National Security Agency
Cybersecurity & Infrastructure Security Agency
| Cybersecurity Information

Use Secure Cloud Key Management Practices

Executive summary

Cryptographic operations enable secure communication, access control, authentication, and data encryption at rest. The security of these operations all rely on proper key management. Cloud service providers (CSPs) use key management systems (KMSs) to offer encryption and key management as a service, including functionality such as:

- Management operations on symmetric and asymmetric keys, including:
  - creation
  - storage
  - rotation
  - deletion
- Secrets management for:
  - application programming interface (API) keys
  - data encryption keys
  - other service secrets

Some CSPs offer subsets of this key management functionality in multiple service offerings. For the purposes of this cybersecurity information sheet (CSI), the term “cloud KMS” refers to any cloud services that perform any of this functionality.

A cloud KMS integrates with other cloud services to give customers some control over the keys used for cryptographic operations within the cloud tenant. Customers can opt to have CSPs manage some or all features of the KMS. [1] Best practices for a cloud KMS will depend on the boundaries of control over key management desired for each specific use case.

Granting a CSP control over key management inevitably carries some risk. The acceptability of such risks depends on several factors including:

- the sensitivity level of the data to be protected
- resources available to manage keys on premises
- level of trust established with the CSP

This CSI outlines key management options based on these factors and recommends best practices to consider when using them. With any use of a cloud KMS, it is critical to understand and document shared security responsibilities. Refer to the NSA CSI: Uphold the Cloud Shared Responsibility Model for additional information on the shared responsibility model. [2]

Common key management options for cloud customers

CSPs typically offer several options for key management, with the availability of these options varying between service offerings. Many service offerings provide users the option to perform cryptographic operations using keys the CSP creates and manages. Examples of this would be a cloud storage service that encrypts data automatically or a compute service that encrypts the storage disks with CSP managed keys. The customer has no responsibility for creating or managing the key material and typically has no control over the key length or encryption algorithm used.

On the opposite end of the spectrum, customers have the option to implement their own KMS and cryptographic processes outside of the cloud and encrypt all data prior to uploading it to a cloud storage service. This ensures that even if a malicious cyber actor (MCA) succeeds in accessing a customer's stored data, they would not have access to the keys needed to decrypt the data. However, this method is incompatible with nearly every cloud service. While users can upload encrypted data to storage services, this limits the functionality normally provided by the CSP, such as document search and other analytic capabilities. Other (non-storage) cloud services rely on access to decrypted data to perform their intended functions.

In the middle of the spectrum there are cloud KMS offerings, which let customers take advantage of a CSP's KMS while still being able to choose key types, key sizes, rotation schedules, backup policies, and access controls. A cloud KMS may offer hardware security module (HSM) and/or non-HSM-backed key options. HSM-backed keys are typically generated, stored, and used within the HSM. Depending on the service offering, the HSM may be used to protect only wrapping keys (i.e., keys used to encrypt other keys), or it may protect all customer keys. These services are generally compatible with a range of cloud services.

In some cases, customers can import keys they have created to manage and use in the CSP's KMS, or connect a third party KMS to manage the customer's keys and perform cryptographic operations. Each option has benefits and drawbacks. The best option will depend heavily on the use case.

Trust and control

Using a public cloud extends the trust boundary beyond the organization. Additional risks are introduced, such as insider threats and a lack of control over some aspects of security operations. [2] The level of control a cloud consumer has over key management varies with the type of cloud service model selected[1,2]. There is a limit to the degree of security assurance that a cloud consumer can expect when the logical and physical organization of the cryptographic resources are entirely under the control of the CSP. [3] When determining if the CSP meets the needs of the organization, it is important to review technical information from the CSP to determine if their security practices satisfy the organization's needs. Some important information to request from the CSP includes:

- Available key configurations [e.g., symmetric or asymmetric, length, purpose (e.g., encryption, signing), algorithm]
- Internal and API-based actions involving key material, including technical details which affect key visibility (e.g., generating, changing or updating, storing, retiring, retrieving, retaining, and destroying key material)
- Recommended customer key management procedures [4]

Additional information to consider when determining the level of control to grant a cloud KMS provider includes:

- How is unauthorized access to key material prevented?
- How does the CSP ensure proper lifecycle management of the keys?
- How does the CSP ensure physical and electronic protection of the KMS?
- How are vulnerabilities detected and removed from software during and after development?
- Are deleted keys recoverable? If so, for how long?
- What processes are in place to detect malicious administrators or other internal threats?

