate_encryption_keys` command.

When the process is complete, the command shows the `Key rotation complete` message.

2. To view your cluster's encryption status and the last key rotation time, use the `qq encryption_get_status` command (page 66).

To Rotate Master Keys Stored in a Key Management Server (KMS)

1. Use the `qq rotate_encryption_keys` command and specify the key ID. For example:

```
qq rotate_encryption_keys --key-id abcd-1234-efgh-5678
```

✓ Tip

The key ID might be different from the key name.

2. To ensure that the system is using the new key, use the `qq encryption_get_key_store` command (page 67).

In the output, ensure that the `key_id` field lists the new key ID.

To Check the Encryption Status of a Qumulo Cluster by Using the Web UI

1. Log in to Qumulo Core.
2. On the **Dashboard** page, in the **Cluster Overview** section, click **More details**.
3. If encryption is enabled for your cluster, the **Cluster** page shows the message **Data Encrypted**.

Encryption Limitations

- Qumulo Core doesn't encrypt host file system data on the node (such as system logs, core files, and so on).
- Qumulo Core doesn't support removing encryption from encrypted clusters.
- On encrypted systems, single-stream throughput and latency might experience up to 5-10% degradation for writes and up to 5% for reads.
- Qumulo Cloud clusters don't support encryption at rest and should use cloud-native solutions for this functionality.

Generating and Storing ECDSA Keys on a Qumulo Cluster

This section explains how to generate Elliptic Curve Digital Signature Algorithm (ECDSA) keys and ECDSA verification signatures that are compatible with the Qumulo file system key store.

Managing Security Keys in the Qumulo File System Key Store

This section explains how to manage security keys in the Qumulo file system key store by using the `qq` CLI.

Node Addition and Replacement

How Drive and Node Failure Protection Works in Qumulo Core

This section provides an overview of how Qumulo clusters ensure continued operation in the event of a drive or node failure.

How Qumulo Core Ensures Fault Tolerance

Qumulo Core protects your cluster with a **6,4 erasure coding** (2 concurrent drive failures or 1 node failure), at minimum. When a drive fails, Qumulo Core begins to rebuild the data that was previously stored on the failed drive.

Note

When Qumulo Core finishes reprotecting the drive, it resets the fault tolerance for the cluster, regardless of whether you have replaced the failed drive.

As a cluster increases in size, Qumulo Core makes additional fault tolerance options available during the cluster creation process. After creating a cluster, you can use [Adaptive Data Protection \(page 82\)](#) to include the cluster's fault tolerance during node-add procedures.

Note

Depending on a cluster's size constraints, certain configurations (such as 1 concurrent drive failure or 4 node failures) might not be possible.

To view the fault tolerance of your Qumulo cluster:

- **Web UI:** Navigate to the **Cluster Overview** page
- **qq CLI:** Use the `qq protection_status_get` command
- **REST API:** Call the `/v1/cluster/protection/status` endpoint

Read-Only Mode Scenario for Hybrid Nodes





When a hybrid node goes offline for a substantial period of time, there is a risk of the node entering read-only mode because Qumulo Core writes all inbound operations only to the node's SSDs.

The length of time before this scenario takes place depends on the number of drives in a node and the rate of incoming writes, deletes, and changes. For more information, see [Understanding Offline Nodes and Checking for Free Space \(page 92\)](#). If you encounter this scenario, [contact the Qumulo Care team](#).

The following sections describe various drive and node failure protection configurations and how they correspond to failure scenarios and data protection states.






2-Drive, 1-Node Protection (2,1)

This is the default system configuration. This configuration requires a minimum of 4 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
3 (or more) drive failures or multiple node failures	 High	The data is unavailable but intact.







3-Drive, 1-Node Protection (3,1)

This configuration requires a minimum of 5 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
3 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
4 (or more) drive failures or multiple node failures	 High	The data is unavailable but intact.







3-Drive, 2-Node Protection (3,2)


This configuration requires a minimum of 11 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
3 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
2 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
4 (or more) drive failures or more than 2 node failures	 High	The data is unavailable but intact.

3-Drive, 3-Node Protection (3,3)








This configuration requires a minimum of 11 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Low	The data is protected. You can replace a failed drive at any time.
3 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
2 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
3 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.

Failure Scenario	Severity	Data Protection State
4 (or more) drive failure or more than 3 node failures	 High	The data is unavailable but intact.



4-Drive, 2-Node Protection (4,2)







This configuration requires a minimum of 12 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Low	Data is protected. You may replace a failed drive at any time.
3 drive failures	 Medium	Data is protected. You may replace a failed drive at any time.
4 drive failures	 Medium	Data is protected. You may replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
2 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
5 (or more) drive failures or more than 2 node failures	 High	The data is unavailable but intact.

4-Drive, 3-Node Protection (4,3)







This configuration requires a minimum of 24 nodes.




Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Low	The data is protected. You can replace a failed drive at any time.

Failure Scenario	Severity	Data Protection State
3 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
4 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
2 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
3 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
5 (or more) drive failure or more than 3 node failure	 High	The data is unavailable but intact.

4-Drive, 4-Node Protection (4,4)

This configuration requires a minimum of 24 nodes.

