WHITEPAPER

DELL EMC DATA DOMAIN ENCRYPTION

A Detailed Review

ABSTRACT

The proliferation of publicized data loss, coupled with new governance and compliance regulations, is driving the need for customers to encrypt their data at rest. Dell EMC® Data Domain® Encryption software enables organizations to enhance the security of their backup data that resides on their Dell EMC Data Domain deduplication storage systems.

August, 2017

Table of Contents

EXECUTIVE SUMMARY

The proliferation of publicized data loss, coupled with new governance and compliance regulations, is driving the need for customers to encrypt their data at rest. Dell EMC® Data Domain® Encryption software option enables organizations to enhance the security of their backup and archive data that resides on their Dell EMC Data Domain deduplication storage systems.

Data Domain Encryption is an optional software license, which is supported on all Data Domain systems and works across all supported protocols and in concert with Data Domain Replicator software. It offers real time inline encryption of all user data on the Data Domain system and provides a complete scheme for securing data at rest, delivering:

- Use of industry standard 128-bit or 256-bit Advanced Encryption Standard (AES) algorithms implemented by FIPS 140-2 validated RSA BSAFE® cryptographic libraries for encrypting and decrypting all stored user data.
- An additional safeguard that encrypts the content-encryption keys[1] stored on the Data Domain.
- Improved security by requiring dual-authentication of administrator and Security Officer in order to access/modify encryption options.
- Encryption of all user data at rest, which safeguards the data in the event of theft of disk shelves and/or individual disks. Additional privileged commands are provided to lock and unlock the file system, which also secures and protects user data during system transport.
- Content-encryption key lifecycle management, providing either local content-encryption key rotation or supports a centralized external content-encryption key lifecycle management through integration with RSA Data Protection Manager (DPM). RSA DPM delivers a robust, external content-encryption key lifecycle management solution for Data Domain systems and the entire enterprise.
- Support for all Data Domain replication types and encryption of data in flight can also be enabled with the Data Domain Replicator software to further secure the data replicated over the WAN, in lieu of using VPN tunneling.

AUDIENCE

This white paper is an overview of the Data Domain Encryption software option and its capabilities as reflected in Data Domain Operating System 5.3. This is intended for customers (operations managers and security officers) interested in knowing more about how the Data Domain Encryption software is implemented, managed, and deployed.

[1] *A content-encryption key is the encryption key used to encrypt user data. Other encryption keys are created within the Data Domain system used for different secure tasks.*

ENCRYPTION OVERVIEW

SECURE DATA MANAGEMENT

Data Domain Encryption software encrypts all incoming and decrypts all outgoing user data real time. Encrypting data at rest can help satisfy some aspects of internal governance rules and compliance regulations.

Once the encryption option is enabled and configured, the administrator has the flexibility to choose (either through the GUI or CLI) to encrypt preexisting user data on the system or just start encrypting new user data on ingest[1]. All encrypted user data requested from the Data Domain system is decrypted in real time as it is read off disk. User data is send back to the requester in its native format, unencrypted.

Data Domain Encryption uses either industry standard 128-bit or 256-bit Advanced Encryption Standard (AES) algorithms implemented by the FIPS 140-2 validated RSA BSAFE® cryptographic libraries for encrypting and decrypting all user data within the system. Depending on IT data protection policies, the block cipher modes for the AES algorithm can be set to use either the standard Cipher Block Chaining (CBC) or if more stringent encryption is preferred, Galois/Counter Mode (GCM) can be used[2].

The encryption/decryption process is done within the Data Domain system, and is transparent to all inbound/outbound protocols, backup, archiving and extended retention applications.

ENCRYPTION ARCHITECTURE

INLINE ENCRYPTION

DD Encryption seamlessly integrates with the high-speed, inline deduplication process used in Data Domain deduplication storage systems. Inline encryption provides an efficient and secure solution and ensures user data never resides in a vulnerable, unencrypted state on the disk subsystem. In conjunction with the Dell EMC Data Domain Stream-Informed Segment Layout (SISL™) scaling architecture, during ingest, segments that already exist on the Data Domain system do not incur additional encryption processing. Encryption is applied only to new, unique incoming segments stored on the system. This further reduces encryption resource overhead and also optimizes ingest performance.

