# Key Management for Public Cloud Migration

## Working Draft

**Authors:**

- Login to your Google account to access full editing permission.
- Change from Editing to Suggesting in the upper right of the Google doc to track each author's edits.
- CSA Technical Content Style Guide for consultation when writing.

*Please contact research-support@cloudsecurityalliance.org to request full access to author this document.*

**Reviewers/Visitors:**

- If you have a Google Account, please login before commenting. Otherwise, please note your name and affiliation in the comment you leave.
- Use the Comments or Suggesting features on Google docs to leave your feedback on the document. Suggestions will be written in and identified by your Google Account. To use the comments feature, highlight the phrase you would like to comment on, right click and select "Comment" (or Ctrl+Alt+M). Or, highlight the phrase, select "Insert" from the top menu, and select "Comment." All suggestions and comments will be reviewed by the editing committee.
- Please review content, logic and factual accuracy of the document. Grammar and spelling will be corrected later by a copy editor.

*For more information about Google's Comments feature, please refer to http://support.google.com/docs/bin/answer.py?hl=en&answer=1216772&ctx=cb&src=cb&cbid=-rx63b0fx4x0v&cbrank=1*

The permanent and official location for the Cloud Key Management Working Group is https://cloudsecurityalliance.org/research/working-groups/cloud-key-management

# Acknowledgments

## Lead Authors

Sunil Arora
Santosh Bompally
Rajat Dubey
Yuvaraj Madheswaran
Michael Roza

## Contributors

Marina Bregkou
Parthasarathy Chakraborty
Alex Rebo
Sam Pfanstiel

## Reviewers

Charan Akiri
Mahesh Prabu Arunachalam
Shruti Dhumak
Nielet D'mello
Rob Doyon
Debrup Ghosh
Kushal Kumar
Vaibhav Malik
Taresh Mehra
Prateek Mittal
Vani Murthy
Ratnangi Nirek
Govindaraj Palanisamy
Meghana Parwate
Neelesh Pateriya
Estevenson Solano
Chantal Spleiss
Sarah Templey
Kalpesh Vaid
Sudheer Vallandas
Rajashekar Yasani

## CSA Global Staff

Marina Bregkou

# Table of Contents

# 1. Introduction

Organizations are more dependent on technology than ever. More and more organizations are adopting public cloud technologies to take advantage of cloud computing agility and scalability. As part of public cloud[1] adoption, on-premises and data center applications are either modernized or lifted and shifted to a cloud environment, resulting in data migration from on-premises to cloud environments.

However, cloud computing has its challenges. The primary focus of this paper is securing the data during and after migration. Effective encryption and key management are critical controls for protecting the data stored in the cloud environment to ensure it is handled securely, mitigating the risk of unauthorized access and data breaches.

This paper documents best practices for managing encryption keys during data migration from on-premises environments to the cloud. It offers practical recommendations for planning and executing migration activities, addressing pre-migration, migration, and post-migration phases to ensure a secure transition and maintain organizational security.

For more detailed information on public cloud key management, please check the other CSA Key Management Working Group publications.

## 1.1 Purpose

This paper aims to provide comprehensive recommendations for managing encryption keys before, during, and after data migration from on-premises to cloud environments. It aims to:

- Outline best practices for implementing encryption to secure data in transit and at rest.
- Discuss key access control mechanisms to ensure only authorized entities can access encryption keys.
- Address additional necessary steps for managing encryption keys effectively throughout the migration process.
- Guide key rotation strategies to maintain the integrity and security of encryption keys over time.

## 1.2 Scope

The scope of this document is to provide key management considerations and best practices for managing keys while moving data from on-premises environments to the public cloud.

[1] Please visit the CSA Online Glossary for definition of terms.

This document covers key management considerations and best practices at various stages of the migration process, such as:

- **Migration Assessment and Planning**: Evaluate current encryption key management practices and their cloud application planning.
- **Data Preparation**: Ensure data is appropriately encrypted and key management protocols are established before migration.
- **Migration**: Implement key management activities during the data transfer to maintain data security and integrity.
- **Post-migration**: Validate that encryption keys are properly managed and maintained in the new cloud environment to ensure ongoing data security.

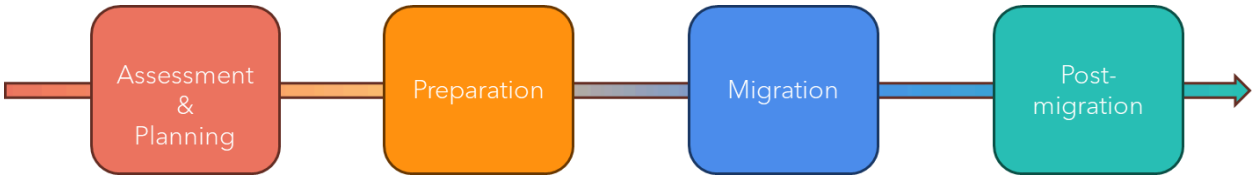


Figure 1. Stages of the Migration Process

## 1.3 Target Audience

- The primary target audience for this document includes Enterprise and Security Architects, IT and IS Professionals, Compliance Professionals, Application and Solution Architects, Cloud Security Consultants, Chief Information Security Officers, Risk Management Professionals, Cloud Architects, and DevOps Engineers.

- The secondary target audience encompasses Developers, Project Managers, System and Network Administrators, Operations Specialists, Business Continuity Planners, Auditors and Internal Reviewers, IT Project Managers, Database Administrators, and End-Users.

# 2. Assessment, Planning, and Data Preparation

The initial phase of a successful data migration involves a thorough assessment and strategic planning. This phase focuses on understanding the data landscape and key management practices, identifying gaps and potential risks, and formulating a comprehensive plan to mitigate those risks.

## 2.1 Assessment

Robust key management practices are crucial for data security during on-premises-to-cloud migration. This section outlines key management considerations and best practices across the various stages of the migration process.

