Qumulo

Qumulo® Secure

Software-Based Encryption

Software-defined, transparent, on-by-default, data-at-rest encryption

December, 2020

Introduction

At Qumulo, we believe that the era of unencrypted data has come to an end and that our customers will expect their data to be cryptographically protected in the coming years.

Today's enterprises require transparent, hardware-agnostic, on-by-default data security. They also need to be protected from data theft that could leave them open to liability. This becomes especially important given increasing security and compliance regulations across various industries and geographies, particularly when it comes to securing data at multi-petabyte-to-exabyte scales.

At these scales, organizations struggle to protect their data-at-rest from threats such as an attacker stealing drives directly from a node, or getting hold of decommissioned disks in the supply chain. In the absence of a solid encryption solution, enterprises resort to methods such as physically destroying disks, but even those solutions are not fully secure, and don't always meet necessary requirements of their security and compliance teams.

Also, purely hardware-based encryption solutions can limit an organization's infrastructure options and their ability to adopt innovative file data platforms like Qumulo All-NVMe storage.

Qumulo's introduction of **software-based encryption** to Qumulo Secure circumvents these challenges, giving customers confidence in their data security and offering flexibility in hardware choice with a fully transparent experience — included for free as a part of Qumulo Core®.

With the Qumulo file data platform, data-in-flight is encrypted with our file access protocol support (SMBv3) and replication features. With the addition of Qumulo software-based encryption, customers can further strengthen their security profile with complete data encryption — both in transit and at-rest. There is no longer a need to worry about potential bad actors reading data from stolen disks.

Qumulo is not only providing a hardware-independent approach to support a truly software-defined storage product, but also offering the only fully encrypted All-NVMe products on the market.

This whitepaper provides an overview of the Qumulo software-based encryption solution targeted to solve a range of security-critical gaps of the modern-day enterprise file system to give security-sensitive customers the confidence that their data is protected against various threat vectors, transparently, and at the highest possible standards expected of the enterprise.

Key Benefits

The highest standards of encryption - for free.
Qumulo software-based encryption leverages the most rigorous AES-256 bit encryption standards for enterprise organizations. It is included in Qumulo Core free of charge.

Protection from physical theft of disks.
Whether a disk is stolen from the cluster itself, or obtained through the supply chain after being decommissioned, Qumulo software-based encryption provides physical protection against malicious actors regardless of access vector.

No hardware restrictions. Ever.
Software-defined, hardware agnostic encryption that is completely transparent to the end-user. There is no hardware controller, disk, or chip dependency. You choose the hardware, we do the rest.

Integrated, fully transparent key-management.
Nothing to set up. Nothing to manage. Easy.

Software-Based Encryption-at-Rest Definition

Data-at-rest refers to inactive data that has been physically stored on persistent storage of some type. The security concerns of data-at-rest differ from that of data-in-transit across a network, data sitting in RAM, or even data being actively processed.

Data that is actively being processed by applications is often data that must be reassembled and is not guaranteed to be in its complete state. Data-at-rest can be vulnerable in the event an entire disk is physically stolen by an attacker or when encountering a malicious actor in the supply chain after a disk has been decommissioned.

Even though the reassembly of unencrypted data is often not a straightforward process with Qumulo as data is striped across multiple nodes and drives, it is important data security is not left to chance.

Qumulo Software-Based Encryption Implementation

Qumulo **software-based encryption** is a software-based, data-at-rest encryption solution first introduced in Qumulo Core v3.1.5. Software-based encryption is on by default on all newly created clusters starting with this release. This means encryption is enabled at cluster-create time to guarantee there is no potential of unencrypted data or metadata living on the disks within the cluster.

Software-based encryption leverages a multi-layered encryption key system that ensures data is protected against bad actors even if they have physical access to the disk themselves. Because the encryption mechanisms are completely in-software within our Qumulo Secure cryptographic module, all on-prem node types offered by Qumulo support software-based encryption. Customers can now enjoy enterprise-grade encryption no matter which hardware they choose.

Qumulo has based its implementation of encryption on rigorous AES-256 bit security standards expected of the enterprise. To boost performance and security, Qumulo Secure leverages the AES-NI extension to the x86 instruction set on each node's CPUs. Encryption is done at the block level below the file system as shown in Figure 1 below, where all data and metadata to be encrypted, including the encryption keys themselves, are sent through our Qumulo Secure cryptographic module.