ENCRYPTION PROCESSING

The encryption/decryption process is done in software within the Data Domain file system. No encryption hardware assist is used. The software stack has full access to all the platform's resources to do the necessary processing. All internal software mechanisms including cleaning and replication are aware of and work seamlessly with the encryption process. The interaction of encryption with these functions will be covered in a later section. For restore (read) operations, decryption is done as data is being read from disk.

ENCRYPTION OF USER DATA

Figure 1 below shows that encryption processing is the last step completed before data is written to disk. The flow of user data from ingest to disk is summarized as follows:

User data flows either directly from the client or through a backup application over one of the supported native protocols. The SISL engine segments incoming user data, and then each segment is fingerprinted and indexed on either the client (when using Data Domain Boost) or as it is ingested into the Data Domain system. On the Data Domain system, unique segments are identified and collected together into compression units; where local compression is applied.

[1] Encrypting preexisting user data can take time; duration depends on many factors, including, but not limited to: the amount of preexisting user data, platform controller and existing system workload.

[2] CBC is the most broadly recognized and widely used cipher; the latter GCM, is a more secure algorithm, but requires more system resources and therefore, can affect overall system performance.

When encryption is enabled, the encryption operation is applied to each compression unit. The encrypted compression units are then collected into compression regions, which are written in 4.5 MB sized containers. The container is the fundamental unit of stored user data on disk. A single content-encryption key is maintained per container. These containers become part of a larger data block that is sent to the RAID layer and then written to disk.