### 2.1.1 Data Classification

- **Identify and Classify Data**: Determine appropriate encryption and key management strategies based on data sensitivity, integrity, availability, ownership, tenancy, and residency, including applicable regulatory requirements.
- **Inventory of Current Key Cryptosystem**[2]: Ensure an accurate inventory of all keys within the current cryptosystem, including key identifiers, purpose, algorithm, strength, approved usage, key management processes, and other applicable metadata. This inventory will be crucial in mapping the future state of such keys, including any key transformations or key encryption keys (KEKs) required to facilitate the migration process in the next phase.

### 2.1.2 Understanding Current (On-prem) and Future (Cloud) Key Management Practices

- **Inventory On-prem Key Management Solutions**: Identify capabilities (such as FIPS 140 level, encryption libraries, cipher suites, and connectivity options) and potential limitations or gaps (such as discrepancies between the capabilities, standards, and configurations of the current on-premises key management solution and the

[2] Please see Glossary at the end of this document.

requirements of the target cloud environment) in the organization's current key management solution. Assess its compatibility[3] with the target cloud environment.

- **Investigate Cloud Key Management Solution**: Review and evaluate the features, security, and compliance certifications of the CSP's Key Management Service (KMS), especially if they offer Bring Your Own Key (BYOK) services, HSM (Hardware Security Module), or external key management solutions [38]. BYOK, customer-managed keys, and external key management enables cloud customers to encrypt their data using keys provided by the customer and maintain complete control and management of their encryption keys [28]. Customers should ensure that the CSPs or external KMS meets their key management and operational requirements.
- **Integration and Interoperability**: Plan for seamless integration (straightforward integration with CSP key management services, interfaces, development, deployment, and maintenance) and key lifecycle management with the cloud provider's key management services.

### 2.1.3 Assess and Identify Access Control Requirements

Determine the access control mechanisms necessary to protect data and keys during migration. Determining the necessary access control mechanisms for protecting data and keys during migration from on-premises to the public cloud involves several key considerations that can be found in best practices and common frameworks such as the CSA Security Guidance, NIST SP 800-144, ISO 27017:2015, etc. [19][20][21]

### 2.1.4 Privacy, Security, and Compliance

- **Identify Compliance and Regulations**: Recognize and address regulatory requirements specific to your industry and business objectives, such as PCI DSS, HIPAA, GDPR, or internal privacy policies. Understand how the migration itself will alter your scope under all applicable standards and whether additional controls may apply to the migration itself (e.g., controls for conveyance of keys and key material). Prioritize

[3] Assessing the compatibility and effectiveness of the current on-premises key management solution in preparation for migration to a cloud environment may be performed according to the following assessment criteria: compliance and regulatory requirements, encryption standards and algorithms, cipher suite compatibility, interoperability, key management features and integration with cloud, performance and scalability, security controls, key backup and recovery strategies, cost analysis and resources location, vendor support and service level agreements (SLAs). [1], [3], [22].

strong encryption[4] for in-scope and regulated data and keys commensurate with the data classification of the protected information.

### 2.1.5 Version Control Tracking

- **Implement Version Control (NIST SP 800-130)**: In preparation for changes to the key management system according to industry standards, such as those outlined in NIST SP 800-130, establish a system for tracking changes and enabling rollback, if needed, to restore previous versions of applications, configurations, and keys in case of key corruption or compromise.

### 2.1.6 Backup Key Management Plan

- **Secure Key Backup (NIST SP 800-57)**: Before transferring data to the cloud, securely backup all encryption keys used to secure the data. Store backup copies in a secure location, such as another CSP region, a different cloud provider, or on-premises KMS or HSM.

### 2.1.7 Cost/Financial Considerations

Cloud migration costs should be considered due to various expenses such as re-architecture and training. The cost of data transfer can be significant. So, estimating expenses and managing the budget can be beneficial. In addition, key insights from industrial experts can help control spending and cost optimization.

### 2.1.8 Training

Personnel who have previously worked only with legacy on-prem technologies would need to be trained to work on cloud services and technologies in public or hybrid cloud environments. The training should be scheduled, budgeted, and allocated resources.

## 2.2 Planning

Key management throughout the data migration is crucial when transitioning from on-premises to the cloud. Consider the following aspects during planning and execution:

[4] "Strong encryption" is not a universally-defined term but is generally accepted to include the use of industry standard algorithms and encryption keys of sufficient bit strength to resist known attacks, including meet-in-the-middle and brute-force attacks, or render such attacks computationally infeasible.

- **Confirm Design of Target Key Management System (NIST SP 800-130)**: Implement any planned changes or implementation of the target key management system, incorporating any changes identified in the assessment stage, and in accordance with best practices related to performance, policy, access controls, key management, and disaster recovery. This system's design is outside this paper's scope but best practices may be found in resources such as NIST SP 800-130.
- **Cloud Contract Validation and SLA Review**: Ensure that cloud service contracts and SLAs include provisions for key recovery and auditing of operations related to encryption key management, covering key creation, invocation, deletion, and all lifecycle operations. Customers should choose a cloud service provider (CSP) that best meets their organizational security requirements, including applicable regulatory requirements.
- **Develop a Cutover Plan**: Incorporate key management tasks[5] into your cutover plan, such as generating new encryption keys for the cloud environment and securely transferring existing keys. For the key management tasks, please refer to this working group's previous publications on Key Management Lifecycle Best Practices and HSMaaS Use Cases, Considerations, and Best Practices.
- **Define Clear Objectives**: Include key management objectives in your cutover planning, such as securely transferring encryption keys to the cloud, key rotation, secure storage, and properly configuring encryption system components' access controls.
- **Risk Assessment and Mitigation**: Assess the risks associated with key management[6] during the transition to the cloud and develop mitigation strategies to address these risks.
- **Shared Responsibility Matrix**: Establish a clear understanding of the responsibilities between the CSP and the customer regarding encryption key management. This matrix should detail which tasks, such as key generation, storage, rotation, and revocation, are handled by the CSP and cloud customer.
- **Cloud Migration Tools**: Select proper cloud migration tools with built-in encryption capabilities. Many CSPs offer these tools or partner with vendors that provide secure data transfer solutions.

