

Hedvig Encrypt360 User Guide

Table of Contents

Introduction to Hedvig Encrypt360	3
Editing KMS Configurations	4
Editing KMS configurations with the Hedvig WebUI	4
Editing KMS configurations with the Hedvig CLI	8
AWS as the KMS.....	8
KMIP as the KMS.....	8
Creating a Hedvig virtual disk with the Encrypted option	9
Encrypt360 Details	10
Workflow	10
Cluster Installation	10
Volume Creation	10
Writes	11
Reads.....	11
Performance Optimization.....	11
Glossary	12

Introduction to Hedvig Encrypt360

Hedvig provides software-based encryption with *Encrypt360*.

This feature enables encryption of data at the point of ingestion (on the Hedvig Storage Proxy server).

Data encrypted with Encrypt360 remains protected:

- in flight, between the Hedvig Storage Proxy and Hedvig Storage Cluster Nodes,
- in flight, between the Hedvig Storage Cluster Nodes (or sites) as part of replication,
- in-use, at the Hedvig Storage Proxy, and
- at rest.

Hedvig provides AES (Advanced Encryption Standard) encryption in the XTS (XEX-based tweaked-codebook mode with ciphertext stealing) disk encryption cipher. Any third-party KMS (key management system) can be plugged in to alleviate key management concerns.

Hedvig can create encrypted volumes from the Hedvig vCenter Plugin, OpenStack Cinder setup, or Docker setup.

Hedvig provides end-to-end encryption with support for AWS and any other KMIP v1.2+ compliant KMS, like KeySecure, Fortanix, etc. For a given cluster, Hedvig can talk to multiple KMS vendors at a given time, per tenancy, based on customer preference.

Hedvig has an inbuilt KMIP (key management interoperability protocol) client running on the cluster. It can communicate with any KMS that is KMIP v1.2+ compliant. Some KMIP-compliant KMS examples include Fortanix, KeySecure.

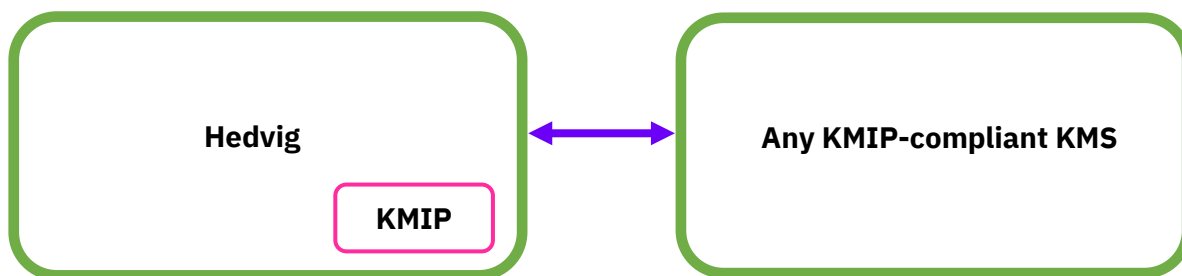


Figure 1: Hedvig - KMS Communication

Editing KMS Configurations

The process of editing a KMS configuration is the same for either a cluster that has been newly installed or a cluster that has had an NDU (non-disruptive upgrade) to accommodate encryption.

Any changes that you make to a KMS configuration are applied to all nodes in the cluster.

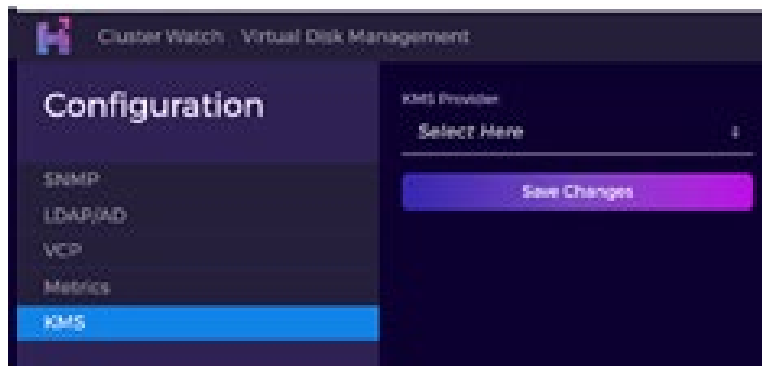
There are two ways to edit KMS configurations:

- [Editing KMS configurations with the Hedvig WebUI](#)
- [Editing KMS configurations with the Hedvig CLI](#)

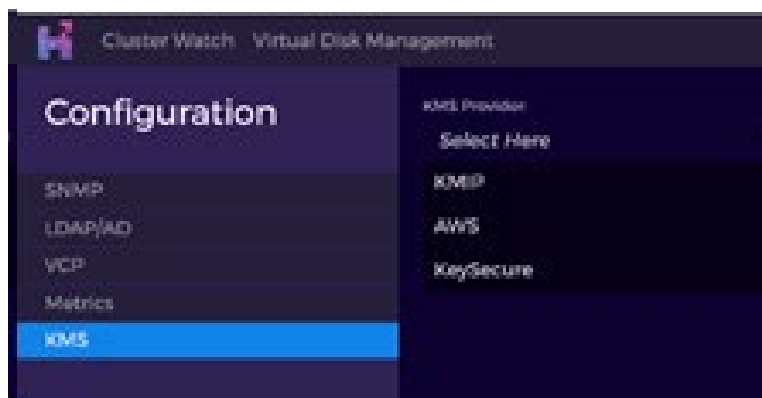
Editing KMS configurations with the Hedvig WebUI

If you are a Super User or a Power User, you can edit KMS configurations with the Hedvig WebUI. Otherwise, you must use the Hedvig CLI (see [Editing KMS configurations with the Hedvig CLI](#)).

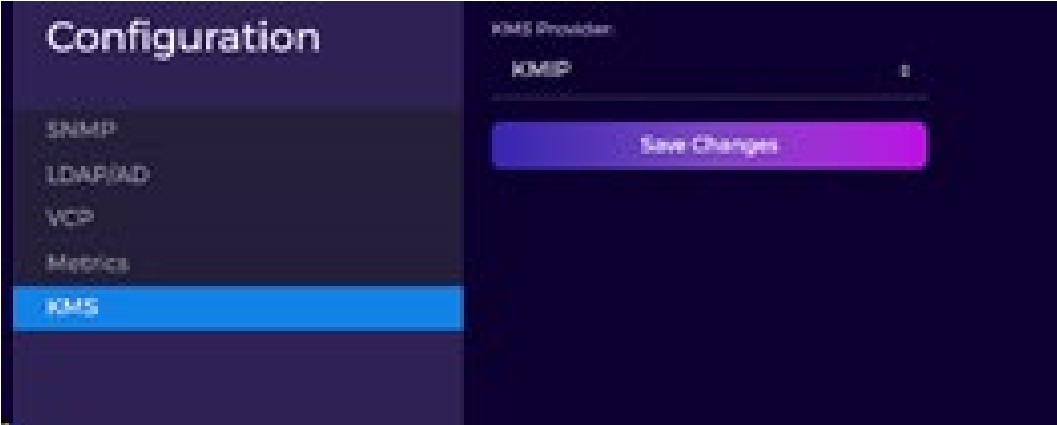
1. Select Configuration > KMS.