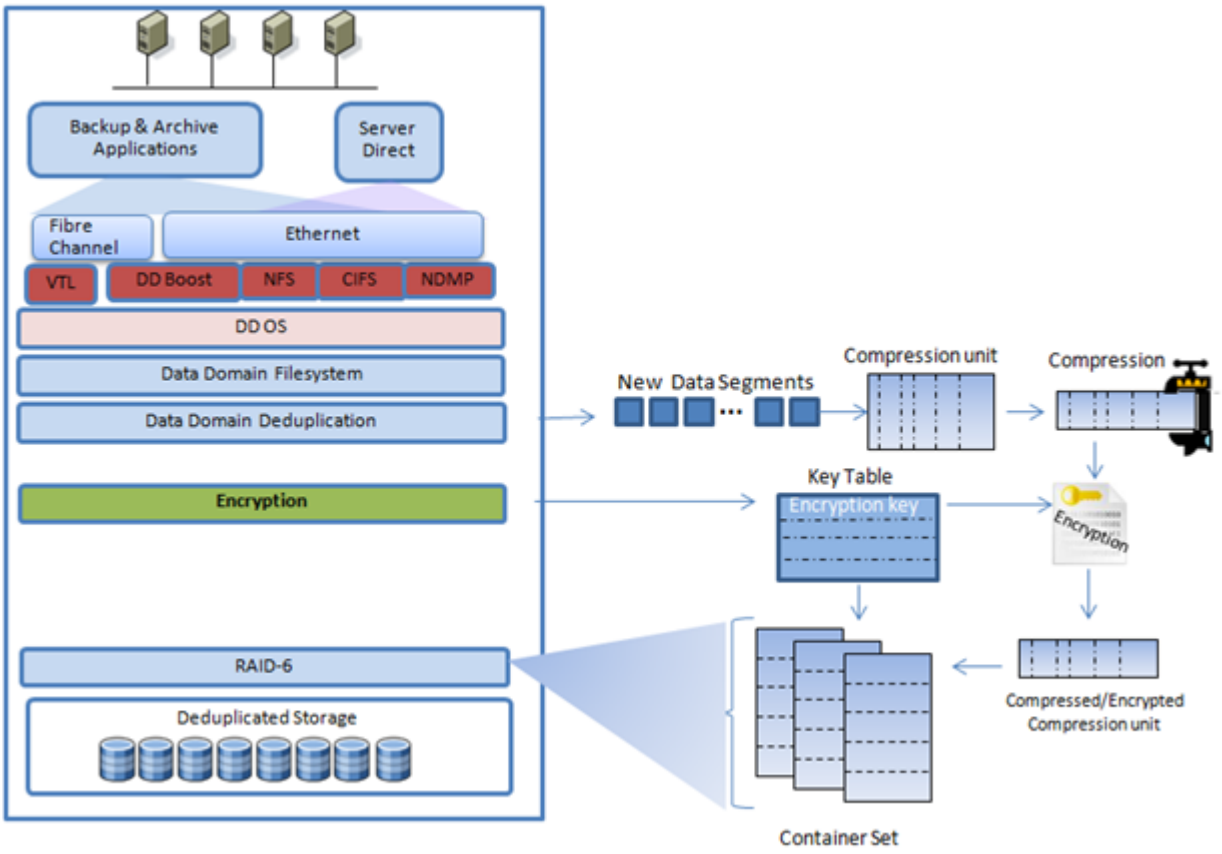

Figure 1: The internal encryption process expanded

VALUE IN ENCRYPTING USER DATA AT REST

DD Encryption was designed to protect against unauthorized access to user data as a result of the theft of disk shelves, individual disks or while failed disks are being returned for repair. For system transport, when the proper steps are performed, the entire system can be protected from unauthorized access.

With DD Encryption, all user data stored on disk are encrypted. The content-encryption key(s) used to encrypt the user data are also stored on disk encrypted. This obfuscation of the content-encryption key, used to encrypt the data, makes it impossible to read user data from an individual disk drive or an entire shelf without access to the original content-encryption key. This ensures that all user data at the individual drive level is protected against forensic hacking; drives can be safely returned to depots for repair without concern.

If security when transporting a Data Domain system is of concern, the file system can be locked before transport and unlocked at its destination. Two levels of administration access are required to lock and unlock the file system.

ENCRYPTION MANAGEMENT ENVIRONMENT

Management of the encryption option is simple and requires an installed Data Domain Encryption software license. Once installed, configuring the encryption option requires:

- The setup of a Security Officer account which is an authorized user in the Security User group on the Data Domain system. Both a system administrator AND the Security Officer are required to change any of the encryption options.
- Establishing a passphrase for the system. The encryption passphrase is a human-readable key, which can be up to 256 bytes in length. It is used in the generation of the encryption key that is used to encrypt the content-encryption keys. The Administrator & Security Officer role together can change the passphrase, which does not affect user data or the underlying content-encryption key used to encrypt the user data.
- Selecting the encryption algorithm (AES 128/256-bit) and the cipher mode (CBC or GCM) to be used.
- Selecting which encryption key management method to be used (none, internal or external)
- If RSA DPM is used for external key management, set up is required on both the DPM Server and on the Data Domain system. RSA DPM requires a mutually authenticated secure session between the DPM server and the Data Domain system(s); which requires a X.509 certificate. The CLI on the Data Domain system is used to import the appropriate externally signed certificate.

FLEXIBLE ENCRYPTION KEY MANAGEMENT

The content-encryption key(s) are managed in one of the three ways. Encryption of all user data on the system will conform to one of these methods for each Data Domain system. Only one of these three methods can be used on an individual Data Domain system. All three can coexist in any combination in the environment.

NONE (SINGLE, STATIC KEY)

A single static content-encryption key is applied to all user data residing on the system. The content-encryption key is internally generated; it cannot be viewed, accessed or stored externally and key rotation is not possible. This is the default method used when DD Encryption is enabled for any Data Domain system up to DD OS 5.2.

INTERNAL KEY MANAGEMENT

Introduced with DD OS 5.3, the user can now choose internal key management, which allows the encryption key rotation to be set on a periodic schedule.