[5] "Key management tasks" refer to the specific activities and processes related to managing cryptographic keys during the transition from an on-premises environment to a cloud environment. The key management tasks that should be incorporated into the plan include: key generation, transfer and storage, key rotation, backup and recovery, access control configuration, key migration testing, and secure decommission and destruction of old encryption keys no longer needed in the on-premises environment to prevent unauthorized access or use. [22], [23].

[6] Main risks to consider: Key exposure, key loss or corruption, unauthorized access to keys, compromised KMSs, insecure key transmission, inadequate key rotation and expiration policies, key management complexity, regulatory compliance and legal risks, integration challenges with cloud key management services, data sovereignty and cross-border issues, insufficient monitoring and auditing, and human error. [1] [22] [24]

- **Migration Planning**
  - **Transition Planning**: Create a plan for transitioning keys from on-premises to the cloud.[7]
  - **Migration Window**: When working with the client during downtime, define a clear migration window. Many migrating clients may have a current product workload, which can only be migrated during downtime to support their current customers.
  - **Testing**: Various types of testing should be considered. Tests themselves should be reviewed with experts to ensure risk-based prioritisations. For example, regression testing is crucial to ensure that applications and services operate post-migration as they did pre-migration. Performance testing should unearth any latency and performance bottlenecks. These findings should be addressed before migration. Furthermore, tests should be conducted to ensure reliability and availability as well.
  - **Rollback Planning**: Define clear rollback strategy and procedures, including communication plan, roles and responsibilities, emergency contact details, and testing protocols as needed in case of data upload failures or unexpected issues during the migration process. This procedure includes reverting to the previous state of on-premises data and ensuring that any changes to encryption keys or configurations are rolled back to their original state for review and resolution before a re-run. Rollback procedures should be well-documented, tested, and regularly updated to account for cloud environment changes or data migration processes.
  - **Post-migration Key Lifecycle Management:** Establish clear procedures for managing keys after migration. This is important for maintaining security, ensuring compliance, and preventing vulnerabilities.
- **Communicate and Coordinate**: Ensure key management practices are communicated to all stakeholders involved in the cutover process and coordinate between teams responsible for key management and data migration.

## 2.3 Data Cleansing and Transformation/Preparation

Unclean and inconsistent data and associated keys and certificates can expose vulnerabilities (e.g., unauthorized access, meet-in-the-middle (MitM) attacks), compromising the system's confidentiality, integrity, and availability. Therefore, efficient measures must be in place to ensure

[7] Transition planning for moving keys from on-premises to the cloud includes: Assessment of the current key management, choosing the right cloud key management solution, mapping key management policies, data and key migration strategy, testing and validation of the new KMS, security and compliance review, risk mitigation strategy, execution and monitoring of the migration plan, training of the personnel, and documentation of the cloud key management, post-migration review. [1], [19],[21], [25], [26], [27].

that data is migrated securely and consistently to the cloud. This section outlines best data cleansing and transformation practices, deduplication, error correction, format standardization, and cloud database compatibility considerations to ensure a secure and successful cloud migration.

### 2.3.1 Deduplication, Error Correction, and Format Standardization

- Ensure data and datasets are classified per organization data classification standards. Organizations should have well-defined data security and data classification standards before data migration.
- Establish data accuracy, completeness, and consistency parameters such as file size, number of files, mandatory fields, and transactions. Data shall be validated before migrations to ensure that it can be verified post-migration.
- Identify and categorize obsolete datasets that no longer serve critical business or compliance purposes or require archiving to streamline migration complexity, reduce bandwidth usage, and optimize storage costs. Additionally, prioritize datasets that need immediate migration while allowing for the later transfer of less critical data.
- Adopt data filtering techniques, file-level encryption, anonymization, or selective migration to minimize data exposure and reduce the attack surface.
- If applicable, implement data validation techniques like checksums or hashes to ensure data integrity during transfer. This validation verifies that the data was not corrupted or tampered with during migration.
- Develop appropriate data cleansing processes and procedures to follow consistent rules and checks for data preparations.
- Create an inventory of encryption keys and classify them [1] based on usage, sensitivity level, and regulatory requirements.
- Identify expired and old keys. In addition, datasets that may need to be decrypted and re-encrypted before or after migration to the cloud environment must be identified.
- Identify and design a secure solution for storing keys securely in the cloud, using options such as cloud-based key management services or hardware security modules.
- Plan and design access controls based on least privilege access, role-based permissions, and audit mechanisms to monitor key usage and prevent unauthorized access.
- Plan for secure key distribution mechanisms and key rotation policies.
- Compile an inventory of SSL/TLS certificates to secure data transfers and communications during migration.
- Validate certificate authenticity, expiration dates, and certificate chains.
- Plan for automated certificate renewal processes and monitor expiration dates.

- Configure certificate revocation lists (CRLs) or the online certificate status protocol (OCSP) to revoke compromised or outdated certificates.

### 2.3.2 Cloud Database and Storage Compatibility Considerations

- Ensure the cloud database service supports required security functions such as authentication, authorization, data security functions, private access, BYOK, and customer-managed key management.
- Ensure the cloud database supports encryption and key functions such as key strength and cipher suites.
- Organizations may have detailed and specific database security logging and audit requirements, whereas cloud PaaS and SaaS databases may have limited monitoring and visibility.
- Ensure compatibility with Security Information and Event Management (SIEM) solutions to effectively monitor, identify, and address potential vulnerabilities within the cloud environment and database configuration.
- A cloud provider may offer various databases, such as relational, non-relational, key-value, and graph databases. Choose a cloud database service compatible with the existing on-premises database engine.
- When moving data to object storage services, verify compatibility with APIs, authentication, access control policies, metadata, versioning, lifecycle management, and other required security features.