This bounded cryptographic module is where the encryption transformation takes place. And in order to reduce the amount of data to be encrypted, all of this is done above the Qumulo protection system layer where erasure coding is done. Error correction is applied to encrypted data instead of encrypting the expanded error corrected data, which means software-based encryption naturally encrypts less-than-pure disk encryption.

Software-based encryption encrypts the data using a two-key system for added security. A Key Encrypting Key (KEK) and the Data Encrypting Key (DEK). The KEK wraps the DEK which means that in order to decrypt the DEK and encrypt/decrypt data, you must first have the KEK. There is one KEK per cluster that can be rotated on-demand via the qq API, and one DEK per cluster that never changes. The KEK is stored on the boot drive of every node within the cluster while the wrapped DEK is stored redundantly on the data SSDs inside the cluster’s Distributed Key Value (DKV) store that Qumulo Core uses to interpret the data drives. After the DEK is unwrapped by the KEK, it is then pulled into memory where it can be used to encrypt and decrypt in-flight data.

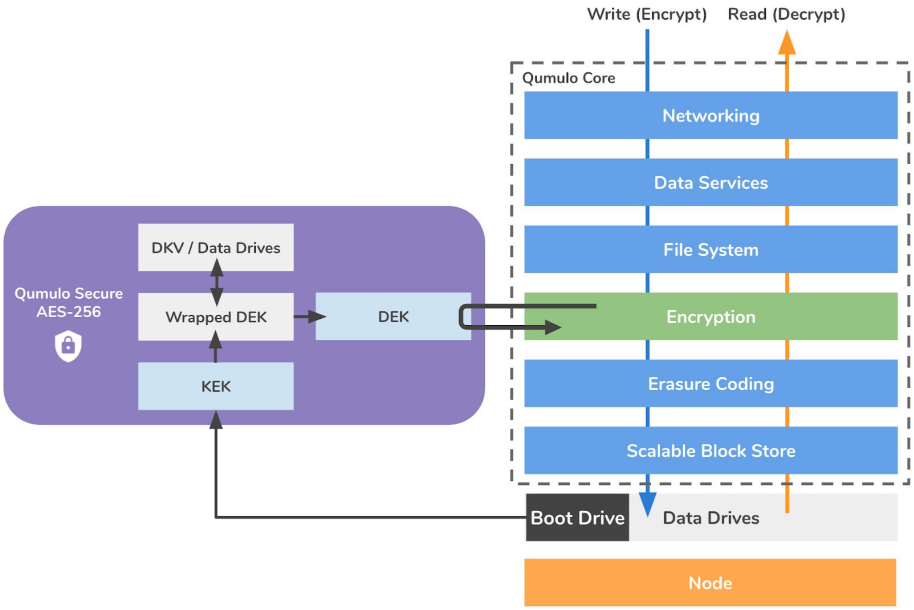


Figure 1. *This diagram shows the life of a write and read when being encrypted/decrypted on the Qumulo file data platform.*

Internal Key Management

Qumulo software-based encryption key management is lightweight and completely transparent to the end user. This is by design. We believe that if encryption is to be an inherent part of the enterprise file system, then it should not compromise overall simplicity. Because of this, the software-based encryption key management is entirely internal and securely handled by the file data platform itself.

Key Encrypting Key (KEK)

Upon cluster formation, the KEK is created using OpenSSL's default random number generator within the Qumulo Secure cryptographic module. The key is then distributed to all other nodes' boot drives within the cluster. During this process, a quorum “leader node” records and sends the newly created KEK to each remaining node within the cluster which is then written to each node's boot drive. This redundancy protects the cluster from KEK loss in the case a boot drive fails, is corrupted, or a new node is added to the cluster.

In these cases the KEK will be redistributed to all nodes within the cluster during the next quorum event such as drive swaps, upgrades, or network drops to ensure consistency across the cluster.

And at the same time during cluster formation, the DEK will be encrypted by the KEK and stored in the DKV. If required, the KEK can be manually rotated via the REST API.