Failure Scenario	Severity	Data Protection State
1 drive failure	 Low	The data is protected. You can replace a failed drive at any time.
2 drive failures	 Low	The data is protected. You can replace a failed drive at any time.
3 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
4 drive failures	 Medium	The data is protected. You can replace a failed drive at any time.
1 node failure	 High	The data is protected. The cluster is at risk of going into read-only mode.
2 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.

Failure Scenario	Severity	Data Protection State
3 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
4 node failures	 High	The data is protected. The cluster is at risk of going into read-only mode.
5 (or more) drive failures or more than 4 node failures	 High	The data is unavailable but intact.

Adding Nodes to an Existing Qumulo Cluster

This section explains how to add new HPE, Supermicro, or Quiver nodes to an existing cluster.

After you connect and power on your new nodes, Qumulo Core discovers any unconfigured nodes automatically and prompts you to add nodes in the Web UI.

If Qumulo Core doesn't discover any unconfigured nodes, it displays the message **No unconfigured nodes found**. If you expect to see nodes, [contact the Qumulo Care team](#).

Note

- Qumulo Core requires a short time to update the total available storage.
- Existing nodes retain their numbering.

Prerequisites

- **Sufficient Static IP Addresses:** The number of static IP addresses must be equal to or greater than the number of nodes in your cluster. For more information, see [IP Failover with Qumulo Core](#) on Qumulo Care.
- **Same Qumulo Core Version on All Nodes:** For information about upgrading Qumulo Core, see [Performing a Clean Installation of Qumulo Core](#) and [Performing Qumulo Core Upgrades by Using the Web UI](#) on Qumulo Care.

Step 1: Resolve Drive Compatibility Issues

Important

If the version of Qumulo Core on your existing nodes predates the Qumulo-certified drives that you received with your new nodes, you can't install a lower version of Qumulo Core on your new node and Qumulo Core displays the message **Installation failed**. Use the cluster logs to identify any incompatible drives.

To receive support for new, Qumulo-certified drives, do one of the following:

- Upgrade your existing cluster to the latest version of Qumulo Core and then install the same version of Qumulo Core on your new node. For more information, see [Performing Qumulo Core Upgrades by Using the Web UI](#) on Qumulo Care.
- Update the Supported Drive List on your new node. For more information, [contact the Qumulo Care team](#).

Step 2: Add Your New Nodes to an Existing Qumulo Cluster

1. Log in to the Web UI.
2. Click **Cluster > Add Nodes**.
3. On the **Add Nodes** page, select unconfigured nodes to add to your cluster.
4. Click **Add Selected Nodes to Cluster**.
5. In the **Add <N> nodes to cluster <name>?** dialog box, click **Yes**.

If you add one or more node model types, a message reminds you about Qumulo Core adding a new model type to your cluster.

Qumulo Core configures your new nodes and adds them to your cluster.

On the **Cluster** page, the Web UI shows the banner **Successfully added <N> nodes to the cluster** and the total available storage.

Increasing the Node Fault Tolerance Level for Your Qumulo Cluster during Node-Add Operations

This section explains how to increase the node fault tolerance level for your Qumulo cluster during node-add operations.

- In Qumulo Core 5.1.3 (and higher) you can increase the node fault tolerance level for an existing cluster during the cluster expansion process.
- In Qumulo Core 6.1.0.3 (and higher), you can manage the drive and node fault tolerance levels during the cluster expansion process by [changing the data protection configuration \(page 82\)](#).

Important

We strongly recommend contacting the [Qumulo Care team](#) before proceeding with cluster expansion.

To Reconfigure Your Cluster's Node Fault Tolerance Level

1. Follow the instructions in [Adding Nodes to an Existing Qumulo Cluster \(page 79\)](#).

Note

- When a tradeoff between increasing node fault tolerance and maximizing usable capacity is available for your cluster, the **Cluster > Add Nodes** page shows the **Accept a trade-off in the increase of usable capacity** checkbox.
- When a tradeoff isn't available, the page shows a message which indicates that it is possible to increase the node fault tolerance level.

2. Before you click **Yes** in the **Add <N> nodes to cluster <MyCluster>?** dialog box, check that the projected capacity matches the expected capacity.
3. After the cluster expansion process finishes, Qumulo Core begins data protection reconfiguration automatically.
4. To monitor this process, click **Cluster > Overview**. On the **Cluster** page, in the protection status section, you can view the rebalance phase status and the estimated time to completion.
5. When the restriper completes the data protection reconfiguration, the **Data Protected** section of the **Cluster** page shows the increased node fault tolerance level.

Adding Nodes with Adaptive Data Protection to a Qumulo Cluster

This section explains how to configure Adaptive Data Protection for your Qumulo cluster during node-add operations.

- Qumulo Core 6.1.0.1 and 6.1.1 support Adaptive Data Protection by letting you reconfigure your cluster's fault tolerance level and storage efficiency only when you add nodes to your cluster.

Depending on your cluster, Qumulo Core shows configuration options that offer better fault tolerance levels, better storage efficiency, or both benefits. To enable Adaptive Data Protection for your cluster, you must [contact the Qumulo Care team](#).