## 2.4 Data Security and Key Management

Data encryption and key management are essential when uploading on-premises data to the cloud to ensure the security, confidentiality, and integrity of sensitive information

- Organizations should use encryption during data transfer to protect it from interception, unauthorized access, or tampering while uploading to the cloud.
- Creating secure, dedicated connections for data transfer between on-premises and the cloud helps minimize the risk of unauthorized access on shared networks. (Refer to section 3.1 for more details.)
- Once in the cloud, data should remain encrypted to prevent unauthorized access. Organizations can choose between server-side encryption offered by cloud providers or client-side encryption for enhanced security and control.

- Maintaining control over encryption keys is crucial. Organizations can opt for customer-managed keys where they generate, manage, and store the keys either on-premises or in a dedicated KMS in the cloud. This approach ensures compliance with industry-specific regulations, such as GDPR and HIPAA, which require proper encryption and key management practices when handling sensitive data in the cloud.
- Implement least privilege to ensure access to encryption keys is restricted and enforce separation of duties to minimize key exposure and mitigate risks of misuse.
- Encryption and key management enable secure collaboration and data sharing in the cloud while maintaining control over access rights through techniques like envelope encryption and key rotation, for example, by controlling access to wrapping keys.

# 3. Migration Execution and Management

This section dives into the execution phase of cloud migration. It covers secure methods for transferring data and encryption keys. Then, it reviews testing and validation strategies to guarantee a seamless transition. Finally, it addresses best practices for maintaining data and key integrity and functionality throughout migration.

## 3.1 Tools and Techniques for Data and Key Transfer

### 3.1.1 Various Techniques for Data and Key Transfer

Organizations have several options for securely transferring data and keys from on-premises environments to cloud providers. For example, latency-sensitive workloads involving data or keys may use a dedicated network connection to optimize performance and avoid the internet. Cloud providers offer dedicated connection services like AWS Direct Connect, Azure ExpressRoute, and Google Cloud Interconnect. While these services provide private, high-performance, and secure connections, minimizing the risk of data breaches or unauthorized access during transfers, they do not automatically encrypt data in transit by default. Additional measures, such as an AWS Site-to-Site VPN or other encryption methods, are needed to ensure complete protection.

### 3.1.2 Cloud-Native Approach

Cloud providers offer cloud-native data migration tools and services tailored to specific needs and functionalities within their respective ecosystems. These tools often utilize APIs and containerization technologies for efficient, secure, automated data and key migration. Examples

include AWS Database Migration Service (DMS) for seamless database migration, Azure Data Factory for orchestrating data movement across various sources, and Google Cloud Dataflow for building data pipelines for migration workflows.

### 3.1.3 Secure Transfer and Storage Options

Cloud providers offer services like AWS Snowball, Azure Data Box, and Google Cloud Transfer to securely transfer large datasets. These services utilize encrypted physical storage devices shipped between on-premises and cloud environments. Additionally, scalable and cost-effective object storage solutions like AWS S3, Azure Storage Account, and Google Storage Bucket provide flexible storage for various data types, including sensitive data and encryption keys.

### 3.1.4 Third-Party Migration Services

Organizations can securely leverage third-party migration tools and services to transfer data and encryption keys to the public cloud. These tools offer various functionalities, including:

- **End-to-end managed services**: These services handle the entire migration process.
- **Vendor-specific solutions**: These are tailored to specific cloud environments and may offer specialized data and key migration features. Examples include RiverMeadow and CloudEndure Migration.

### 3.1.5 Batch Transfer vs. Continuous Data Replication (CDR)

**Batch transfer** involves moving large volumes of data and keys (datasets) from an on-premises environment to the public cloud in discrete, pre-defined batches at specific points in time.

- **Batch example**: Daily or weekly production line data from various sensors and machines can be transferred in batches to the cloud for analysis and performance monitoring. This data is typically large but changes at predictable intervals, making batch transfer efficient.

**Continuous Data Replication (CDR)** involves near-instantaneous and automatic data and key (dataset) synchronization between on-premises environments and the public cloud. It continuously monitors for data changes and replicates them to the cloud in near or actual real-time, allowing a consistent copy of datasets across both environments.

- **CDR example**: User keys for a critical application must be continuously synchronized between on-premises and cloud environments to ensure uninterrupted access. CDR provides real-time replication, minimizing the risk of access disruptions due to critical discrepancies.

The table below illustrates some drivers to consider when selecting the most applicable transfer method. Note that it's not either/or. Some datasets (data or keys) can be batch-transferred, while others can be transferred using CDR. Locked accounts (e.g., inactive accounts) can prevent transfers from completing. It is recommended that these accounts be identified and the issues resolved (e.g., account deletion) before transfer.

| Driver | Batch Transfer | Continuous Data Replication (CDR) |
|---|---|---|
| **Size and Velocity** | Ideal for large, static datasets | Efficient for minor, frequently changing datasets |
| **Downtime** | Acceptable for planned downtime (maintenance) windows | Critical applications requiring minimal downtime |
| **Security Requirements** | Suitable for moderate security needs, if implemented correctly | Requires strong security measures due to continuous data flow |
| **Compliance Regulations** | Can meet compliance requirements with proper logging and data management | It is easier to demonstrate compliance due to constant data availability and auditability |
| **Technical Expertise** | Requires less technical expertise to set up and manage | Requires more technical knowledge for ongoing monitoring and maintenance |
| **Scalability** | Can scale by increasing batch size or frequency and can also adjust schedules and resources | More scalable due to continuous data movement but requires scaling infrastructure accordingly |
| **Cost** | Batch transfers may incur lower costs due to less frequent data movement. Also, consider resource allocation and scheduling overhead. | Continuous replication involves ongoing resource utilization. Costs depend on infrastructure, bandwidth, and cloud provider pricing models. |
| **Exposure** | Risk increases as more data are exposed simultaneously during batch transfers | Continuous data flow exposes data consistently, potentially increasing the risk |
| **Granular Control** | Provides flexibility to transfer specific keys or subsets of keys selectively. Allows fine-tuning of transfer parameters. | Continuous, fine-grained replication control and prompt synchronization of individual keys. Granularity ensures precise data availability. |

Table 1: Transfer Method Considerations

Regardless of the method chosen, use strong encryption for datasets both in transit and at rest. Implement strong access controls limiting access to authorized personnel.