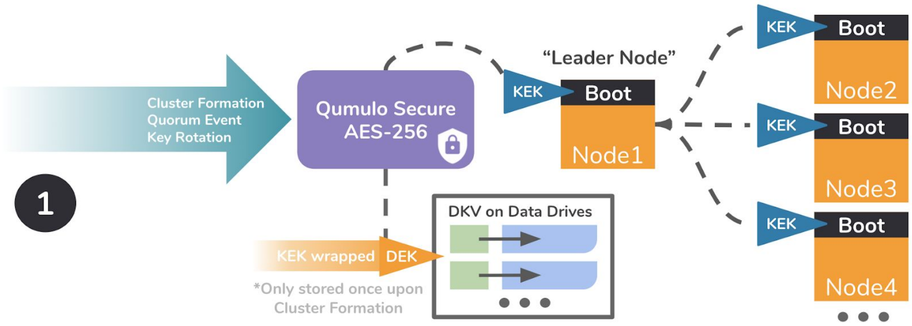


Figure 2. Upon cluster formation, a quorum event or key rotation, the "leader node" will distribute the Qumulo Secure-generated KEK to the boot drive of each node. The DEK is wrapped and stored in the Distributed Key Value (DKV) store once upon cluster formation.

Data Encrypting Key (DEK)

The DEK is created at cluster creation time, wrapped by the KEK and stored in the DKV. Once created, the DEK never changes and cannot be changed externally. The encrypted DEK is read from the DKV at quorum start, is unwrapped by the KEK and then stored in memory. It's also read from the DKV after a key rotation, replacing the old unencrypted DEK in memory for consistency. It then stays in memory until the end of the quorum or the next key rotation to encrypt/decrypt data. To protect against DEK memory leakage, the DEK is specifically marked never to be included in core dumps, should they take place.

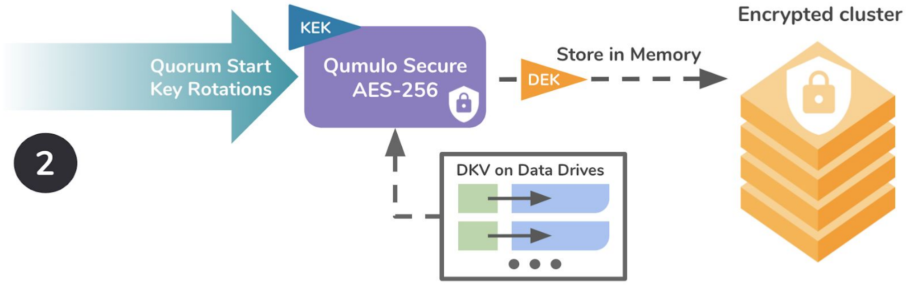
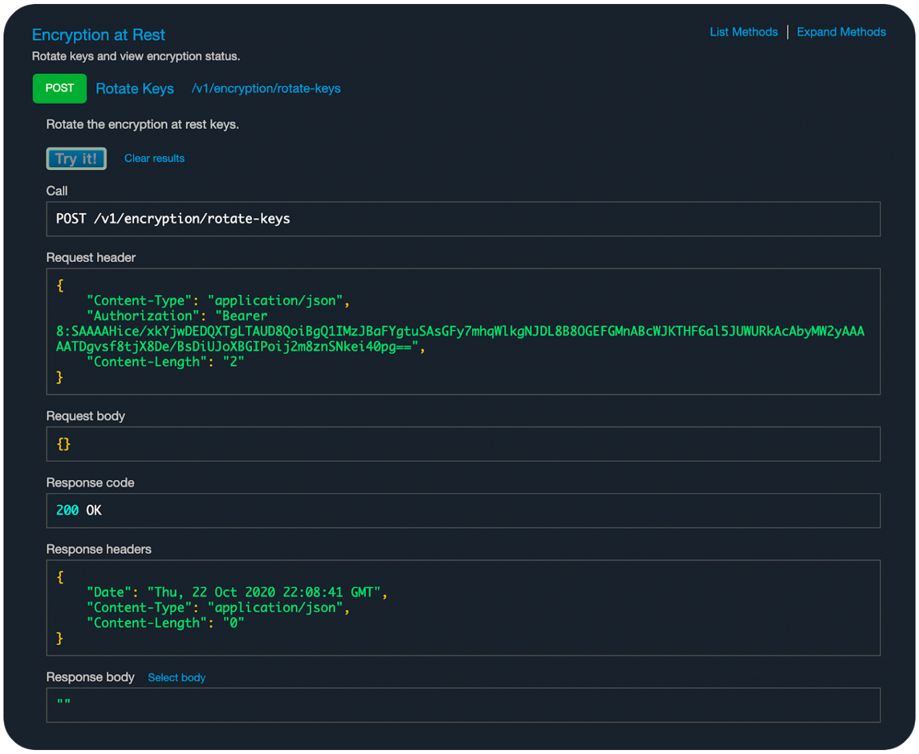


