FORTANIX TECHNICAL DOCUMENTATION

User Guide

FORTANIX DATA SECURITY MANAGER WITH AWS XKS - CONCEPTS

VERSION 1.1

Fortanix, Inc. | 3910 Freedom Circle | Suite 104 | Santa Clara, CA 95052| United States of America | ☎ +1 (650) 943-2484 |
✉ info@fortanix.com | 🌐 www.fortanix.com

1.0 INTRODUCTION

Welcome to the **Fortanix Data Security Manager** (DSM) with **Amazon Web Services** (AWS) **External Key Store** (XKS) documentation.

The Fortanix DSM integration with AWS XKS enables organizations to protect the data in AWS with keys stored in Fortanix DSM.

This document describes the following topics:

- DSM with AWS XKS use cases.
- Advantages of DSM with AWS XKS integration.
- DSM with AWS XKS workflow.
- DSM with AWS XKS integration.

1.1 TERMINOLOGIES

- **Fortanix Data Security Manager (DSM) –**

  Fortanix DSM is the cloud solution secured with Intel® SGX. With Fortanix DSM, you can securely generate, store, and use cryptographic keys and certificates, as well as secrets, such as passwords, API keys, tokens, or any blob of data.

- **AWS KMS –**

  AWS Key Management Service (AWS KMS) lets AWS users create and control cryptographic keys and their access policies for use by a large number of AWS Services.

- **AWS XKS –**

  The External Key Store (XKS) feature allows AWS KMS users to establish an external Root of Trust for encryption keys managed by KMS. With XKS, AWS users can externally generate, host, and manage Root Keys to protect data encryption keys used by AWS Services and applications.

1.2 DSM WITH AWS XKS – USE CASES

The Fortanix DSM with AWS XKS integration provides Fortanix customers the ability to migrate privacy-sensitive workloads for highly regulated industries, such as financial services and healthcare, to the public cloud and comply with the highest data privacy regulations.

1.3 DSM WITH AWS XKS - ADVANTAGES

The following are the advantages of Fortanix DSM with AWS XKS integration:

- The users have complete custody of their keys and full control over the data encryption policies within AWS. This control helps them in specifying the location of the keys, and from where they may be accessed.
- Fortanix DSM offers comprehensive audit logs, so the users may demonstrate that their security controls adhere to regulations like the GDPR.
- AWS provides strong key protection, and Fortanix does not compete with these functions. Instead, Fortanix provides Segregation of Duties with external, granular access control.

1.4 AWS XKS IMPLEMENTATION CONSIDERATIONS AND REFERENCES

The following Amazon Web Service (AWS) references will assist in ensuring a successful implementation of the AWS External Key Store (XKS). This procedure must be reviewed carefully and used for planning before the actual integration steps required for the Fortanix DSM integration.

- **AWS XKS Announcement or Overview**

  For more information, refer to

  https://aws.amazon.com/blogs/aws/announcing-aws-kms-external-key-store-xks/
- **AWS Developer's Guide (KMS XKS)**

  For more information, refer to

  https://docs.aws.amazon.com/kms/latest/developerguide/keystore-external.html
- **AWS KMS Service Features**

  For more information, refer to

  https://aws.amazon.com/kms/features/#AWS_Service_Integration
- **AWS XKS FAQ**

  For more information, refer to https://aws.amazon.com/kms/faqs/#External_key_store

In addition to these general AWS resources, the following specific AWS XKS documents provide essential details for deciding how best to implement AWS XKS when planning an enterprise deployment:

- **AWS XKS Requirements and Planning**
  For more information, refer to
  https://docs.aws.amazon.com/kms/latest/developerguide/create-xks-keystore.html#xks-requirements
  Note the deployment options and connectivity considerations for the required XKS proxy, which is included in the DSM SaaS offering from Fortanix.
  - For on-premises deployments where a publicly accessible proxy Uniform Resource Identifier (URI) endpoint will not be used, the customer must create, host, and maintain their own XKS proxy to ensure proper communication with AWS and the external key manager instance (Fortanix DSM).
  - For non-SaaS deployments, refer to the networking requirements for configuring the Virtual Private Cloud (VPC) endpoint service connectivity: https://docs.aws.amazon.com/kms/latest/developerguide/vpc-connectivity.html#xks-vpce-service-requirements
- **AWS KMS IAM Permissions for Creating an External Key Store**
  For information on controlling access to your external key store, refer to.
  https://docs.aws.amazon.com/kms/latest/developerguide/authorize-xks-key-store.html
- **Permissions for Creating XKS-backed KMS Keys**
  For creating KMS keys in an external key store, refer to
  https://docs.aws.amazon.com/kms/latest/developerguide/create-xks-keys.html#xks-key-requirements
- **Connectivity Options**
  - For AWS XKS SaaS deployments leveraging Fortanix, an XKS proxy is easily configured to communicate to the public URI path configured within the Fortanix DSM options for XKS support. Refer to the *Fortanix Data Security Manager with Amazon Web Service External Key Store Integration Guide* for the detailed steps.
  - For on-premises and non-public endpoint service connectivity, this is an advanced process that will be owned by the customer.
  - **Configuring VPC Endpoint Service Connectivity**