### 3.1.6 Monitoring and Managing Data and Key Integrity During the Transfer

Monitoring and managing data integrity when migrating data and keys from on-premises to the cloud necessitates implementing rigorous technical strategies and practices. The primary objective is to guarantee data and key accuracy throughout the migration process.

### 3.1.7 Real-Time Monitoring

- **Dedicated migration monitoring tools**: These tools offer comprehensive dashboards to track various aspects of the migration in real-time.
- **Transfer rates:** Monitor the speed and progress of data and key transfer to identify potential bottlenecks or slowdowns (e.g., Mbps, transfer throughput).
- **Error logs and job statuses**: Analyze error logs for any issues arising during the migration, such as network errors, data or key validation failures, or job processing problems using monitoring services (e.g., ELK Stack).
- **Resource utilization**: Monitor cloud resources used during the migration, such as CPU, memory, and storage, to ensure optimal performance and identify potential resource constraints using cloud-native observability services (e.g., CloudWatch, Azure Monitor).
- **Integrate with existing monitoring systems**: Leverage existing monitoring tools for the on-premises infrastructure to improve oversight of the migration process and correlate any potential issues with other system activities.
- **Set up alerts and notifications**: Configure real-time alerts and notifications for critical events during migration. For example, exceeding error thresholds for data (e.g., more than 1% of data records contain errors) or key validation (e.g., more than 0.5% of keys fail validation), encountering unexpected delays, or experiencing resource exhaustion. Real-time alerts allow for prompt intervention and resolution of any issues.

### 3.1.8 Integrity Checks

- **Use checksum/hash verification throughout the process**: Calculate checksums (e.g., MD5, SHA-256) for data and keys on the source system, during transfer, and upon arrival in the cloud. Compare these checksums to ensure data and key integrity.
- **Employ digital signatures for enhanced security**: Implement digital signatures for sensitive data and encryption keys (e.g., RSA, DSA). Digital signatures allow for verification of both data and key integrity and origin, further safeguarding against unauthorized modifications.
- **Leverage data integrity features**: Cloud providers offer built-in features like object versioning and immutability (e.g., AWS S3 Object Lock, Azure Blob Versioning), which can provide additional protection against accidental or malicious data and key corruption.

## 3.2 Securing Migration Tools and Processes

Securing migration tools and processes is essential to ensuring the confidentiality and integrity of data and keys throughout migration from on-prem to the cloud. This process involves implementing measures to harden the software and utilities that migrate data and encryption keys:

- **Regularly evaluate and address vulnerabilities in migration tools**: Use established security frameworks, such as OWASP ASVS or the NIST Cybersecurity Framework, to guide your assessments. Conduct activities like penetration testing and automated vulnerability scanning, with a focus on remediating critical and high-severity issues.
- **Evaluate and test emerging migration tools:** Perform proof-of-concept (PoC) tests on new migration tools to assess and gain confidence in their security capabilities as they become available.
- **Harden applications, OS, network, etc., with secure configurations that follow industry benchmarks (Center for Internet Security (CIS) benchmark)**: Create a hardened baseline and scan periodically to ensure compliance.
- **Implement strong access controls**: Restrict unauthorized access to data and encryption keys (e.g., Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC) Policy-Based Access Control (PBAC), Privileged Access Management (PAM)).
- **Regularly update migration tools**: Keep migration software up to date (e.g., patch management, version control) to address emerging security threats.
- **Use encryption protocols**: Implement protocols explicitly designed for safeguarding sensitive data and encryption keys (e.g., TLS 1.2 (conversion to improved TLS 1.3 is recommended above), AES-256).
- **Ensure strict compliance with standards and best practices:** Follow industry standards (e.g., ISO 27001, NIST) throughout the data and key migration process.
- **Establish a monitoring system**: Implement a system capable of real-time detection and response to security incidents (e.g., SIEM, IDS/IPS).
- **Regularly validate security measures**: Focus on data and key protection to proactively address potential vulnerabilities (e.g., security audits, compliance checks).

## 3.3 Testing and Validation

### 3.3.1 Conduct Unit, Integration, and End-to-End Testing

Testing ensures the security and integrity of data and keys when migrating from on-premises to the public cloud. Conducting thorough testing at various process stages is essential to identify and address any issues before deployment. Also, with rigorous unit, integration, and end-to-end testing, an organization can mitigate risks associated with key management, ensuring compliance with security standards and safeguarding sensitive information against unauthorized access or data breaches.

**Integration testing** is critical when integrating on-premises systems with cloud services to ensure that different components work together seamlessly. The scope involves verifying the interaction between on-premises and cloud components, focusing on data integrity, security, and

communication. Thorough integration testing helps detect and address issues arising from integration points, ensuring a smooth and reliable transition during migration.

Examples:

- Check the performance of real-time data transfer by evaluating the speed and reliability of data transfer between on-premises and cloud servers (e.g., Mbps, latency).
- Monitor the interaction between different components or services by validating their interoperability using tools like those made for API testing (e.g., Postman, SoapUI).
- Verify the seamless integration of encryption and decryption functionalities to ensure the proper integration with other application components, such as data transfer protocols (e.g., HTTPS, FTPS) and cloud storage services (e.g., AWS S3, Google Cloud Storage).
- Validate end-to-end encryption workflows by testing key exchange protocols (e.g., Diffie-Hellman, RSA) and cryptographic algorithms (e.g., AES-256, RSA-2048) to ensure data confidentiality during transit and storage.
- Consider automated testing frameworks and continuous integration pipelines to streamline integration testing processes and identify integration issues early in the development lifecycle (e.g., Jenkins, Travis CI).