Internal key management allows the administrator to leave the content-encryption key static, or rotate the content-encryption key on a regular basis. The helps satisfy compliance mandates where regular content-encryption key rotation is required. The administrator selects a particular period (in months) when rotation will occur. The system will support up to 254 different content-encryption keys. There is only a single active read/write key used for all data ingested at any one point in time.

When a content-encryption key is rotated, the currently active read/write content-encryption key is replaced with another content-encryption key; once the file system is restarted the system will begin using the new content-encryption key as new data is ingested. The state of the previously active content-encryption key state will change to read only; the user data associated with it also becomes read only. The existing data and content-encryption key will remain on the system until the associated user data expires and is deleted.

The related content-encryption key cannot be deleted until all the associated data have been deleted. Other key states and actions needed on the Data Domain system are addressed later in the document. With internal key management, for disaster recovery purposes, the content-encryption keys can also be externally exported, via secure file transfer to a secure external location for safe external key vaulting. Internal key management is the default method used when DD Encryption is enabled for any Data Domain system on DD OS 5.3 or above.

EXTERNAL KEY MANAGEMENT

With external key management, the key states are essentially the same as with internal key management. Only the content-encryption keys are now externally generated and managed through the RSA Data Protection Manager© (DPM) external key management solution.

RSA Data Protection Manager enables centralized key management and transparent, automated policy enforcement for encrypting data at rest across the information lifecycle: in primary storage, host databases and protection storage including Data Domain systems. Unique in the industry, Data Protection Manager also provides application protection, via application encryption and tokenization; a true, robust enterprise-class external key management solution.

When an RSA DPM external key management solution is used, the Data Domain DPM embedded client is enabled. The Data Domain system will begin polling the RSA DPM server once a day for any key state changes. The key states and actions needed on the Data Domain system are explained below. External key management is supported with DD OS 5.2 and above.

You can move from the single, static encryption key to internal encryption key management or external key management at any time.

KEY STATES

The Data Domain system will maintain different key states, when either the internal or external key management is selected. These key states are:

- **Activated-RW:** There is only one key in this state on any Data Domain system and is used for reading and writing data since the last key rotation. This key is also used by the GC process to re-encrypt containers. This is same as the RSA DPM's activated key state.
- **Pending-Activated:** There is only one key in this state on any Data Domain system. This identifies the key that will become Activated-RW after the next file system restart.
- **Activated-RO:** Any Data Domain system can have multiple Activated keys. The most recent one will be in the Activated-RW state; the rest will be used for reading the user data only (equivalent to DPM deactivated state). Keys in this state are used for reading existing data on the system.
- **Deactivated-RO:** This state is a result of the DPM administrator manually deactivating a content-encryption key. When the Data Domain system picks up this state, it will change the state of the Activated-RO key to deactivate. This has the same net affect as the Activated-RO state, and is used for reading existing user data. *This state is used with RSA DPM only.*
- **Compromised:** A key in this state indicates the data associated with it will be re-encrypted with a new key at the next GC cycle. This key will be used for reading data on the DD system until the next GC process is run. During GC, all containers encrypted with compromised keys will be re-encrypted using the Activated-RW key.
- **Marked-For-Destroyed:** An administrator can mark a key in this state. When GC is run, all containers encrypted with Marked-For-Destroyed keys will be re-encrypted using the Activated-RW key.
- **Destroyed:** A key in the Marked-For-Destroyed state goes into this state when there is no data associated with it.
- **Destroyed-compromised:** A key in the Compromised state goes into this state when there is no data associated with it.

CHANGING KEY MANAGEMENT TYPE

Only one key management method can be in active use on a Data Domain system at any one time. The key management type can be changed at any time. However, when the key management type is changed, all of the data previously encrypted data will not be affected. As the GC process runs, it will take care of re-encrypting any of the preexisting data with the currently Active-RW content-encryption key; for instance, when data movement from one container to another is necessary.

Also, when changing from static or internal content-encryption key to RSA DPM, none of the pre-existing static or local content-encryption keys are transferrable into RSA DPM. These will be managed separately on the Data Domain system. RSA DPM has no visibility to these keys and they cannot be protected or managed by DPM.