Figure 3. At quorum start or after a KEK rotation, the encrypted DEK is decrypted through Qumulo Secure using the KEK which is then stored in memory in order to encrypt and decrypt data.

Encryption Key Rotation

Though internal key management is transparent, some customers would like a way to rotate the KEK to meet requirements from their security and compliance teams. You can leverage the qq rotate_encryption_keys CLI command to rotate the KEK, which is also available in the GUI using our interactive API under the “APIs & Tools” page:

```
qq rotate_encryption_keys
```



Encryption at Rest

Rotate keys and view encryption status.

Rotate the encryption at rest keys.

Call

```
POST /v1/encryption/rotate-keys
```

Request header

```
{
    "Content-Type": "application/json",
    "Authorization": "Bearer
8:SAAAAHice/xkYjwDEDQXTgLTAUD8QoiBgQ1IMzJBaFYgtuSAsGFy7mhqWlkgNJDL8B8OGEFGMnABcWJKTHF6al5JUWURkAcAbyMW2yAAA
AATDgvsf8tjX8De/BsDiUJoXBGIPoij2m8znSNkei40pg==",
    "Content-Length": "2"
}
```

Request body

```
{}
```

Response code

```
200 OK
```

Response headers

```
{
    "Date": "Thu, 22 Oct 2020 22:08:41 GMT",
    "Content-Type": "application/json",
    "Content-Length": "0"
}
```

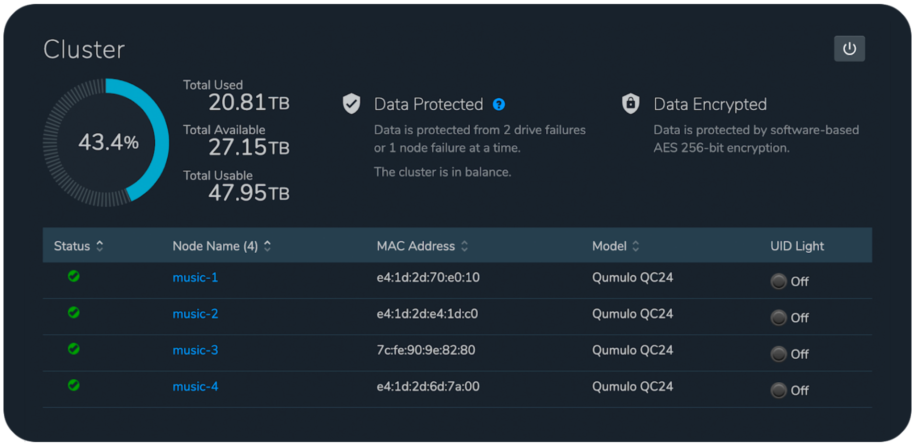
Response body

```
""
```

Visual Interface for Software-Based Encryption at Rest

Even though all new clusters created at Qumulo Core v3.1.5 or later are encrypted with software-based encryption, it's easy to confirm whether your cluster is encrypted via our visual interface.

Under the Cluster → Overview page, you can confirm your cluster is indeed encrypted as seen in the screenshot:



Cluster

43.4%

Total Used
20.81TB

Total Available
27.15TB

Total Usable
47.95TB

Data Protected

Data is protected from 2 drive failures or 1 node failure at a time.

The cluster is in balance.