**End-to-end testing (E2E)** validates the entire system, including cloud services, from a user's perspective during migration. Its scope involves simulating user interactions across the entire application stack to ensure smooth transitions, data flow, and external service calls. Thorough E2E testing verifies that the application behaves correctly, considering external dependencies such as databases, APIs, and cloud services. Regression is crucial for end-to-end testing to ensure that applications and services operate post-migration as they did pre-migration. Performance testing should unearth any latency and performance bottlenecks. These findings should be addressed before migration. Furthermore, tests should be conducted to ensure reliability and availability as well.

Examples:

- Automate the scripting process for data migration, key rotation using cryptographic libraries (e.g., OpenSSL), and updating access control based on authorization methods (e.g., RBAC, ABAC, PBAC, PAM).
- Validate that cloud-based APIs function correctly, handling user authentication using OAuth protocols and returning data in expected formats (e.g., JSON, XML).
- Ensure seamless user authentication and authorization across on-premises and cloud components using single sign-on (SSO) solutions.
- Evaluate the entire data uploading process, from data acquisition on-premises to storage in the public cloud (e.g., AWS S3, Azure Blob Storage), verifying system behavior under real-world conditions.
- To maintain data security, validate the complete encryption lifecycle, including key generation, distribution, rotation, and revocation (e.g., PKI, AES).

- Test cases should include various data types (e.g., text, binary), file formats (e.g., CSV, XML), and transfer methods (e.g., SFTP, HTTPS) to assess the resilience of key management mechanisms across different scenarios.
- Perform scalability and performance testing (e.g., load testing, stress testing) to ensure key management processes can handle varying workloads and data volumes effectively without compromising security or performance.

# 4. Transition and Optimization

Data migration from on-premises environments to the public cloud does not conclude with the successful transfer of data. The post-migration phase is critical to ensuring the cloud infrastructure operates efficiently and securely. This phase involves best practices for managing encryption keys and other security measures to safeguard sensitive data. Key focus areas include performance monitoring, compliance validation, incident response, and cost optimization to ensure the cloud environment meets operational expectations. Additionally, continuous improvement and adaptation strategies are essential to refining and evolving key management processes in response to dynamic business needs and technological advancements.

## 4.1 Post-Migration Monitoring, Optimization, and Incident Response

### 4.1.1 Performance Monitoring and Cost Optimization

- **Effective Performance Monitoring**: Implementing effective performance monitoring and cost optimization are essential when transferring on-premises data to the public cloud to ensure seamless data transfer, optimal resource utilization, and alignment with business needs. Key considerations include tracking usage patterns, identifying bottlenecks, right-sizing resources, and leveraging cloud provider tools for real-time monitoring and troubleshooting. Optimization strategies include instance right-sizing, storage configuration optimization, data compression, storage tiering, and selecting cost-effective storage options.
- **Continuous Monitoring and Optimization**: Continuously monitor and optimize to ensure the cloud environment delivers expected benefits while aligning costs with business needs and budgets. Real-time monitoring enables timely detection of bottlenecks and issues, allowing for prompt optimization. Employing cloud-provided key management services or integrating external solutions ensures data encryption, access control, and compliance with regulatory requirements, mitigating data breaches and unauthorized access risks. By implementing cost monitoring and tagging resources for tracking, organizations can make informed decisions regarding resource utilization and align cloud costs with business needs and budgets, ensuring a cost-effective cloud deployment.
- **Proactive Key Monitoring**: Continuously monitor for signs of key misuse or anomalies in access patterns using cloud-native or third-party security tools.

- **Key Rotation and Revocation**: Establish automated key rotation schedules to minimize the risk of key compromise. Implement immediate key revocation procedures if a breach is detected.
- **Incident Response Plan**: Develop and maintain a detailed incident response plan specific to key management. This plan should include immediate containment procedures, key revocation, data encryption reassessment, and communication protocols to notify stakeholders.
- **Post-Incident Analysis**: Conduct root cause analysis following any key-related incident to identify vulnerabilities and implement corrective actions to prevent future occurrences.

### 4.1.2 Continuous improvement and adaptation

To effectively implement continuous improvement and adaptation in key management practices, organizations must adopt a proactive approach that involves the following key elements:

- **Regular Audits and Reviews**: NIST standards, such as SP 800-53 and SP 800-171, outline security and privacy controls that organizations must implement to comply with various regulatory requirements. Regular audits and reviews ensure that organizations remain compliant with these standards by validating the implementation and effectiveness of key management controls. These assessments should encompass the technical aspects of key storage and encryption and the policies and procedures governing their usage.
- **Current Industry Standards and Best Practices**: The landscape of cloud security is constantly evolving, with new threats emerging regularly. Active participation in organizations such as the Cloud Security Alliance provides valuable insights into the latest industry standards and best practices. By staying informed, organizations can adapt their key management strategies to align with the most effective security measures.
- **Automation and Orchestration**: Automation plays a crucial role in streamlining key management processes and ensuring consistency across the board. Implementing automated key rotation, for instance, reduces the risk of key exposure due to human error or oversight. Additionally, orchestration tools can help orchestrate the deployment and management of encryption keys across multiple cloud environments, simplifying the complexity of hybrid or multi-cloud architectures.
- **Regular Penetration Testing and Red Teaming**: To ensure the effectiveness of key management practices, organizations should regularly subject their systems to penetration testing and red team exercises. These simulations help identify potential vulnerabilities and weaknesses in the infrastructure, allowing for proactive remediation before malicious actors can exploit them.