[1] Cloud service models include Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).
[2] For more details on the cloud service models, see NIST SP 800-145 "The NIST Definition of Cloud Computing"

- Are cryptographic processes that handle customer keys securely separated from other processes?
- Is there a roadmap to transition to quantum-resistant cryptography?

Answers to these questions will provide additional context when researching the CSP and may assist in building a trusted relationship.

Shared security responsibilities

No matter which service model is chosen for other cloud services, a KMS will be needed for secure communication and authentication with these services. The KMS itself may be provided by the CSP, or it may be an external key store.

When using a CSP's KMS, the customer must first identify the required security assurances for a particular use case. A variety of decisions will be based on these requirements, starting with the CSP's KMS service model. [3] The boundaries of responsibility between the customer and CSP for KMS security will depend on the cloud service model chosen. [1] However, regardless of the model chosen there will still be keys the customer is responsible for protecting.

With all of these service models and options, the customer will be responsible for defining user and administrator roles and policies for key management. Established best practices for separation of duties and least privilege apply for all cloud KMS uses. [5]

In some cases, a CSP may use the KMS services of a different CSP. In situations where an intermediary provider and primary provider partner to offer cloud services, key management responsibilities may need to be divided between the two. [3] The customer may be unaware of how the KMS functions are divided. Seek clarification from the CSP to determine if the KMS they offer is owned and managed solely by that CSP.

Software as a service KMS model

When using a Software as a Service (SaaS) model for a KMS, customers interact with the API of a CSP's KMS to make key management, signing and verifying, and encryption/decryption requests. These operations are performed in software owned and managed by the CSP running on hardware the CSP manages. A secure session must be established for this interaction.

CSPs offering a SaaS model KMS, may offer a choice between provider-managed keys and customer-managed keys. The customer has greater control over the lifecycle of a customer-managed key. For example, the customer-managed key can be created on the CSP’s KMS or generated on premises and then imported to the cloud KMS, a model commonly referred to as bring your own key (BYOK). Customer-managed keys can potentially provide greater flexibility over data access controls. However, this would be at the expense of greater resources required to properly perform key management duties. The CSP may also restrict the types of keys that can be imported.

Infrastructure as a service KMS model

An example of the Infrastructure as a Service (IaaS) model for a KMS would be a dedicated FIPS 140 level 3 validated HSM running on a virtual enclave, or cloud HSM, which the customer owns and manages. This model minimizes CSP access. The CSP provisions the HSM, handling physical and network security, rack space, power, and network integration, and exposes it to the customer’s virtual enclave. The customer must provide the expertise to configure and maintain the HSM. The use of IaaS for a KMS is rare compared to its SaaS counterpart. This is generally a highly specialized service used by large-scale organizations to satisfy regulatory requirements.

Hardware security modules

Regardless of the chosen KMS implementation, using a validated HSM-based key management system is a best practice for managing keys used to encrypt highly sensitive data. HSM-based keys are generated, used, and stored on the HSM, so it has a smaller attack surface than other key management options. HSM’s are often designed with physical anti-tamper protections to make it extremely difficult to extract keys that are not configured to be exported. However, the CSP’s plan for interaction with the HSM will affect the security of the KMS. HSMs may be shared, logically partitioned, or dedicated.

With shared HSM offerings, no single customer has control over how resources are allocated or consumed. The hardware itself is a shared resource and data separation is dependent on CSP software. This presents a risk of unauthorized access to sensitive data from other tenants. If an MCA found a weakness in the CSP’s software, they could register as a customer and exploit the weakness to gain access to another customer’s data.

Partitioned HSMs provide logical but **not** physical isolation from other customers' data. Each partition should have separate administrative access and its own data, access controls, and security policies. As multiple customers interact with the device there is a risk that one customer may be able to exploit a side-channel technique to read memory on the HSM that is outside of their partition, allowing them to exfiltrate other customers' keys. To mitigate the risk of a partition isolation breach, HSMs should be configured to disable features that allow users to run their own code.

Dedicated HSMs provide **physical** separation from other customers, which eliminates the risks described above. With dedicated HSMs, the customer provisions a hardware device and has full administrative responsibility for the HSM. This higher level of control may itself present a risk if the customer does not have sufficient resources to manage the device properly.

Integrating a KMS

Another option for using a KMS with cloud services is a KMS that is external to the CSP hosting services. This can be either an enterprise owned and managed KMS or a secondary CSP KMS integrated with a primary CSP. CSPs will not have the ability to store and manage encryption keys for cryptographic operations for customers if the customer uses their own KMS or a separate, distinct KMS.