- In Qumulo Core 6.1.2 (and higher), you can change your cluster's data protection configuration when you add or replace nodes by using the `qq` CLI.

Note

Your cluster's usable capacity doesn't increase until the data protection reconfiguration is complete. Because data protection reconfiguration is a long process (that can take days or weeks), we don't recommend beginning this process when your cluster is nearly running out of usable capacity. In this scenario, [contact the Qumulo Care team](#).

To Add a Node with Adaptive Data Protection to a Qumulo Cluster

A cluster's data protection configuration includes the stripe configuration (for example, `10.8`) and the node fault tolerance level.

Important

Ensure that your configuration and node order are correct. Unless you add more nodes, it isn't possible to revert this operation or reorder nodes after adding them to a cluster.

1. To select the new data protection configuration for your nodes, [contact the Qumulo Care team](#).
2. Rack and wire your new nodes and then power them on.
3. To determine the UUIDs of the nodes to add to your cluster, use the `qq unconfigured_nodes_list` command.
4. Write down the UUIDs of the nodes that you want to add to the cluster, in the order that you want to add them.

5. To add your nodes to the cluster, use the `qq add_nodes` command and specify the node UUIDs (their ordering in the command determines their ordering in the cluster), the configuration type, and the node fault tolerance level.

In the following example, we change the stripe configuration to 10.8 and the node fault tolerance level to 2 nodes.

```
qq add_nodes \  
  --node-uuids 12345a6b-7c89-0d12-3456-78fe9012f345 \  
    abcdef2-g3hi-j4kl-mnop-qr56stuv7wxy \  
  --target-stripe-config 10 8 \  
  --target-max-node-failures 2
```

The following is example output from the command.

```
Current cluster:  
  Usable capacity: 600 TB  
  Node fault tolerance level: 1 node  
With the selected node-add operation and data protection reconfiguration:  
  Usable capacity: 800 TB  
  Node fault tolerance level: 2 nodes
```

6. To confirm the configuration with the selected node-add and data protection configuration operations, enter `yes`.

Monitoring the Data Protection Reconfiguration Process

To view the progress of the three stages of the data protection reconfiguration process, log in to the Qumulo Core Web UI and click **Cluster**.

1. Qumulo Core begins to move data to new nodes in the cluster and the Web UI displays the message **Rebalancing for data protection reconfiguration**.
2. Qumulo Core reencodes all data on your cluster and the Web UI displays the message **Reconfiguring data protection**.

Note

In certain scenarios, this stage might appear to pause while the system performs preparatory work on the cluster.

When this stage is complete, your data is protected according to the cluster's new configuration and the system begins to use the new drive and node fault tolerance levels.

3. Qumulo Core adds new capacity to your cluster and the Web UI displays the message **Rabalancing**.

If you initiated the reconfiguration process as part of a node replacement step, the system migrates data from the existing nodes in the cluster.

Cluster Availability During the Reconfiguration Process

Your cluster remains available throughout the data protection reconfiguration process.

- You can upgrade Qumulo Core.
- Your cluster maintains the ability to recover from node and drive failure automatically.

During the reconfiguration process, drive and node fault tolerance levels remain at the minimums that the existing and new configurations specify. For example, if your existing cluster has 2-node and 2-drive fault tolerance, and you initiate reconfiguration where the new configuration has 1-node and 3-drive fault tolerance, your cluster has 1-node and 2-drive fault tolerance during the reconfiguration process.

Note

- To avoid impact to frontend workloads, Qumulo Core slows down the reconfiguration process automatically.
- When Qumulo Core finds missing nodes or drives, it pauses the reconfiguration process. When you replace or bring the nodes or drives online, the reconfiguration process continues.
- It isn't possible to add or replace nodes during the reconfiguration process.

Replacing Nodes in a Qumulo Cluster by Performing a Transparent Platform Refresh

This section explains how to replace nodes that have reached retirement or end of life by performing a two-stage transparent platform refresh on clusters that run Qumulo Core 6.1.0.3 (and higher).

Note

- Qumulo Core doesn't support replacing nodes in clusters with more than 100 nodes.
- In Qumulo Core 6.1.2.2 (and higher), you can use the `qq` CLI to replace nodes. To replace nodes on a lower version of Qumulo Core, contact the [Qumulo Care team](#).

How Transparent Platform Refresh Works

Transparent platform refresh comprises two stages. For help with your node replacement plan, [contact the Qumulo Care team](#).

Stage 1: Register a Node Replacement Plan

In this stage, you [register a node replacement plan \(page 86\)](#) with your cluster. The plan includes information about the nodes to replace and the data protection configuration.

In following example, we use a four-node cluster and:

- Replace nodes 1-4 with five new nodes in a single step
- Change the data protection configuration to the **8.6** stripe configuration with 1-node fault tolerance

Stage 2: Execute the Node Replacement Plan Steps

In this stage, you [execute the node replacement plan's steps \(page 87\)](#) and Qumulo Core performs data protection reconfiguration.

Note

It isn't possible to add nodes or begin another node replacement step while a node replacement step is already in progress.

There are two node replacement plan types:

- **Single-Step Node Replacement:** Qumulo Core adds all new nodes and removes all nodes marked for replacement in a single step. Use this approach when the node replacement speed is a priority.