KEY MANAGEMENT STATES FOR KEY ROTATION AND RELATED ACTIONS

Table 1 below explains the various actions and effects on the Data Domain system as a result of the major content-encryption key state changes. The four rows contain actions and effect on the Data Domain system based on the four main key states.

Data Domain Effects	Activated (Key Rotation)	Deactivated	Compromised	Destroyed
System Alerts	Administrator notified of key changes pending	None	Administrator notified of key changes pending	None
Admin Actions	Admininstrator restarts File System	None	If Active key - admin restarts File System If deactivated, key no FS restart necessary	None
System Behavior	New (Active) key will be applied to all new data Ingest	Old active key becomes deactivated Read only for existing data using this key	Affected data using compromised key will get re-encrypted at next GC process	Keys can be deleted, GC process ensures no data is associated with key before deletion
Affect on GC process	Apply '*apply-all-changes*' option to e-encrypted ALL pre-existing data *Most impact to GC Process*	Container optimization will cause some data to re-key Small impact to GC process	*Depends on the amount of data needing to be re-key* *Moderate-high impact to GC Process*	None

ENCRYPTION EFFECTS ON THE GLOBAL CLEANING (GC) PROCESS

The GC process focuses on reclaiming space used by unreferenced data segments. Data segments become unreferenced in the Data Domain system when data exceeds the retention period as defined by the backup or archive application policies. When the cleaning process runs, the cleaning activity removes unreferenced segments in specified containers that are marked for deletion. In addition when appropriate, GC also repacks certain segments into other containers (carry forward segments) for optimizing locality.

Encryption adds an overhead to the cleaning process cycle in the following areas:

- All segments that have to be read out of containers or written into containers will go through the encrypt/decrypt process at the compression unit level. Cleaning completion times will extend as a result of this extra step. The extent of the impact will depend on the number of segments that have to be touched in the cleaning process and also the processing resources available to perform the decryption/encryption on that Data Domain system.
- Containers that are targeted for cleaning are decrypted as the compression units are read into memory. Segments that are carried forward to other containers, for container optimization, will get added to compression units in other containers, where the different content-encryption key is being used. When this happens, the newly formed compression unit will get encrypted with the content-encryption key assigned to that container.

The amount of additional time the cleaning process will take to complete is dependent on the amount of deleting, repacking/carrying forward of segments during that cleaning cycle; as these segments being read into memory are decrypted as they are read, and encrypted again as the new compression units are formed and written to their respective containers.

In addition to the above, the cleaning process will also make any encryption operations needed to handle a compromised key condition. In the case of a compromised key, the decryption/encryption processing will have to be applied to all the containers affected by this compromised key. Because this particular key state can have the most effect on the systems' GC performance, careful consideration is advised before compromising a content-encryption key.

ENCRYPTION EFFECTS ON OVERALL SYSTEM PERFORMANCE

When DD Encryption is enabled, encryption/decryption operations will require resources along with all the other internal processes running within the operating system. Here are some general performance related considerations when encryption of data at rest is enabled:

- Read performance will be impacted, as reads require decryption of all user data read from disk.
- Write performance will be minimally impacted, as encryption is only performed on new, unique user data written to the Data Domain system.
- If encryption of data at rest is enabled, all read requests will impact performance as reads require decryption user data read from disk.
- There will be a one-time performance impact, for a period of time, if the user chooses to encrypt user data that existed prior to enabling encryption in the Data Domain system. Duration depends on the amount of data to encrypt.
- If a content-encryption key is marked as compromised (available with RSA DPM or internal key management), user data associated the compromised content-encryption key will have to be decrypted and then re-encrypted with a new content-encryption key. The next cleaning cycle will initiate the decrypt/re-encrypt operations on those segments/containers associated with that compromised key.
- There is a tradeoff between using a stronger cipher and performance (CBC vs. GCM). Specifically, GCM, though the most secure cipher, requires slightly more system resources, and could result in some performance impact.
- There is no noticeable performance difference between AES128 and AES256.