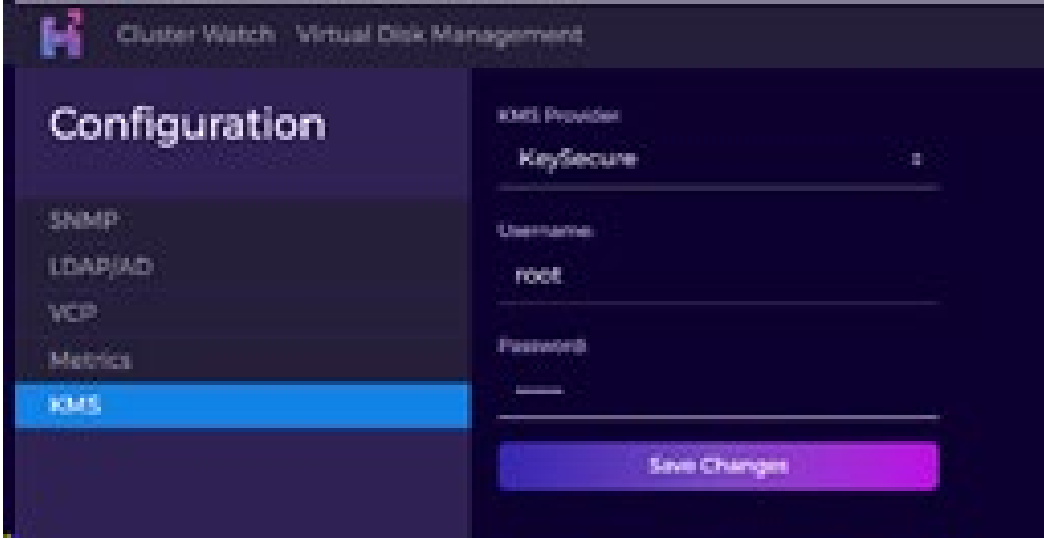
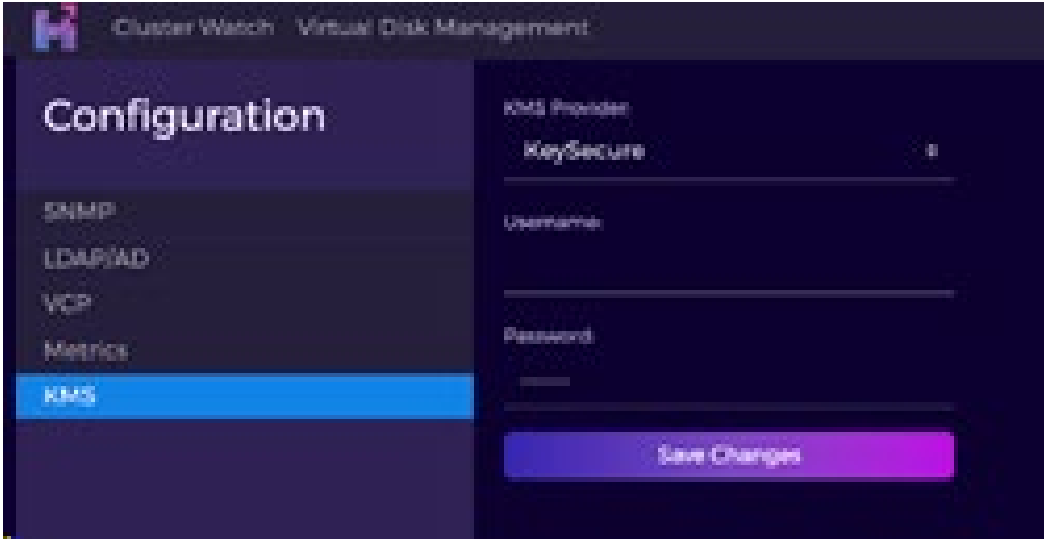
2. Select the KMS Provider.