# 5. Conclusion and Future Outlook

## 5.1 Conclusion

Migrating data from on-premises infrastructure to a public cloud environment is a complex process that demands careful planning, proper execution, and ongoing management. Effective key management plays a critical role in ensuring the security and accessibility of data exclusively to authorized entities. This whitepaper presented a comprehensive overview of best practices for key management during public cloud migration, addressing various stages from assessment and planning to post-migration optimization.

### 5.1.1 Key Takeaways

**Here are some of the key takeaways of this whitepaper:**

- **Migration Assessment**: A thorough assessment and strategic planning phase are essential for seamless data migration and efficient key management.
- **Strong Key Management**: A robust Key Management System (KMS) is crucial for generating, storing, rotating, and controlling access to encryption keys. Cloud providers offer KMS solutions, but organizations can leverage on-premises or hybrid KMS options.
- **Inventory Management**: Create an inventory of the data and encryption keys, and classify them based on usage.
- **Key Rotation**: Identify the older and expired keys. Regularly rotating encryption keys minimizes the risk of compromise, even if a key is exposed.
- **Least Privilege Access**: Granting the least privilege access to encryption keys ensures that only authorized personnel can manage and use them.
- **Secure Transfer Methods**: To safeguard data during upload, utilize secure transfer methods like encrypted channels and protocols, dedicated network connections, or physical transfer appliances.
- **Cloud-Native Security Features**: Cloud providers offer various security features, such as object-level encryption and access controls, to enhance data protection further.

## 5.2 Future Outlook

Data security and key management for cloud migrations are constantly evolving. Here are some trends to consider for the future:

- **Integration with Data Migration Tools**: Tighter integration between KMS and data migration tools can streamline secure data movement to the cloud. Integrated tools could

involve automated key generation and association with specific datasets during migration.

- **AI-Powered Threat Detection**: AI algorithms can analyze data transfer patterns and key usage during migration to detect anomalies that might indicate unauthorized access attempts or potential security breaches.
- **Zero Trust for Data Migration**: Zero trust principles apply to data and key migration. For example:
    - **Multi-Factor Authentication (MFA)**: MFA is required for all access attempts to KMS and data migration tools during migration.
    - **Just-in-Time (JIT) Access**: This entails granting temporary access to keys and data only for the duration necessary to complete specific migration tasks.
    - **Continuous Monitoring**: Monitor data transfer activity and key usage during migration to identify suspicious behavior and potential security threats.
- **Quantum-Resistant Cryptography**: As quantum computing advances, organizations may need to transition to quantum-resistant cryptography algorithms and quantum key distribution to protect keys during future migrations. [30]

By staying informed about these trends and implementing best practices for key management during data migration, organizations can ensure a secure and efficient transition to the public cloud.

# 6. References


1. National Institute of Standards and Technology (NIST). NIST Special Publication 800-57 Part 1 Revision 5. (2020). Recommendation for Key Management: Part 1 – General https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf

2. National Institute of Standards and Technology (NIST). NIST Special Publication SP 800-171. (2015). *NIST Special Publication 800-171: Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations*. Gaithersburg, MD: National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-171r2

3. National Institute of Standards and Technology (NIST). NIST Special Publication 800-130. (2013). A Framework for Designing Cryptographic Key Management Systems. SP 800-130, A Framework for Designing Cryptographic Key Management Systems | CSRC

4. Amazon Web Services (AWS). (n.d.). *AWS Direct Connect User Guide*. Available at: AWS Direct Connect

5. Microsoft. (n.d.). *Azure ExpressRoute: Introduction*. Available at: Azure ExpressRoute Overview: Connect over a private connection | Microsoft Learn

6. Google Cloud. (n.d.). *Google Cloud Interconnect: Overview*. Available at: Cloud Interconnect overview

**Cloud-Native Data Migration Tools and Services**

7. Amazon Web Services (AWS). (n.d.). *AWS Database Migration Service (DMS)*. Available at: AWS Database Migration Service

8. Microsoft. (n.d.). *Azure Data Factory*. Available at: Azure Data Factory - Data Integration Service | Microsoft Azure

9. Google Cloud. (n.d.). *Google Cloud Dataflow*. Available at: Dataflow: streaming analytics | Google Cloud

10. Device42. (2024, March 22). Cloud Migration Best Practices: A Comprehensive Guide - Device42. Cloud Migration Best Practices: A Comprehensive Guide - Device42

11. Sighom, J.R.N., Zhang, P. and You, L. (2017) Security Enhancement for Data Migration in the Cloud. Future Internet, 9, 23. Security Enhancement for Data Migration in the Cloud

12. Suthar, K. and Patel, J. (2015) EncryScation: A Novel Framework for Cloud IAAS, DAAS Security Using Encryption and Obfuscation Techniques. Proceedings of the 2015 5th Nirma University International Conference on Engineering (NUiCONE), Ahmedabad, 26-28 November, 2015.

EncryScation: A novel framework for cloud IaaS, DaaS security using encryption and Obfuscation techniques | IEEE Conference Publication

13. Hussein, A. A. (2021). Data Migration Need, Strategy, Challenges, Methodology, Categories, Risks, Uses with Cloud Computing, and Improvements in Its Using with Cloud Using Suggested Proposed Model (DMig 1). Journal of Information Security, 12(01), 79–103. https://doi.org/10.4236/jis.2021.121004

14. Iqbal, A., & Colomo-Palacios, R. (2019). Key Opportunities and Challenges of Data Migration in Cloud: Results from a Multivocal Literature Review. Procedia Computer Science, 164, 48–55. Key Opportunities and Challenges of Data Migration in Cloud: Results from a Multivocal Literature Review - ScienceDirect

15. Amazon Web Service. (n.d.). What is ELK Stack? What is ELK stack? - Elasticsearch, Logstash, Kibana Stack Explained - AWS