- **Multi-Step Node Replacement:** Each step of the plan adds some the new nodes and removes some nodes marked for replacement. Use this approach when rack space or switch port capacity in your data center is limited.

Cluster Properties During Node Replacement

- When a replacement step begins, Qumulo Core distributes floating IP addresses among the nodes in the [combined cluster \(page 86\)](#). After Qumulo Core removes nodes marked for replacement, it redistributes any client connections that use floating IP addresses among the nodes that remain in the cluster.

•

While a node replacement step is in progress, both new nodes and nodes marked for replacement appear on the **Cluster** page of the Web UI and clients can connect to any of the nodes in the *combined cluster* while the step is in progress.

- When a node replacement step is complete, the reassignment of static IP addresses differs between versions of Qumulo Core:
 - In Qumulo Core 6.3.0.1 (and higher), the static IP addresses assigned to nodes remain unchanged and Qumulo Core removes only the static IP addresses for nodes removed from the cluster.
 - In Qumulo Core versions lower than 6.3.0.1, Qumulo Core reassigns static IP addresses to different nodes. To view the reassigned IP addresses in the Web UI, click **Cluster > Network Configuration**.
- When Qumulo Core adds nodes to a cluster, it assigns node IDs sequentially, without reusing or changing IDs.

For example, if you have a four-node cluster with node IDs 1-4, and you replace node IDs 2 and 3 with two new nodes, after node replacement the cluster contains node IDs 1, 4, 5, and 6. If you add another node, it has the ID 7.

- A cluster's usable capacity doesn't increase until:
 - Any data protection reconfiguration is complete
 - The last step of the node replacement plan is in progress

For example, if you replace nodes in a single step without data protection reconfiguration, usable capacity increases as soon as Qumulo Core begins the step.

Prerequisites

Ensure that the number of static and floating IP addresses is equal to or greater than the number of nodes in the [combined cluster \(page 86\)](#).

Step 1: Register a Node Replacement Plan by Using the qq CLI

1. Use the `qq replace_nodes register_plan` command and the `--nodes-to-be-replaced` flag to specify the nodes to replace and the `--target-stripe-config` flag to specify the stripe configuration. For example:

```
qq replace_nodes register_plan \  
  --nodes-to-be-replaced 1 2 3 4 \  
  --target-stripe-config 8 6
```

Qumulo Core stores the node replacement plan on your cluster.

Note

- If your plan includes data protection reconfiguration, Qumulo Core records only the stripe configuration. You specify the node fault tolerance when you execute the plan steps.
- If your plan doesn't include data protection reconfiguration, you can omit the `--target-stripe-config` flag.
- To replace all nodes in the cluster, use the `--replace-all` flag instead of the `--nodes-to-be-replaced` flag.

2. Rack and wire your new nodes and then power them on.
3. To determine the UUIDs of the nodes to add to your cluster, use the `qq unconfigured_nodes_list` command.
4. Write down the UUIDs of the nodes that you want to add to the cluster, in the order that you want to add them.

Step 2: Execute the Node Replacement Plan Steps by Using the qq CLI

1. Use the `qq replace_nodes add_nodes_and_replace` command to initiate each step, the `--nodes-being-replaced` flag to specify the nodes to replace, and the `--node-uuids` flag to specify the nodes to add during the current step.

Important

Qumulo Core adds nodes to the cluster in the order in which you list their UUIDs after the `--node-uuids` flag. When you begin the node replacement step, it isn't possible to revert this operation or reorder nodes after adding them to a cluster.

If your plan includes data protection reconfiguration, use the `--reconfigure-data-protection` and `--target-max-node-failures` flags to initiate the reconfiguration during the current step. For example:

```
qq replace_nodes add_nodes_and_replace \  
  --nodes-being-replaced 1 2 3 4 \  
  --node-uuids 12345a6b-7c89-0d12-3456-78fe9012f345 abcdef12-g3hi-j4kl-mnop-qr  
56stuv7wxy \  
  --reconfigure-data-protection \  
  --target-max-node-failures 1
```

The following is example output from the command:

```
Current cluster:  
  Usable capacity: 200 TB  
  Node fault tolerance level: 1 node  
With the selected node replacement step:  
  Usable capacity: 220 TB  
  Node fault tolerance level: 1 node
```

Note

To replace all nodes in the cluster, use the `--replace-all` flag instead of the `--nodes-being-replaced` flag.

2. To confirm the reconfiguration with the selected node-replace and data protection configuration operations, enter `yes`.

For more information, see [Monitoring the Data Protection Reconfiguration Process \(page 89\)](#).

3. Wait for the node replacement step to complete.

After each node replacement step, Qumulo Core begins to migrate data from existing nodes in the background.

Note

This is a long process (that can take days or weeks). When the data migration is complete, Qumulo Core removes the nodes marked for replacement from the cluster. These nodes no longer appear on the **Cluster** page of the Web UI.

4. Unrack the removed nodes from your data center.

5. Initiate the next node replacement step.

Viewing, Editing, and Cancelling the Node Replacement Plan

- To view the current node replacement plan, use the `qq replace_nodes get_plan` command.