CSPs have differing mechanisms and standards for utilizing or managing an integrated KMS. For example, in addition to their own CSP-specific APIs, or APIs supplied by the vendors of the hosted HSM, some CSPs offer compatibility with the Key Management Interoperability Protocol (KMIP) or with Public-Key Cryptography Standards (PKCS) #11. Check with the provider to ensure that the existing or planned service is compatible with the protocols and procedures offered by the CSP.

Protecting sensitive information

Organizations operating in the cloud should take precautions to protect sensitive information. Organizations should periodically audit key usage within the cloud environment to verify key usage aligns with its intended purpose, including where, how, and by whom keys are used. Service level agreements should contain language describing the CSP's key management policies, including any mechanisms in place to isolate the CSP from customer-controlled keys. [6]

Metadata in cloud environments is frequently exposed in audit trails and in some cases to the CSP in order to operate and maintain services. To prevent exposure of sensitive information it is important not to use sensitive information in these data types. These data types may vary by vendor, but a common example is key labels.

Additionally, customers should verify that CSPs offer the proper mechanisms to encrypt customer keys while at rest and in transit, and take precautions to ensure customer keys are always encrypted at rest and secure channels are used when the keys are transmitted between internal cloud services. Managing sensitive information used with applications (e.g., API keys, database connection strings, data encryption keys, passwords) is especially challenging because it might be insecurely stored in application code, configuration files, or integration and deployment pipelines. Minimize the exposure of customer keys by using automated scanning to detect exposed keys in these places. MCAs may also attempt to acquire enough permissions to pull these keys and credentials directly from the cloud KMS or any other cloud-native secrets manager being used in the environment. Permissions to read these secrets should be limited and queries should be monitored.

ATT&CK® Tactic	Technique
Credential Access	Unsecured Credentials: Credentials in Files [T1552.001]
Credential Access	Credentials from Password Stores: Cloud Secrets Management Stores [T1555.006]

D3FEND™ Tactic	Countermeasure
Platform Hardening	Disk Encryption [D3-DENCR]

Special care should be taken to preserve cryptographic keys that are required to access encrypted forensic evidence. Any issues that may arise in preserving these cryptographic keys may cause organizations to lose the ability to decrypt forensic data stored in the cloud. [7]

Key destruction considerations

Customers should be knowledgeable about a CSP's key destruction process. When a key is deleted, it is often moved to a "deletion pending" state for a specified length of time before it is actually purged from the CSP. While awaiting destruction, the key may be recoverable; however, the length of time and whether this action can be reversed varies by vendor. When keys are deleted, any data encrypted by the destroyed key can no longer be decrypted. Organizations should ensure the proper controls are in place to verify a key is no longer needed before deleting. Organizations should also be aware of what the CSP's key destruction commitments are before selecting a KMS offering. Cloud storage offerings do not always provide actual erasure of data, relying instead on cryptographic erasure where erasure of the encryption keys provides assurance that deleted data is not recoverable. However, destruction of deleted key material is frequently only guaranteed after several months. Organizations should take this into account when deciding what level of sensitivity of data to store in the cloud protected by these encryption keys.

KMS functionality applied to cloud service models

The KMS service model can be different from the cloud services it interacts with. For example, an IaaS KMS model can interact with SaaS data storage. Regardless of the KMS service model, the following subsections describe the KMS functionality required for each service model.

IaaS

Security capabilities involving a KMS that are essential in IaaS cloud services include authenticating predefined VM images, authenticating API calls to the VM management interface, and securing the communication of administrative operations on the VM instances. [3]

ATT&CK Tactic	Technique
Defense Evasion	Modify Cloud Compute Infrastructure [T1578]
Execution	Command and Scripting Interpreter: Cloud API [T1059.009]
Execution	Cloud Administration Command [T1651]

In the IaaS service model, customers must be able to securely administer the virtual machines, the applications running on the VMs, user communication with the VMs, and data storage. These operations will require asymmetric key pairs to perform digital signing, secure communication, and authentication. Symmetric keys may also be needed for encryption. [3]

Customers need to secure the private key of a public/private key pair on the customer's systems both at rest and in use. [3]

In an IaaS service model, symmetric keys used for file encryption can be stored on the customer’s site using an enterprise KMS. The customer encrypts the files and then stores them in the cloud.

