Security of Data in the ActiveScale™ Cloud Object Storage System

With any cloud deployment, whether it's public, private or hybrid, security of data is paramount. Quantum's ActiveScale is an extremely secure and efficient cloud object storage system with a comprehensive suite of security features for both data-at-rest and data-in-flight.

Security of Data at Rest

ActiveScale cloud object storage systems support system-wide and object-level encryption for data-at-rest. Encryption is configurable at the object level or system-wide, so the entire array need not be encrypted if it is not required. The default condition is encryption turned off. Encryption can be enabled by the storage administrator if desired. With encryption enabled, all objects are encrypted before they are erasure coded and stored on disk. ActiveScale cloud object storage systems use the symmetric cipher AES 256-CTR (256-bit Advanced Encryption Standard in Counter mode) to encrypt objects. The input consists of plain text data, a 256-bit encryption key, and a 128-bit initialization vector. The output consists of cipher text data with the same size as the plain text data.

ActiveScale cloud object storage systems encrypt both **data** and **metadata, including custom metadata:**

- To encrypt **data**, ActiveScale uses a 256-bit encryption key and one or more unique initialization vectors. One encryption key is generated per object, and it calculates initialization vectors based on the Globally Unique object IDentifier (GUID). This identifier is a deterministic hash of the object. The encryption process involves encrypting the data first, then erasure coding it.
- To encrypt **metadata**, ActiveScale uses a 256-bit key and a random* initialization vector. The encryption key can be provided by the storage administrator or randomly generated per object, and both the key and the vector are stored as metadata, as well. They are encrypted using the user-provided master key and the same AES-256/CTR encryption method as described above. Even though the randomly generated initialization vector is unencrypted when stored in the object metadata, the metadata itself is encrypted.

Creating a Master Key

A master key is created by specifying a master password. This master key must be created before enabling encryption. The ActiveScale system uses a user-specified master password to generate the master key and a unique encryption key for each object that has encryption enabled. The master key is used to encrypt each of these uniquely generated encryption keys.

Replacing or Revoking an Encryption Key

It is possible to replace an encryption key. When the encryption key is replaced, only new data uploaded to the system is encrypted with the new key. Existing data in the system remains encrypted with the original encryption key.

When a system's encryption key is revoked, the key is no longer used to encrypt data in the system. In other words, new data uploaded to the system is no longer encrypted. However, existing data in the system does not change and remains encrypted with the revoked key. At this point, if the encrypted data that remains needs to be decrypted, it must be rewritten.

Backing Up the Encryption Key Database

The encryption key database stores all encryption keys and encryption policies. It is an environment MetaStore, and like all MetaStores it is distributed over three Controller Nodes. Since the object content encryption key must be stored in the object metadata, the encryption key, itself, is encrypted with a separate metadata encryption key.

Follow a best practice by backing up the encryption key database whenever encryption is enabled on a system or whenever a new encryption key is generated. Remember that a new key is generated for each object created by using the master key.

Encryption Key Management

There is no encryption key management for stored data. The master key is loaded into the encoding or decoding process memory on every Controller Node when the daemon is started. Each object can be encrypted with its own key, which is stored in the database and encrypted with the master key.

Security of Data In-flight

By default, Quantum ActiveScale systems are configured to use the HTTPS protocol to secure data while in transit from client systems. The HTTPS protocol uses one of two secure protocols to encrypt communications: SSL (Secure Sockets Layer) or TLS (Transport Layer Security). Both protocols use a highly secure asymmetric Public Key Infrastructure (PKI) system. An asymmetric system uses two keys to encrypt communications" a public key and a private key. Anything encrypted with the public key can be decrypted by the private key only, and often it will use the assistance of a third-party certificate authority as a verification agent to ensure that the keys in use are legitimate.

*Note: "random" / "randomly generated" means that a pseudo-random number generator seeded with a high-quality seed was used to obtain cryptographically strong random numbers.

To allow for a wide degree of customer flexibility, ActiveScale enables users to dial in the appropriate level of transport security and supports the ability to selectively activate HTTPS, provide their own PKI certificates and certificate rotation policy, and customize which cyphers will be supported, along with their order of priority.

While protocols like HTTPS are highly secure, it is essential to protect the data from the possibility of eavesdropping on the transaction to record the exchange, perform offline analysis on the data, alter the package, and re-submit it to data stream at a later date. This eavesdropping is called a "Replay Attack."

It is extremely important that all systems be synchronized using an accurate NTP source. The reason is that every read/write request received by the ActiveScale cloud object storage includes a time stamp, which is part of the structure of the S3 authentication signature. If the receiving ActiveScale system receives a request that has a time stamp that is more than a few minutes out of sync with the client, it rejects the request as invalid. This can lead to false rejections due to the client's system clock being out of sync the ActiveScale cloud object storage system.

Additional Security

ActiveScale cloud object storage systems have an additional security feature that enhances it as a trusted tool. A digital signature is affixed to the object metadata when it is first created, and the system permanently stamps the information with a unique seal identifying the contents of the object. If the data at rest were altered through a malicious action or through something innocuous, such as bit rot, the signature would no longer match the contents.

For the life of the object, both the sender and the recipient have a way of proving that the contents have remained intact and unchanged since the data was created. This eliminates concern by either party that the data was tampered with throughout the data transfer and/or storage process. ActiveScale cloud object storage systems use this same validation method to periodically scan the data entrusted to it to ensure that the data remains consistent, and it automatically corrects errors that may creep in, contributing to its world-class 19 nines of data durability.

To learn more, visit www.quantum.com/objectstorage.