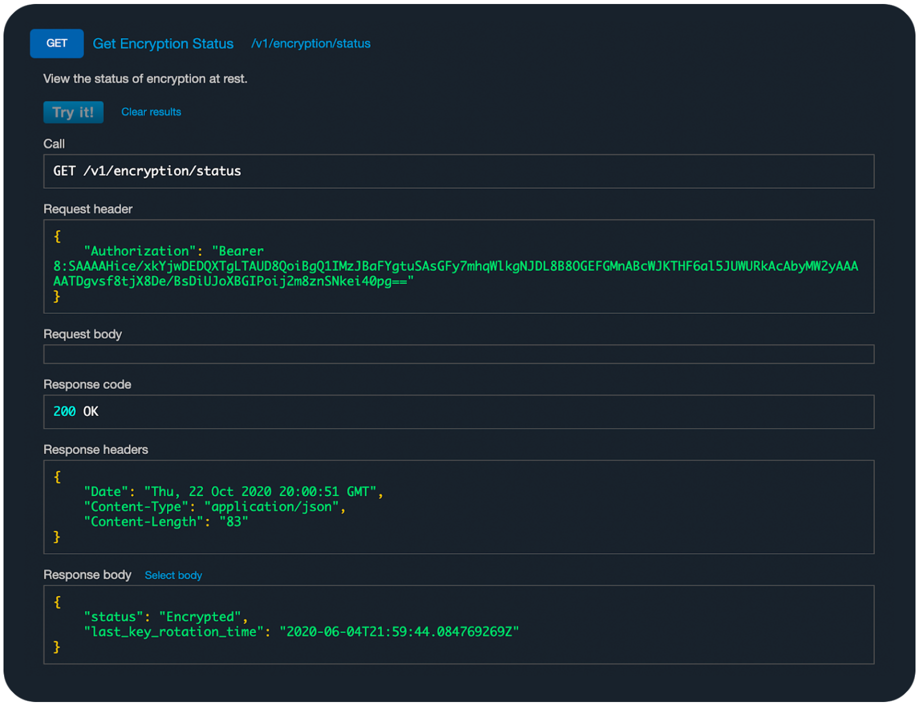
Data Encrypted

Data is protected by software-based AES 256-bit encryption.

Status	Node Name (4)	MAC Address	Model	UID Light
✓	music-1	e4:1d:2d:70:e0:10	Qumulo QC24	● Off
✓	music-2	e4:1d:2d:e4:1d:c0	Qumulo QC24	● Off
✓	music-3	7c:fe:90:9e:82:80	Qumulo QC24	● Off
✓	music-4	e4:1d:2d:6d:7a:00	Qumulo QC24	● Off

Or leverage the qq encryption_get_status CLI command which is also available in the visual interface using our interactive API tool under the "APIs & Tools" page:

```
qq encryption_get_status
```



View the status of encryption at rest.

Call

```
GET /v1/encryption/status
```

Request header

```
{
    "Authorization": "Bearer
8:SAAAAHice/xkYjwDEDQXTgLTAUD8QoiBgQ1IMzJBaFYgtuSAsGFy7mhqWlkgNJDL8B8OGEFGMnABcWJKTHF6al5JUWURkAcAbyMW2yAAA
AATDgvsf8tjX8De/BsDiUJoXBGIPoij2m8znSNkei40pg=="
}
```

Request body

Response code

```
200 OK
```

Response headers

```
{
    "Date": "Thu, 22 Oct 2020 20:00:51 GMT",
    "Content-Type": "application/json",
    "Content-Length": "83"
}
```

Response body

```
{
    "status": "Encrypted",
    "last_key_rotation_time": "2020-06-04T21:59:44.084769269Z"
}
```

Qumulo Secure

The Qumulo Secure cryptographic module within Qumulo Core defines a clear security boundary around all the places within the software that uses cryptography to secure data that is stored on disk, data in flight, and data access.

This boundary is defined and implemented in the form of a self-contained, version-controlled cryptographic module that is currently listed as 'Implementation Under Test' with NIST for FIPS 140-2, where federal certification is expected in 2021 after third party lab certification is complete. The boundary is responsible for securely fielding read and write requests/traffic for encryption at-rest, HTTPS, FTPS, replication, SMBv3, and LDAPS. This technology supports secure traffic such as replication between Qumulo clusters or S3 (Qumulo Shift) via TLS, FTP via TLS, access security with role-based access (RBAC) / Active Directory / LDAP servers for authentication and identity information, and REST access via HTTPS that underpins communication with the visual interface. Qumulo Secure is the backbone of the software-based encryption technology.