If a node replacement step is in progress, the command shows the list of nodes in process of being replaced during the current step.

- To edit the node replacement plan after you register it with your cluster, use the `qq replace_nodes register_plan` command with a [new node replacement plan \(page 86\)](#).
- To cancel the current node replacement plan, use the `qq replace_nodes cancel_plan` command.

Important

Cancelling a node replacement plan after executing one or more steps might make it impossible to reregister and complete the plan.

Monitoring the Data Protection Reconfiguration Process

To view the progress of the three stages of the data protection reconfiguration process, log in to the Qumulo Core Web UI and click **Cluster**.

1. Qumulo Core begins to move data to new nodes in the cluster and the Web UI displays the message **Rebalancing for data protection reconfiguration**.
2. Qumulo Core reencodes all data on your cluster and the Web UI displays the message **Reconfiguring data protection**.

Note

In certain scenarios, this stage might appear to pause while the system performs preparatory work on the cluster.

When this stage is complete, your data is protected according to the cluster's new configuration and the system begins to use the new drive and node fault tolerance levels.

3. Qumulo Core adds new capacity to your cluster and the Web UI displays the message **Rabalancing**.

If you initiated the reconfiguration process as part of a node replacement step, the system migrates data from the existing nodes in the cluster.

Cluster Availability During the Reconfiguration Process

Your cluster remains available throughout the data protection reconfiguration process.

- You can upgrade Qumulo Core.
- Your cluster maintains the ability to recover from node and drive failure automatically.

During the reconfiguration process, drive and node fault tolerance levels remain at the minimums that the existing and new configurations specify. For example, if your existing cluster has 2-node and 2-drive fault tolerance, and you initiate reconfiguration where the new configuration has 1-node and 3-drive fault tolerance, your cluster has 1-node and 2-drive fault tolerance during the reconfiguration process.

Note

- To avoid impact to frontend workloads, Qumulo Core slows down the reconfiguration process automatically.
- When Qumulo Core finds missing nodes or drives, it pauses the reconfiguration process. When you replace or bring the nodes or drives online, the reconfiguration process continues.
- It isn't possible to add or replace nodes during the reconfiguration process.

Improving Performance by Migrating a Qumulo Cluster to a Different License Class

This section explains how you can improve the performance of a cluster that runs Qumulo Core 6.1.0.3 (or higher) by migrating it a different license class (for example, from hybrid to all-NVMe nodes).

⚠ Important

Because it isn't possible to have nodes with different license classes in the same cluster, you must migrate all nodes in your cluster to the same license class.

For help with migrating your cluster to a different license class, [contact the Qumulo Care team](#).

Understanding Offline Qumulo Core Nodes and Checking for Free Space

This section explains what happens when a Qumulo Core node goes offline and how you can check the remaining free space.

What Happens When a Qumulo Core Node Goes Offline

Qumulo Core uses *erasure coding* to let multiple drives or nodes to go offline but continue to serve data. For more information, see [Qumulo Drive Failure Protection](#) on Qumulo Care.

On hybrid Qumulo nodes (that have HDDs and SSDs), Qumulo Core attempts to maintain cluster functionality for as long as possible. When a node goes offline, the cluster evicts existing data promoted to SSDs on the remaining nodes and makes all writes to the free space on the SSDs. When the node comes online, Qumulo Core begins to push writes to the HDDs that back the SSDs.

Important

Writes, deletes, and changes count towards SSD space. For more information, see [Checking Remaining Free Space on Your Cluster \(page 92\)](#).

A cluster can operate with an offline node until its SSD space fills up. When a cluster has no more free space, the cluster goes into read-only (`ENOSPC`) state until the node comes back online and all nodes rejoin the quorum.

Note

The amount of time that users have before the cluster enters the `ENOSPC` state depends on the rate of change in new rates to the cluster.

Checking Remaining Free Space on Your Cluster

You can estimate the amount of time before your cluster enters the `ENOSPC` state by running multiple iterations of the `debug_metrics_get` command (`metrics_get` in Qumulo Core 4.2.0 and lower) on every node in your cluster.

The output of the command shows the number of valid tokens remaining for writes to the cluster. At 50,000 (or fewer) tokens, an `ENOSPC` event is imminent.

Note

Because SSD space on the cluster is limited, new writes, deletes, and changes consume any reclaimable tokens.

Qumulo Core Version	Command
4.2.1 (and higher)	<pre>sudo qsh -c /opt/qumulo/qq debug_metrics_get \ --measurement space_agent grep reclaimable; sleep 60; \ sudo qsh -c /opt/qumulo/qq debug_metrics_get \ --measurement space_agent grep reclaimable</pre>
3.3.2 - 4.2.0	<pre>sudo qsh -c /opt/qumulo/qq metrics_get \ --measurement space_agent grep reclaimable; sleep 60; \ sudo qsh -c /opt/qumulo/qq metrics_get \ --measurement space_agent grep reclaimable</pre>
3.1 - 3.3.1	<pre>sudo /opt/qumulo/qq metrics_get \ --measurement space_agent grep reclaimable; sleep 60; \ sudo /opt/qumulo/qq metrics_get \ --measurement space_agent grep reclaimable</pre>

You can also [contact the Qumulo Care team](#) for a time estimate.