3. For KMIP:

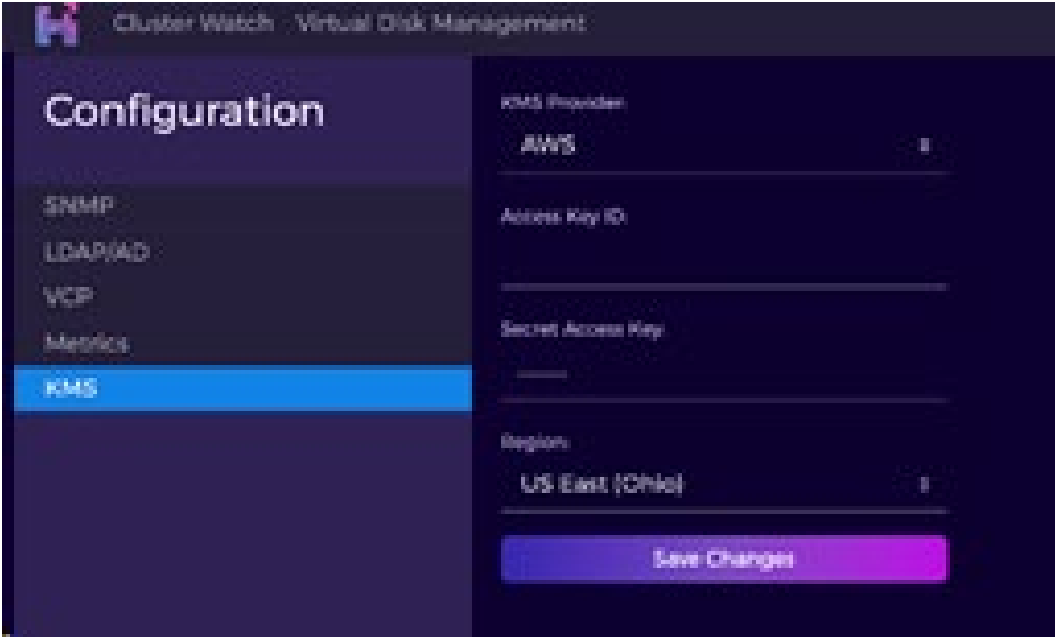


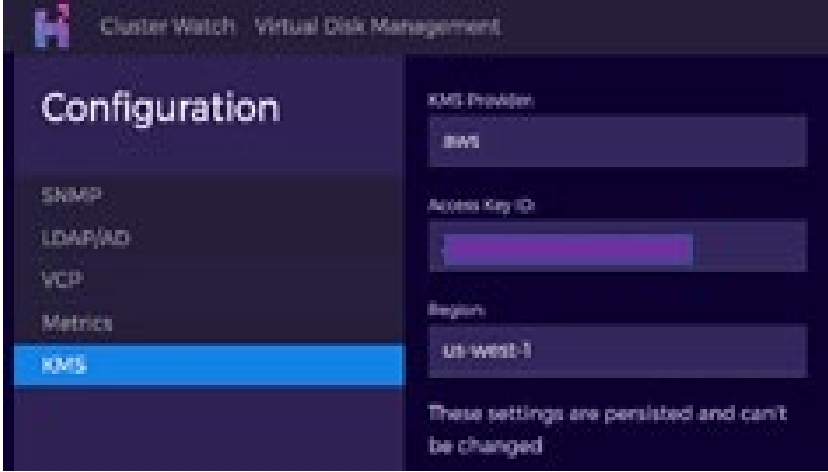
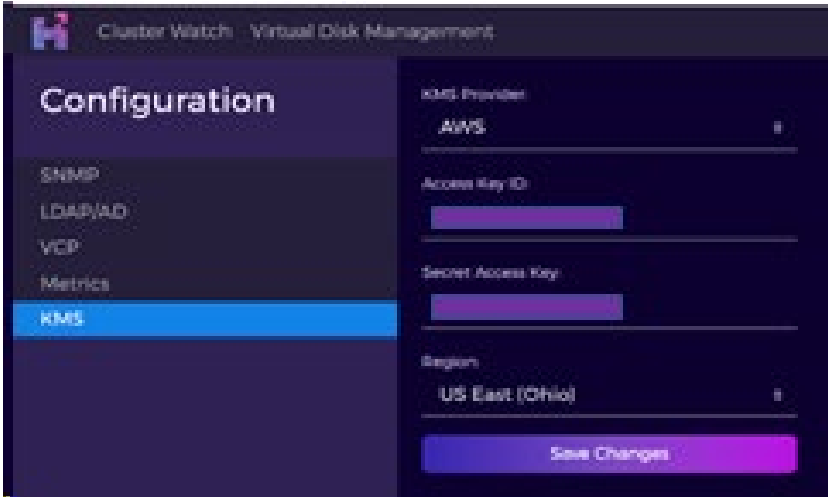
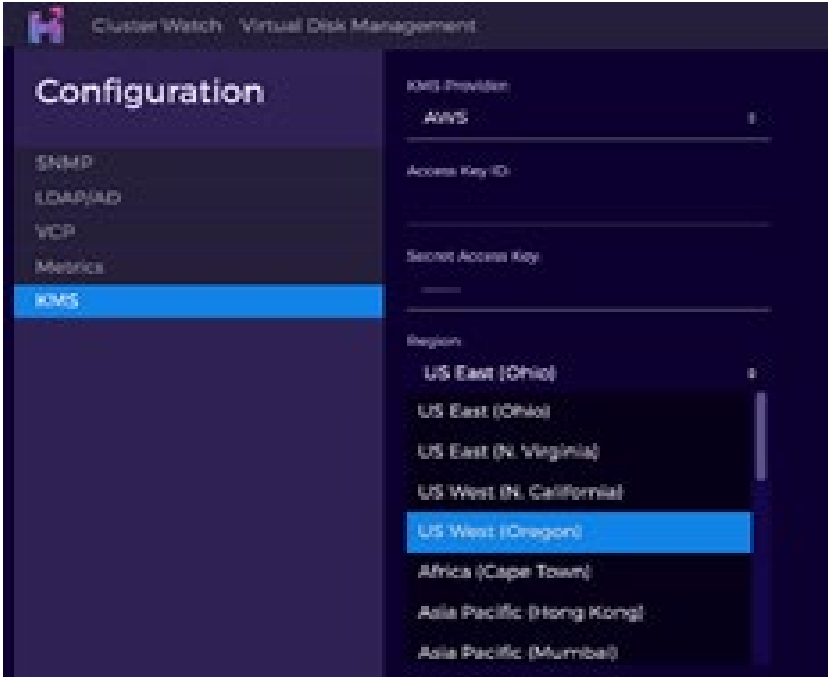
4. For KeySecure:



5. For AWS, complete all of the fields:

form field	description
KMS Provider	AWS
Access Key ID	Your AWS Access Key ID, used as part of the credentials to authenticate the user.
Secret Access Key	Your AWS Secret Access key, used as part of the credentials to authenticate the user.
Region	The AWS Region that will be used to accessing AWS services. For best performance, choose a region that is the closest geographically.





Editing KMS configurations with the Hedvig CLI

From the Hedvig CLI, run the following command:

```
setupkms -k <kms_name> -l <location> -u <user> -p <password>
-t <tenant_name>
```

The parameters vary, depending on whether you are using AWS or KMIP.

AWS as the KMS

If you are using AWS as the KMS, here are the input parameter details. All of these parameters are **required**.

- `provider = aws`
- `location = AWS_REGION`

This is the AWS Region that will be used for accessing AWS services. For best performance, choose a region that is the closest geographically. See <https://docs.aws.amazon.com/general/latest/gr/rande.html#regional-endpoints>.

- `username = AWS_ACCESS_KEY_ID`

This is your AWS Access Key ID, which is used as part of the credentials to authenticate the user.

- `password = AWS_SECRET_ACCESS_KEY`

This is your AWS Secret Access key, which is used as part of the credentials to authenticate the user.

- `tenant_name = Hedvig tenant name`

KMIP as the KMS

If you are using the KMIP protocol to integrate with any KMS, here are the input parameter details:

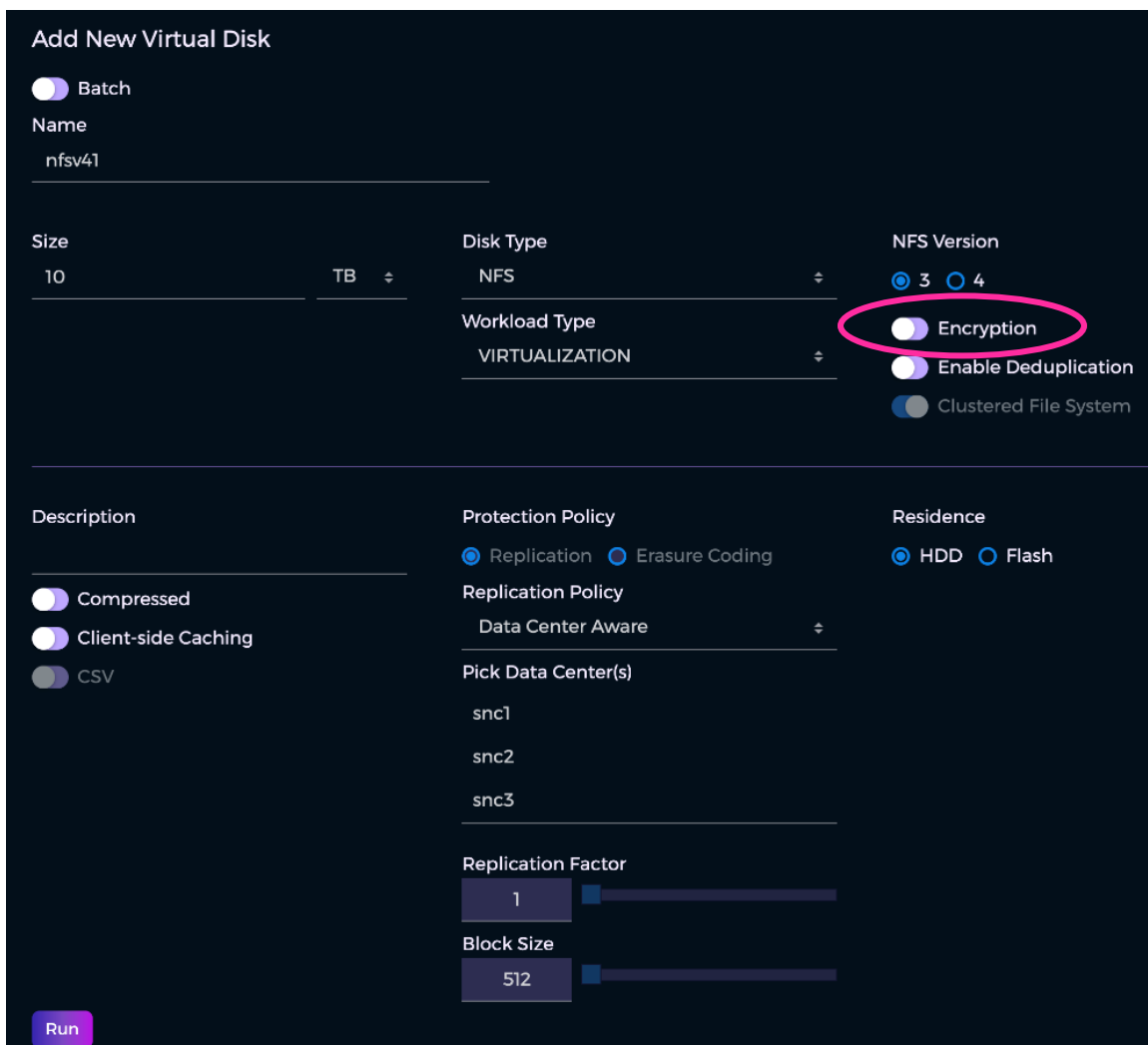
- `provider = kmip`
- `tenant_name = Hedvig tenant name`

Note: Hedvig stores only *references* to encryption keys and never persists the original encryption keys anywhere. These references are used to generate encrypted keys for every volume.

Creating a Hedvig virtual disk with the Encrypted option

There are three options for creating a Hedvig virtual disk with the Encrypted option:

- Hedvig CLI: Run the `mkvdisk` command, using the `-i` option to mark the volume encrypted.
- Hedvig WebUI: When you create the Hedvig virtual disk, select **Encryption**.
- Hedvig REST API: When you create the Hedvig virtual disk, include the `encryption` parameter.



The screenshot shows the 'Add New Virtual Disk' dialog in the Hedvig WebUI. The dialog is dark-themed and contains several sections of configuration options. At the top, there is a 'Batch' toggle switch. Below that is a 'Name' field with the value 'nfsv41'. The 'Size' is set to '10 TB'. The 'Disk Type' is 'NFS'. The 'Workload Type' is 'VIRTUALIZATION'. The 'NFS Version' is set to '3'. The 'Encryption' option is highlighted with a red circle. Other options include 'Enable Deduplication' and 'Clustered File System'. The 'Description' section has 'Compressed', 'Client-side Caching', and 'CSV' toggle switches. The 'Protection Policy' section has 'Replication' and 'Erasure Coding' radio buttons. The 'Residence' section has 'HDD' and 'Flash' radio buttons. The 'Replication Policy' is 'Data Center Aware'. The 'Pick Data Center(s)' section has three options: 'snc1', 'snc2', and 'snc3'. The 'Replication Factor' is set to '1' and the 'Block Size' is set to '512'. A 'Run' button is at the bottom left.

Figure 2: Add New Virtual Disk dialog in the Hedvig WebUI

Encrypt360 Details

- [Workflow](#)
- [Cluster Installation](#)
- [Volume Creation](#)
- [Writes](#)
- [Reads](#)
- [Performance Optimization](#)

Workflow

Hedvig performs inline encryption/decryption during write and read operations for encryption-enabled volumes.