DATA DOMAIN ENCRYPTION AND DATA DOMAIN REPLICATOR SOFTWARE

DD Encryption is supported with all Data Domain replication types: directory replication, MTree replication, managed file replication and collection replication. As long as both the source and destination in a replication pair are running DD OS 4.9 or higher, there are no other special rules imposed by encryption on compatibility of the replication systems. Encryption can be enabled on source, destination or both. The content-encryption key remains in the system's memory for the remainder of the replication session, once the replication session ends, it is deleted. The following sections explain the encryption operation for the different replication types.

DIRECTORY REPLICATION, MTREE REPLICATION, AND MANAGED FILE REPLICATION

If both the source and destination have encryption enabled and different content-encryption keys:

- All data sent directly to the source Data Domain system (e.g. through backup/archive) is encrypted with the source system's content-encryption key. This encryption work is done by the source Data Domain system.
- When replicating unique segments, the source decrypts the local data, re-encrypts it using the destination system's content-encryption key, and then replicates the encrypted data to the destination Data Domain system.
- Any data sent to the destination Data Domain system outside of replication (e.g. direct backup/archive at the remote site) is encrypted using the destination system's content-encryption key by the destination Data Domain system.

If the source has encryption disabled and destination has encryption enabled:

- All data sent directly to the source Data Domain system (e.g. through backup/archive) is not encrypted.
- When replicating, the source encrypts the user data using the destination Data Domain content-encryption key and then replicates the encrypted data to the destination Data Domain system.

- Any data sent to the destination Data Domain system outside of replication (e.g. direct backup/archive at the remote site) is encrypted using the destination system's content-encryption key by the destination Data Domain system.

If source has encryption enabled and destination has encryption disabled:

- All data sent directly to the source Data Domain system (e.g., through backup/archive) is encrypted with the source system's key by the source Data Domain system.
- When replicating, the source decrypts the user data and then replicates the unencrypted data to the destination Data Domain system.
- Any data sent to the destination Data Domain system outside of replication (e.g., direct backup/archive at the remote site) is not encrypted.

COLLECTION REPLICATION

Note: Source and destination Data Domain systems must have the same version of DD OS running and encryption must be either enabled or disabled on both.

Source and destination both have encryption enabled:

- All data sent directly to the source Data Domain system (e.g., through backup/archive) is encrypted with source system's content-encryption key by the source Data Domain system.
- When replicating, the source sends the encrypted containers (encrypted using the source system’s key) to the destination Data Domain system.
- No data can be written to the destination Data Domain system outside of replication, as the destination is a read-only system.

Source and destination both have encryption disabled:

- When the encryption option is disabled, neither source nor destination Data Domain system will encrypt data at rest.

CONCLUSION

The Data Domain Encryption software option is a robust, secure data management solution that can encrypt all user data stored on a Data Domain deduplication storage system. It protects user data from theft or loss of system, disk shelves, disks, or for disks returned to factory. The Data Domain Encryption software can help satisfy internal governance rules and helps with meeting compliance regulations.

DD Encryption helps meet compliance regulations by using industry standard AES-128 or AES-256 encryption algorithms and the Dell EMC RSA BSAFE FIPS 140-2 validated cryptographic libraries as well as supporting both standard CBC and the stronger cipher mode GCM for additional security.

In concert with the SISL processes, encryption is done inline before it is written to disk, so it never lands on disk in an unencrypted state. By default, system resources are not expended on encrypting existing data.

Encryption key management and integrity is maintained using the internal key manager, or using RSA Data Protection Manager for centralized, external encryption key lifecycle management. Internal or external encryption key management is supported on any data domain system in the environment, and any single management type can coexist on Data Domain systems in the same environment.

The software requires two levels of administration- Security Officer /Administrator. Passphrase protected encryption keys offer a strong administration framework.

Data Domain Encryption software is transparent to all ingest protocols and backup/archiving applications and works with all Data Domain replication types and also works with Dell EMC Data Domain Retention Lock software option.

Learn more about Dell EMC Data Domain solutions

Contact a Dell EMC Expert