Data Replication

Creating and Managing a Continuous Replication Relationship in Qumulo Core

This section explains how to create, authorize, modify, and delete a replication relationship by using the Qumulo Core Web UI.

Using Qumulo Shift-To to Copy Objects to Amazon S3

This section explains how to use Shift-To to copy objects from a directory in a Qumulo cluster to a folder in an Amazon Simple Storage Service (Amazon S3) bucket and how to manage Shift relationships.

Using Qumulo Shift-From to Copy Objects from Amazon S3

This section explains how to use Shift-From to copy objects from a folder in an Amazon Simple Storage Service (Amazon S3) bucket (cloud object store) to a directory in a Qumulo cluster and how to manage Shift relationships.

File System Changes

How File System Change Notifications Work in Qumulo Core

This section describes how file system change notifications work in Qumulo Core and explains request filtering, recursion, and the three configuration modes for notification requests.

Watching for File Attribute and Directory Changes by Using SMB2 CHANGE_NOTIFY

This section lists the completion filters that an SMB client can request and the corresponding actions that Qumulo Core returns for a matched change.

Watching for File Attribute and Directory Changes by Using REST

This section describes how to configure Qumulo Core and watch for file attribute and directory changes by using REST.

NFS

Creating and Managing an NFS Export in Qumulo Core

This section explains how to create, modify, and delete an NFS export by using the Qumulo Core Web UI.

Enabling and Using NFSv4.1 on a Qumulo Cluster

This section explains how to configure your cluster for a supported export configuration and enable or disable NFSv4.1 on your cluster.

Managing File Access Permissions by Using NFSv4.1 Access Control Lists (ACLs)

This section explains how to use Qumulo Core's implementation of NFSv4.1 with access control lists (ACLs) to manage access permissions for files.

Configuring and Troubleshooting Host Access Rules for NFS Exports in Qumulo Core

This section explains how host access rules work in Qumulo Core and how to configure and troubleshoot them.

NFSv4.1 with Kerberos

How NFSv4.1 Works with Kerberos in Qumulo Core

This section provides an overview of how NFSv4.1 works with Kerberos in Qumulo Core.

Prerequisites for Joining a Qumulo Cluster to Active Directory

This section describes the prerequisites for joining a Qumulo Cluster to Active Directory for using NFSv4.1 with Kerberos.

Configuring Active Directory for Use With Kerberos

This section describes the Active Directory Domain Controller (DC) configuration changes necessary for enabling NFSv4.1 with Kerberos.

Performing Additional Cluster Configuration after Joining Active Directory

This section describes additional Qumulo cluster configuration that can affect the behavior of NFSv4.1 with Kerberos.

Using Kerberos Permissions in the Qumulo File System

This section describes how NFSv4.1 interacts with the secure file permissions that Kerberos enables for the Qumulo Core file system.

Configuring a Linux Client for NFSv4.1 with Kerberos

This section describes how to configure a Linux client for using NFSv4.1 with Kerberos.

Configuring Cross-Domain Active Directory Trusts

This section describes how the configuration of cross-domain Active Directory (AD) trusts supports NFSv4.1 with Kerberos.

Troubleshooting NFSv4.1 with Kerberos

This section describes common troubleshooting procedures for configuring NFSv4.1 to work with Kerberos.

SMB

Creating and Managing an SMB Share in Qumulo Core

This section explains how to create, modify, and delete an SMB share by using the Qumulo Core Web UI.

S3 API

Configuring and Using the S3 API in Qumulo Core

This section explains how to configure and get started working with the S3 API. This API lets clients and applications interact with the Qumulo file system natively, by using the [Amazon S3 API](#).

Creating and Managing S3 Access Keys in Qumulo Core

This section explains how to create and manage credentials that S3 API actions in Qumulo Core require to access file system resources, such as access key pairs that sign requests.

Creating and Managing S3 Buckets in Qumulo Core

This section explains how to create and manage S3 buckets for a Qumulo cluster. These buckets expose a part of your Qumulo file system to applications that use the [Amazon S3 API](#).

Managing Access to S3 Buckets in a Qumulo Cluster

This section explains how to manage access to S3 buckets in a Qumulo cluster.

Managing Access Policies for S3 Buckets in a Qumulo Cluster

This section explains how to manage access policies for S3 buckets in a Qumulo cluster.

Managing Multipart S3 Uploads in Qumulo Core

This section explains how multipart S3 uploads affect usable capacity on a Qumulo cluster and how to abort and clean up multipart uploads manually or automatically.

Supported Functionality and Known Limits for S3 in Qumulo Core

This section documents Qumulo Core support for S3 API functionality and S3 API limits.

Monitoring and Metrics

Enabling Cloud-Based Monitoring and Remote Support

This section explains how to enable Cloud-Based Monitoring and Remote Support for your Qumulo cluster.

⚠ Important

To let the Qumulo Care team provide fast support when you need it most, we strongly recommend enabling both Cloud-Based Monitoring and Remote Support.

How Cloud-Based Monitoring Works

[Enabling Cloud-Based Monitoring \(page 124\)](#) lets the Qumulo Care team monitor your Qumulo cluster proactively.