D3FEND Tactic	Countermeasure
Platform Hardening	File Encryption [D3-FE]

PaaS

In the Platform as a Service (PaaS) model, customers must be able to securely interact with applications and store data. As with IaaS, these operations will require asymmetric key pairs to perform digital signing, secure communication, and authentication. Symmetric keys may also be needed for encryption. [3]

Customers should secure the private key of a public/private key pair on the customer’s systems both at rest and in use. [3]

SaaS

In the SaaS service model, the CSP is responsible for the secure interaction with the application. However, secure storage of data may be the customer’s responsibility, depending on the CSP’s service configurations. In this case, symmetric keys may be needed for encryption. [3]

MCAs may attempt to use compromised private keys and other secrets to forge credentials, such as session tokens or cookies, in order to gain access to the application. Customers need to secure the client-side private key of a public/private key pair on the customer’s systems both at rest and in use. The CSP must manage the server-side private key. All encryption keys are under the control of the CSP. [3]

ATT&CK Tactic	Technique
Credential Access	Forge Web Credentials [T1606]

Depending on the scale of application data that needs to be encrypted, encryption keys may need to reside with the CSP. If the selection of data to be encrypted varies, the encryption may have to take place on the customer end. [3]

Standards and certifications

There are established criteria for assessing CSP claims. These include determining if the CSP follows recommended standards and is participating in validation programs.

Key generation systems must be the most secure of all systems, and it is critical that these systems are correctly doing only what they are designed to do throughout their operational life. Verify that the CSP has implemented a key generation mechanism that conforms to National Institute of Standards and Technology (NIST) SP 800-90A Rev 1. [8]

Conformance testing for implementations can provide independent validation that a system meets certain requirements. [9] NIST has programs that provide validation testing of FIPS-approved and NIST-recommended cryptographic algorithms and modules. [10] Using a validated HSM-based key management system is a best practice. For CSPs that offer FIPS 140-3 validated hardware security module (HSM) based services, detailed information about the HSM and related security policy will be found on the NIST Cryptographic Module Validation Program (CMVP) Search page. [11]

In particular, integrity protection must be provided for all keys, and confidentiality protection must be provided for all secret keys. Appropriate confidentiality protection is provided for a key stored in a cryptographic module that conforms to FIPS 140-3 at a security level that is consistent with the impact level associated with the data to be protected by the key. [12] The CMVP certificate will give the overall security level and any security level exceptions. [13] The Federal Risk and Authorization Management Program (FedRAMP) validates that this requirement is met when authorizing cloud platforms for use by the federal government.

When selecting algorithms used for cryptographic purposes, such as key establishment, encryption, and digital signatures, NSA has provided recommendations in the Commercial National Security Algorithms (CNSA) Suite version 2.0 for quantum-resistant algorithms. [14] Organizations should look to meet these recommendations as much as possible. At a minimum, they should use CNSA Suite 1.0 recommended algorithms and have a plan for transitioning to CNSA Suite 2.0 quantum-resistant algorithms, where applicable, when they are available. Verify that the CSP’s KMS supports keys that meet the necessary requirements.

The National Information Assurance Partnership (NIAP) provides certifications and validation reports for products that are found to be compliant with documented security criteria. In 2021, NIAP established a cloud working group to determine an approach to certifying cloud service deployments. To check for any updates on this effort, visit NIAP's website. [15]

Best Practices

Encryption keys are an integral part of secure operations in the cloud. Organizations should adhere to the following best practices for cloud key management as relevant to their chosen KMS model:

- Keys generated on an HSM should never be exported in plain text.
- Understand where a key is stored at rest, whether in an HSM or in software, and how it is protected. Be sure key storage meets the use case requirement.
- Key policies establish access rules for keys. Key policies can be a combination of identity based, role based, and/or attribute based for a particular key. Different CSPs have different approaches.
- Be aware that there are usually lag times between setting a key policy and its effect due to latency.
- In general, key policies should have an implicit deny rule, meaning policy administrators must explicitly grant privileges for the key's use or access. Be aware that default key policies may differ depending on whether the key is created programmatically or via a management console.
- Enforce separation of duties so that no single person has all of the access required to perform a critical business function. For keys that protect sensitive resources/capabilities, separate the ability to manage keys from the ability to use keys for cryptographic operations. For key management roles, create more granular roles for stages of the key lifecycle. Consider how the defined roles will scale as the enterprise grows.
- Enforce least privilege so that each person has the minimum access required to perform their assigned duties.
- KMS APIs may add new operations over time. Granting access to all or a CSP defined category of API operations in a key policy may expand access for a role over time as well. This could lead to unintended access permissions. Therefore, access

policies should specifically grant access to each API operation needed, even if that includes all API operations that exist at a given point in time.