Cluster Installation

Hedvig runs the `requestclusterkey` command one time per tenant before starting the use of encryption. Every tenant can contact its own preferred KMS vendor.

This command requests the KMS to generate one key for the cluster. A reference to this key is persisted on Hedvig.

Volume Creation

Hedvig performs encryption at a volume level. The contents of each volume get encrypted with two unique AES256 keys.

These keys are generated by Hedvig and are encrypted by the KMS vendor. Hedvig persists the encrypted keys on the Hedvig system.

All deduplicated disks share the same keys at system-deduplication disk level.

Writes

During data ingestion, the Hedvig Storage Proxy encrypts each block of data using a unique volume key. This assures that data is encrypted at the compute layer, itself, and remains secure during network transfer to the backend storage layer. This also assures that data is implicitly encrypted at rest.

Hedvig never decrypts data at the storage layer. Therefore, encryption keys are used only during read or write operations at the compute layer by the Hedvig Storage Proxy.

For deduplication-enabled volumes, Hedvig performs encryption and ensures that only unique data is encrypted, without hurting any deduplication-ratios.

Reads

During reads, the Hedvig Storage Proxy fetches encrypted data from the storage backend and decrypts it using the same unique volume key that was used during encryption.

Performance Optimization

Hedvig leverages storage acceleration libraries to offload the processing to Intel CPUs, minimizing the performance overhead. These Intel CPUs must support the following instruction sets:

- Intel® Advanced Encryption Standard-New Instruction (AES-NI),
- Intel® Streaming SIMD Extensions (Intel® SSE),
- Intel® Advanced Vector Extensions (Intel® AVX),
- Intel® Advanced Vector Extensions 2 (Intel® AVX2).

You can run the `lscpu` command to ensure that the CPU supports the above-mentioned instruction sets.

Glossary

This glossary contains definitions of terms used in this document.

Table 3: Glossary of terms

term	definition
AES	The <i>Advanced Encryption Standard</i> is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology in 2001.
AVX	<i>Advanced Vector Extensions</i> (also known as <i>Sandy Bridge New Extensions</i>) are extensions to the x86 instruction set architecture for microprocessors from Intel and AMD proposed by Intel in March 2008 and first supported by Intel with the Sandy Bridge processor shipping in Q1 2011 and later on by AMD with the Bulldozer processor shipping in Q3 2011.
AWS	<i>Amazon Web Services</i> , a subsidiary of Amazon.com, offers a suite of cloud-computing services that make up an on-demand computing platform.
Hedvig Storage Cluster	A <i>Hedvig Storage Cluster</i> is an elastic cluster, formed by using any type of commodity server(s).
Hedvig Storage Cluster Node	A <i>Hedvig Storage Cluster Node</i> is an individual commodity server running Hedvig Storage Service software.

term	definition
Hedvig Storage Proxy	A <i>Hedvig Storage Proxy</i> is a lightweight software component that deploys at the application tier as a virtual machine or Docker container, or on bare metal, to provide storage access to any physical host or virtual machine in the application tier. The storage proxy presents block, file, and object (Amazon S3) storage access to app hosts, accelerates read performance with flash caching, drives efficiency with deduplication, and secures data with encryption. This may also be referred to as an HSP, controller, CVM, target, or tgt.
KMIP	The <i>Key Management Interoperability Protocol</i> is an extensible communication protocol that defines message formats for the manipulation of cryptographic keys on a key management server.
KMS	A <i>key management system</i> [also known as a cryptographic key management system (CKMS)], is an integrated approach for generating, distributing, and managing cryptographic keys for devices and applications.
XTS	<i>XEX-based tweaked-codebook mode with ciphertext stealing</i> is a block cipher mode of operation used for full disk encryption

Commvault Systems, Inc., believes the information in this publication is accurate as of its publication date. The information is subject to change without notice. The information in this publication is provided as is. Commvault Systems, Inc., makes no representations or warranties of any kind with respect to the information in this publication and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any Commvault Systems, Inc., software described in this publication requires an applicable software license. All trademarks are the property of their respective owners. Revision date: 102021.

Software-defined AES-256, FIPS compliant encryption of data in flight and at rest.