⚠ Important

Cloud-Based Monitoring *doesn't* collect file names, path names, client IP addresses, or account credentials.

Qumulo Care Response Times

We use a proprietary application that aggregates diagnostic cluster data and alerts the Qumulo Care team if an issue arises. Depending on the issue severity and cluster state, a member of the Qumulo Care team reaches out. The following table outlines Qumulo Care response times.





Severity Level	Service Availability	Response Time	Description	Common Examples
Sev0	24×7	2 hours	<p>Business Impacting: A Qumulo cluster is offline, impacting regular business operations, with potential productivity or financial losses.</p> <div><p>⚠ Important</p><p>For Severity 0 cases, call one of our toll-free numbers (page 122) or select Severity 0 when you open a case (page 122).</p></div>	<ul style="list-style-type: none">• A Qumulo cluster is unable to form a quorum• A Qumulo cluster is unable to maintain a quorum

Severity Level	Service Availability	Response Time	Description	Common Examples
Sev1	24×7	2 hours	High Priority: A Qumulo cluster is operational. However, a node is offline or the cluster experiences a severe performance degradation.	<ul style="list-style-type: none"> • A node in a Qumulo cluster is offline. • Multiple business units experience degraded performance.
Sev2	24×5	2 hours	Normal Priority: A Qumulo cluster and the Qumulo Core software are operational. However, an issue with the cluster or moderate performance degradation causes applications to operate suboptimally.	<ul style="list-style-type: none"> • CPU cores recover to normal operation • Drive failures (within expected parameters) • Inconsistent performance issues
Sev3	24×5	6 hours	Low Priority: A Qumulo cluster or the Qumulo Core software experiences an issue, cosmetic UI defect, or minor performance degradation that has minimal or negligible impact on a production system or regular business operations.	<ul style="list-style-type: none"> • PSU failure • Loss of communication with Cloud-Based Monitoring • Upgrade issues

Severity Level	Service Availability	Response Time	Description	Common Examples
Sev4	24x5	6 hours	Informational: Informal inquiries about product functionality	<ul style="list-style-type: none"> • Questions about Qumulo Core configuration • Documentation requests • Qumulo Care Slack channel access permissions

Ways to Get Help

The Qumulo Care team is always here to help you. You can contact us by using any of the following ways.

-  Browse the [Qumulo Knowledge Base](#)
-  Open a case by [filing a request](#) or [emailing us](#).
-  [Message us on Slack](#) (if you are a current customer). For more information, see our [Slack Tips and Tricks](#).
-  Call one of our toll-free numbers:
 - North America: [+1 855-577-7544 \(page 0\)](#)
 - United Kingdom: [+44 808-164-6656 \(page 0\)](#)
 - Germany: [+49 800-000-7047 \(page 0\)](#)
 - Australia: [+61 1800-954-952 \(page 0\)](#)
 - U.S. Government Customers: [+1 844-962-3777 \(page 0\)](#)

How Remote Support Works

[Enabling Remote Support \(page 125\)](#) lets the Qumulo Care team access your Qumulo cluster solely to assist you with a software update or perform diagnostics or troubleshooting on your cluster from the command line.

When you install VPN keys in the `/etc/openvpn` directory, an authorized member of the Qumulo Care team uses SSH to connect to the `ep1.qumulo.com` server and then uses SSH through a secure VPN connection to connect to your cluster (normally, this VPN connection is closed).

By default, the VPN tunnel remains open for four hours to allow members of the Qumulo Care team to perform operations such as uploading logs to `monitor.qumulo.com` or to a secured Amazon S3 bucket and sending diagnostic data to a private Amazon EC2 instance for analysis.

Note

Currently, Qumulo Core doesn't support VPN connections with IPv6.

You can configure the connection period and enable or disable Remote Support at any time.

What Data Gets Sent to Qumulo

Cloud-Based Monitoring and Remote Support let your cluster send the following detailed diagnostic data to Qumulo through an encrypted connection.

- Cluster name
- Number of nodes in cluster
- Hardware and software incidents
 - Drives
 - CRC errors
 - S.M.A.R.T. status alerts
 - Capacity triggers
 - Nodes
 - PSU failure
 - Fan failure
 - Recused node
 - Offline node
 - Unreachable cluster
 - Qumulo Core
 - New process core dump
- Configuration data (such as users, groups, SMB shares, and NFS exports)
- Logs, stack traces, and code dumps

Prerequisites

Before you can use Cloud-Based monitoring and Remote Support, you must:

- [Install VPN keys on your Qumulo cluster \(page 35\)](#)
- Enable the following destination hostnames for TCP on port 443.

Hostname	Description
<code>api.nexus.qumulo.com</code>	Nexus monitoring
<code>ep1.qumulo.com</code>	Remote Support <div>⚠ Important If your organization has an intrusion detection device or a firewall that performs SSL or HTTPS deep-packet inspection, you must add an exception to the IP address that resolves to <code>ep1.qumulo.com</code>. To identify this IP address, log in to your Qumulo cluster and run the <code>nslookup ep1.qumulo.com</code> command.</div>
<code>api.missionq.qumulo.com</code>	Cloud-Based Monitoring connectivity
<code>missionq-dumps.s3.amazonaws.com</code>	Proxy forwarding
<code>monitor.qumulo.com</code>	Cloud-Based Monitoring log uploads

Enabling Cloud-Based Monitoring

You can enable Cloud-Based Monitoring by using the Web UI or `qq` CLI.