- When a key is destroyed, content that was encrypted with the key can no longer be decrypted. Ensure controls are in place to verify the key is no longer needed before destroying it.

Further guidance

Supplementary guidance is available at NSA Cybersecurity Advisories & Guidance. Particularly relevant ones include:

- Mitigating Cloud Vulnerabilities
- Top 10 Mitigation Strategies
- Advancing Zero Trust Maturity throughout the User Pillar
- Secure Data in the Cloud

Works cited

[1] Defense Information Systems Agency. Department of Defense Cloud Computing Security Requirements Guide, Version 1, Release 4. 2022. https://dl.dod.cyber.mil/wp-content/uploads/cloud/zip/U_Cloud_Computing_SRG_V1R4.zip

[2] National Security Agency. Cloud Security Basics. 2018. https://media.defense.gov/2019/Jul/16/2002158059/-1/-1/0/CSI-CLOUD-SECURITY-BASICS.PDF

[3] National Institute of Standards and Technology. NISTIR 7956: Cryptographic Key Management Issues and Challenges in Cloud Services. 2013. https://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7956.pdf

[4] International Organization for Standardization. ISO/IEC 27017:2015 Code of practice for information security controls based on ISO/IEC 27002 for cloud services. 2015. https://www.iso.org/standard/43757.html

[5] National Institute of Standards and Technology. NIST SP 800-192: Verification and Test Methods for Access Control Policies/Models. 2017. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-192.pdf

[6] International Organization for Standardization. ISO/IEC 19086-4:2019 Cloud computing - Service level agreement (SLA) framework - Part 4: Components of security and of protection of PII. 2019. https://www.iso.org/standard/68242.html

[7] National Institute of Standards and Technology. NIST IR 8006: Cloud Computing Forensic Science Challenges. 2020. https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8006.pdf

[8] National Institute of Standards and Technology. NIST SP 800-90A Rev. 1: Recommendation for Random Number Generation Using Deterministic Random Bit Generators. 2015. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90Ar1.pdf

[9] National Institute of Standards and Technology. Security Testing, Validation and Measurement. 2022. https://www.nist.gov/itl/csd/security-testing-validation-and-measurement

[10] National Institute of Standards and Technology. Cryptographic Module Validation Program. 2022. https://www.nist.gov/programs-projects/cryptographic-module-validation-program-cmvp

[11] National Institute of Standards and Technology. Cryptographic Module Validation Program Search page. 2024. https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search

[12] National Institute of Standards and Technology. NIST SP 800-57 Part 1 Rev. 5: Recommendation for Key Management: Part 1 - General. 2020. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf

[13] National Institute of Standards and Technology. Cryptographic Module Validation Program FIPS 140-3 Standards. 2024. https://csrc.nist.gov/projects/cryptographic-module-validation-program/fips-140-3-standards

[14] National Security Agency. Announcing the Commercial National Security Algorithm Suite 2.0. 2022. https://media.defense.gov/2022/Sep/07/2003071834/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS_.PDF

[15] National Information Assurance Partnership. Common Criteria Evaluation and Validation Scheme 2021 Report. 2021. https://www.niap-ccevs.org/Ref/Tracked/Progress_Report_2021.pdf

Disclaimer of endorsement

The information and opinions contained in this document are provided "as is" and without any warranties or guarantees. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement, recommendation, or favoring by the United States Government, and this guidance shall not be used for advertising or product endorsement purposes.

Trademarks

ATT&CK and MITRE and are registered trademarks of The MITRE Corporation. D3FEND is a trademark of The MITRE Corporation.

Purpose

This document was developed in furtherance of the authoring agencies' cybersecurity missions, including its responsibilities to identify and disseminate threats to National Security Systems, Department of Defense, and Defense Industrial Base information systems, and to develop and issue cybersecurity specifications and mitigations.

Contact

Cybersecurity Report Feedback: CybersecurityReports@nsa.gov
General Cybersecurity Inquiries: Cybersecurity_Requests@nsa.gov
Defense Industrial Base Inquiries and Cybersecurity Services: DIB_Defense@cyber.nsa.gov
Media Inquiries / Press Desk:
NSA Media Relations: 443-634-0721, MediaRelations@nsa.gov
CISA Media Inquiries: 703-235-2010, CISAMedia@cisa.dhs.gov