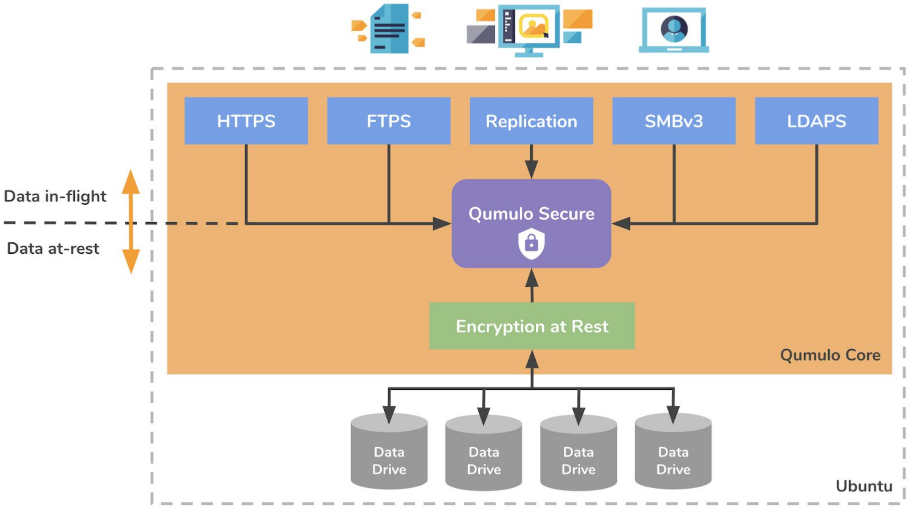


__Figure 4.__ The Qumulo Secure cryptographic module boundary is defined strictly around OpenSSL 1.1.1, which is installed into our redistributed Ubuntu image. Our custom implementation of AES-XTS is optimized for typical Qumulo workloads. The OpenSSL version of AES-XTS is not exposed outside of Qumulo Secure. OpenSSL's libraries are dynamically linked with the Qumulo daemon, into which the code is loaded and runs as part of the Qumulo Core process start time.

Compliance Validations

Qumulo is committed to supporting the most security-sensitive customers in industries such as public sector and federal agencies, financial services, healthcare and universities. Qumulo software-based encryption is an important step in completing required FIPS-2 and Common Criteria certifications required by organizations that demand standardized, hardened, and tested storage solutions that meet government security compliance regulations. Software-based encryption is made possible by our bounded Qumulo Secure cryptographic module which is currently listed as 'Implementation Under Test' with NIST for FIPS 140-2.

Encryption and Replication

Over the wire encryption is completely separate from encryption at-rest. Though they are both serviced from the Qumulo Secure module, they are not dependent on each other. Today, replication between Qumulo clusters, or replication from Qumulo to Amazon S3 (Qumulo Shift), is secured via TLS over the wire. Qumulo also offers fully encrypted SMBv3 client connectivity. If you combine this with Qumulo software-based encryption, a truly end-to-end encrypted solution emerges.

There are no restrictions on Qumulo-to-Qumulo replication regardless of whether the source or target is using software-based encryption or not. If a cluster is software encrypted at-rest, it will be written to as encrypted during a transfer. In order to meet security requirements of keeping all the clusters encrypted, customers are recommended to keep both source and target clusters encrypted and set up replication across them.

Replication transfer configurations that are supported are:

Cluster A →	Cluster B
Encrypted	Encrypted
Encrypted	Unencrypted
Unencrypted	Encrypted
Encrypted	Unencrypted-cloud
Unencrypted-cloud	Encrypted
Encrypted	Amazon S3 (Qumulo Shift)

Encryption in the Cloud

Qumulo Core software-based encryption is not currently available with our Qumulo for Amazon Web Services or Qumulo for Google Cloud Platform cloud images today. It is only available with on-prem deployments. If you require encryption in the cloud as well, the backing media for Amazon Web Services or Google Cloud Platform offer AES-256 bit encryption solutions that are either on by default or can be enabled during cluster configuration.