For more information, refer to https://docs.aws.amazon.com/kms/latest/developerguide/vpc-connectivity.html#xks-vpce-service-requirements

- **Choosing a Proxy Connectivity Option**

  For more information, refer to https://docs.aws.amazon.com/kms/latest/developerguide/plan-xks-keystore.html

- **GitHub for AWS XKS Proxy API Specification**

  For more information, refer *to* https://github.com/aws/aws-kms-xksproxy-api-spec. This also includes a helpful architectural diagram for networking options and flows: https://github.com/aws/aws-kms-xksproxy-api-spec/blob/main/XKS_arch_v8.png

- **XKS Latency Considerations and Monitoring**

  AWS KMS recommends that your external key store proxy be configured to handle up to 1800 requests per second and respond within the 250 millisecond timeout for each request. AWS recommends that you locate the external key manager close to an AWS region so that the network round-trip time (RTT) is 35 milliseconds or less. You can use an external key store proxy for more than one external key store, but each external key store must have a unique URI endpoint and path within the external key store proxy for its requests.

  - **Latency Troubleshooting**

    For more information, refer to https://docs.aws.amazon.com/kms/latest/developerguide/xks-troubleshooting.html#fix-xks-latency

  - **Monitoring an External Key Store**

    For more information, refer to https://docs.aws.amazon.com/kms/latest/developerguide/xks-monitoring.html

  - Learn about the Amazon CloudWatch metrics and dimensions that AWS KMS records for external key stores. AWS strongly recommends that you create alarms to monitor your external key store so you can detect the early signs of performance and operational problems: https://docs.aws.amazon.com/kms/latest/developerguide/monitoring-cloudwatch.html#kms-metrics

2.0 DSM WITH AWS XKS WORKFLOW

XKS allows AWS KMS to use external, customer-managed root keys, giving the customer more control over key management and data security initiatives. Fortanix DSM is solely responsible for creating, safeguarding, and using the customer's root keys.

The following figure depicts how AWS XKS integration with Fortanix DSM works:

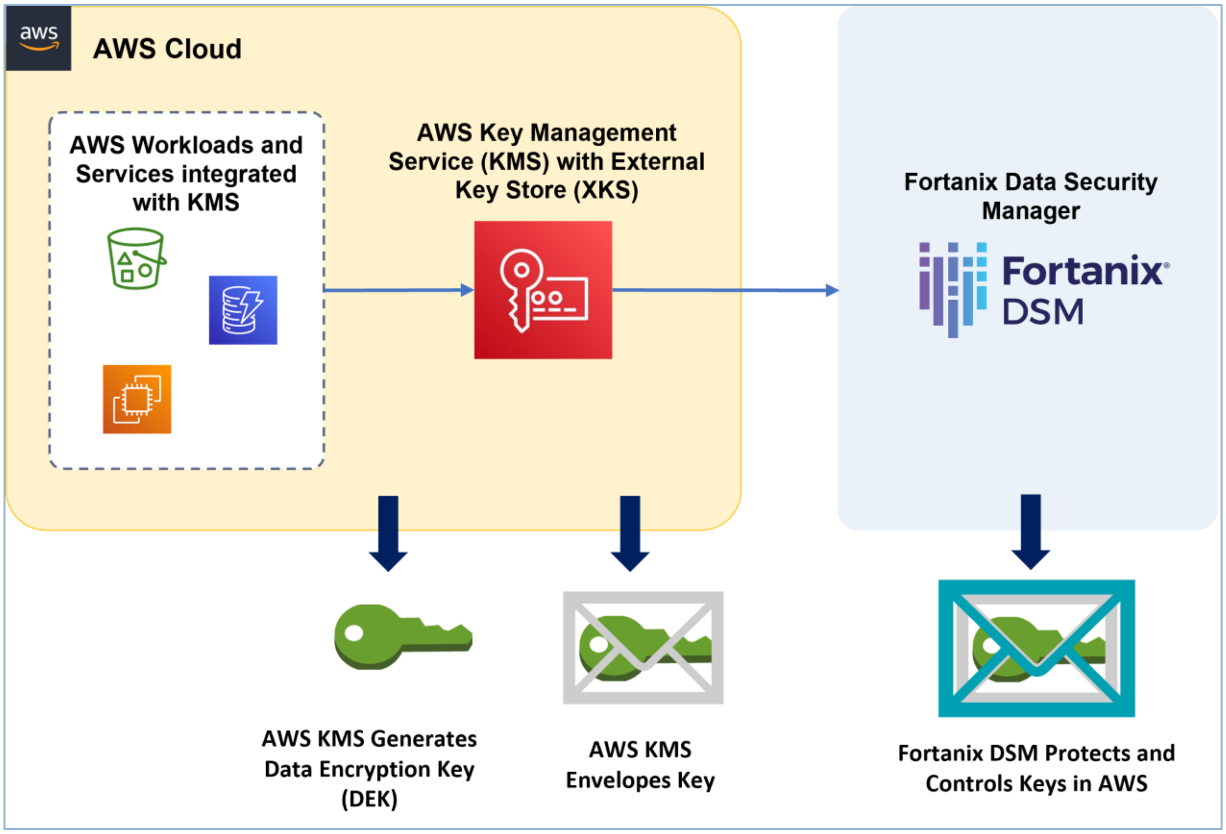
aws
AWS Cloud
AWS Workloads and Services integrated with KMS
AWS Key Management Service (KMS) with External Key Store (XKS)
Fortanix Data Security Manager
Fortanix DSM
AWS KMS Generates Data Encryption Key (DEK)
AWS KMS Envelopes Key
Fortanix DSM Protects and Controls Keys in AWS

FIGURE 1: FORTANIX DSM WITH AWS XKS WORKFLOW

The workflow between the various services is as follows:

1. A supported AWS service calls KMS and asks for a new Data Encryption Key (DEK) in an XKS-backed Key Store. For instance, this could be S3 to encrypt an uploaded file for storing in a bucket.
2. KMS generates the DEK and envelopes (encrypts) it is using a key store-specific Key Encryption Key (KEK)
3. KMS calls Fortanix DSM which, upon satisfying access controls and policies, envelopes the already enveloped key using a DSM-protected Root KEK.
4. DSM sends the double encrypted DEK back to KMS.
5. KMS stores the double enveloped DEK in its Key Store.
6. KMS then immediately retrieves the DEK, unseals the double envelope by sending it back to DSM and opening the inner envelope itself, and hands the DEK to the calling service for use.

Every time the calling service needs the DEK (for instance, S3 to satisfy a download request of the encrypted file), the last step (*Step 6*) is repeated.

3.0 DSM WITH AWS XKS - INTEGRATION

With FIPS 140-2 Level 3 certified™ HSM protection, Fortanix DSM is available as an on-premises solution as well as a SaaS offering.

- In an AWS XKS with DSM SaaS integration, the AWS service reaches out to the Fortanix DSM service to access the user’s Root Key.
- For on-premises Fortanix DSM clusters, the users need to allow network access from AWS.

For the Fortanix DSM with AWS XKS integration steps, click <here>.

4.0 DOCUMENT INFORMATION

4.1 DOCUMENT LOCATION

The latest published version of this document is located at the URL:

https://support.fortanix.com/hc/en-us/articles/11201117419412-Fortanix-DSM-with-AWS-External-Key-Store-XKS-Concepts

4.2 DOCUMENT UPDATES

This document will typically be updated on a periodic review and update cycle.

For any urgent document updates, please send an email to: support@fortanix.com

4.3 REVISION HISTORY

DOCUMENT VERSION	DATE PUBLISHED	CONTRIBUTORS	PREPARED BY	APPROVED BY	SUMMARY OF CHANGES
1.0	29-NOV-2022	SANDER TEMME RENE PAAP JOHN WYSS	NITHYA RAMAKRISHNAN		INITIAL DRAFT
1.1	14 JUNE 2023	JEFF JONES	SHIVANI GARG		ADDED SECTION 1.4

NOTICE: This document was produced by Fortanix, Inc. (Fortanix) and contains information which is proprietary and confidential to Fortanix. The document contains information that may be protected by patents, copyrights, and/or other IP laws. If you are not the intended recipient of this material, please destroy this document and inform info@fortanix.com immediately.