To Enable Cloud-Based Monitoring by Using the Web UI

1. Log in to the Web UI.
2. Click **Support > Qumulo Care**.
3. On the Qumulo Care page, do the following:
 - a. In the Cloud-Based Monitoring section, click **Edit**.
 - b. Click **Yes, I want Qumulo Cloud-Based Monitoring** and then click **Save**.

If your configuration is valid, the Web UI shows the status **Enabled | Connected**.

To Enable Cloud-Based Monitoring by Using the qq CLI

- To enable Cloud-Based Monitoring, run the `qq set_monitoring_conf --enabled` command.
- To disable Cloud-Based Monitoring, run the `qq set_monitoring_conf --disabled` command.
- To check the status of Cloud-Based Monitoring, run the `qq monitoring_conf` command.

Enabling Remote Support

You can enable Remote Support by using the Web UI or `qq` CLI.

To Enable Remote Support by Using the Web UI

1. Log in to the Web UI.
2. Click **Support > Qumulo Care**.
3. On the Qumulo Care page, do the following:
 - a. In the **Remote Support** section, click **Edit**.
 - b. Under **Do you want to enable Qumulo Remote Support?**, click **Yes** and then click **Save**.

If your configuration is valid, the Web UI shows the status **Enabled | Connected**.

To Enable Remote Support by Using the qq CLI

- To enable Remote Support, run the `qq set_monitoring_conf --vpn-enabled` command.
- To disable Remote Support, run the `qq set_monitoring_conf --vpn-disabled` command.
- To check the status of Remote Support, run the `qq set_monitoring_conf` command.

Connecting to Cloud-Based Monitoring and S3 by Using a Custom Proxy

This section explains how to connect to Cloud-Based Monitoring and S3 by using a custom proxy.

In Qumulo Core 2.6.4 (and higher), you can specify a custom proxy for both Cloud-Based Monitoring and S3 in different ways.

Configuring a Custom Proxy for Cloud-Based Monitoring and S3

The following examples show some common configurations for custom proxies by using the `qq set_monitoring_conf` command.

⚠ Important

Remote VPN support isn't available when you connect to Cloud-Based Monitoring by using a custom proxy.

You can specify a custom proxy hostname and port *only* for Cloud-Based Monitoring. For example:

```
qq set_monitoring_conf
--enabled
--mq-proxy-host mq-proxy.example.com
--mq-proxy-port 123
```

You can also specify a custom proxy hostname and port *only* for the S3 proxy (for the endpoint that Qumulo Core uses to store core dumps and logs). For example:

```
qq set_monitoring_conf
--enabled
--s3-proxy-host s3-proxy.example.com
--s3-proxy-port 456
```

Finally, you can specify a custom proxy hostname and port for *both* Cloud-Based Monitoring and your S3 proxy. For example:

```
qq set_monitoring_conf  
  --enabled  
  --all-proxy-host mq-s3-proxy.example.com  
  --all-proxy-port 789
```


Restoring the Default Values for Cloud-Based and Nexus Monitoring

This section explains how to set the default values for Cloud-Based Monitoring and Nexus Monitoring.

When you no longer [connect to Cloud-Based Monitoring by using a custom proxy \(page 126\)](#), you can use the `qq` CLI to restore the default values for Cloud-Based and Nexus Monitoring.

To Restore the Default Values for Cloud-Based and Nexus Monitoring by Using the qq CLI

1. Connect to your cluster by using SSH. For example:

```
ssh admin@203.0.113.0
```

2. Log in to Qumulo Core by using the administrative account. For example:

```
qq login -u admin -p mypassword
```

3. To restore the default values for Cloud-Based and Nexus Monitoring, use the `qq set_monitoring_conf` command.

In the following example, we:

- Enable Cloud-Based Monitoring (MQ)
- Enable Nexus monitoring
- Disable HTTPS for the S3 proxy
- Set the hostname and port for Cloud-Based Monitoring
- Disable the proxy for Cloud-Based Monitoring (by setting it to `0`)
- Set the monitoring polling interval to 60 seconds
- Set the hostname and port for the S3 proxy
- Specify the custom VPN for your organization, `example.qumulo.com`

```
qq set_monitoring_conf
  --enabled
  --nexus-enabled
  --s3-proxy-disable-https
  --mq-host api.missionq.qumulo.com
  --mq-port 443
  --mq-proxy-port 0
  --period 60
  --s3-proxy-host monitor.qumulo.com
  --s3-proxy-port 443
  --vpn-host example.qumulo.com
```

4. To confirm that Cloud-Based Monitoring is working correctly, log in to the Qumulo Core Web UI and then click **Support**.

If your configuration is valid, the Web UI shows the status **Enabled | Connected**.

Qumulo OpenMetrics API Specification

This section lists the names, types, labels, and descriptions for the metrics that Qumulo Core 5.3.0 (and higher) emits in OpenMetrics API format.