16. Jenkins. (n.d.). Jenkins. https://www.jenkins.io

17. Travis CI. (n.d.). *Test and deploy with confidence*. https://www.travis-ci.com/

18. National Institute of Standards and Technology (NIST). (n.d.). COMPUTER SECURITY RESOURCE CENTER. Cryptographic system. cryptographic system (cryptosystem) - Glossary | CSRC

19. Cloud Security Alliance (CSA). (2017). *Security Guidance for Critical Areas of Focus in Cloud Computing v4.0*. Security Guidance for Critical Areas of Focus in Cloud Computing | CSA

20. National Institute of Standards and Technology (NIST). (2011). *Guidelines on Security and Privacy in Public Cloud Computing*. Special Publication 800-144. NIST SP 800-144, Guidelines on Security and Privacy in Public Cloud Computing

21. International Organization for Standardization (ISO). (2015). *ISO/IEC 27017:2015 - Information technology - Security techniques - Code of practice for information security controls based on ISO/IEC 27002 for cloud services*. ISO/IEC 27017:2015 - Information technology — Security techniques — Code of practice for information security controls based on ISO/IEC 27002 for cloud services

22. Cloud Security Alliance. (2023.). *Key Management Lifecycle Best Practices*. Retrieved from Key Management Lifecycle Best Practices | CSA

23. Cloud Security Alliance. (2024). *HSM as a Service: Use Cases, Considerations, and Best Practices*. HSM-as-a-Service Use Cases and Best Practices | CSA

24. Amazon Web Services (AWS). (2023). *Operational best practices for ENISA cybersecurity guide for SMEs*. Operational Best Practices for ENISA Cybersecurity guide for SMEs - AWS Config

25. Amazon Web Services (AWS). (2023). *Best Practices for Managing AWS KMS Keys.* Available at: Security best practices for AWS Key Management Service

26. Microsoft. (2023). *Azure Key Vault Security Overview.* [online] Available at: Azure Key Vault security overview | Microsoft Learn

27. Google Cloud. (2023). *Best Practices for Cloud Key Management.* [online] Available at: https://cloud.google.com/kms/docs/best-practices

28. Thales Group. (n.d.). What is Bring Your Own Key (BYOK)? What is Bring Your Own Key (BYOK) Encryption?

29. Cloud Security Alliance. (2024). Multi-Cloud KMS (Work in progress) https://docs.google.com/document/d/1vOAQn0ykCD-q6UiTo5gHkNoHppdyMiNVVOP-gO9VPLg/edit

30. National Security Agency (NSA) (n.d.) *Quantum Key Distribution (QKD) and Quantum Cryptography (QC)*. Quantum Key Distribution (QKD) and Quantum Cryptography QC

**Secure Transfer and Storage Options**

31. Amazon Web Services (AWS) Snowball. (n.d.). *Offline data transfer device, petabyte - AWS Snowball*. Retrieved from https://aws.amazon.com/snowball/

32. Microsoft Azure. (n.d.). *Azure Data Box – Secure, offline, and online data transfer*. Retrieved from https://azure.microsoft.com/en-us/products/databox

33. Google. Google Cloud. (n.d.). *Storage Transfer Service*. https://cloud.google.com/storage-transfer-service

**Object Storage Solutions**

34. Amazon Web Services (AWS). (n.d.). *Amazon S3 – Cloud object storage*. https://aws.amazon.com/s3/

35. Microsoft. (n.d.). *Azure Storage account overview*. https://learn.microsoft.com/en-us/azure/storage/common/storage-account-overview

36. Google Cloud. (n.d.). *Cloud Storage classes*. https://cloud.google.com/storage/docs/storage-classes

**Third-Party Migration Services**

37. RiverMeadow. (n.d.). *Workload mobility, multi-cloud migration, OS modernization*. https://www.rivermeadow.com/

38. CloudEndure. (n.d.). *Disaster recovery and cloud migration*. https://www.cloudendure.com/

39. Google Cloud. (n.d.). *External Key Manager (EKM) documentation*. Retrieved from https://cloud.google.com/kms/docs/ekm

40. Amazon Web Services (AWS). (n.d.). *AWS KMS: Using an external key store (XKS)*. https://docs.aws.amazon.com/kms/latest/developerguide/keystore-external.html

# Glossary

Terms from the CSA Glossary (main/primary):

- **Cryptosystem**: A cybersecurity (CS) item to provide a single means of encryption or decryption [18]
- **Cutover plan**: A cutover plan is a detailed strategy that outlines the steps and procedures required to transition with minimal disruption. In the context of migrating data and encryption keys to the cloud, a cutover plan ensures that all critical tasks are identified, scheduled, and executed in a controlled manner to maintain security, integrity, and availability throughout the process.

Terms from the CSA Quantum-Safe Security Glossary:

- **Quantum key distribution**: Quantum key distribution is an example of quantum cryptography that allows the information-theoretically secure distribution of keys between two spatially separate parties connected by an insecure optical channel. There are two complementary approaches to QKD: (1) discrete variable quantum key distribution (DVQKD) uses single-photons or weak coherent states and single-photon detectors, and (2) continuous-variable quantum key distribution (CVQKD), which uses coherent or squeezed states of light and homodyne detectors. Both continuous and discrete approaches have been experimentally demonstrated; just as importantly, both have been proven to be information-theoretically secure.

# Cloud Providers Migration Tools

| Cloud Provider | Cloud-Native Data Migration Tools | Secure Transfer and Storage Options |
|---|---|---|
| **Amazon Web Services (AWS)** | AWS DMS (Data Migration Service)<br>https://aws.amazon.com/dms/<br><br>AWS Application Migration Service<br>https://aws.amazon.com/application-migration-service/ | AWS Snowball (physical transfer)<br>https://docs.aws.amazon.com/snowball/latest/developer-guide/getting-started.html<br><br>AWS Direct Connect (dedicated network)<br>https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html<br><br>Amazon S3 (object storage)<br>https://aws.amazon.com/s3/ |
| **Microsoft Azure** | Azure Migrate<br>https://learn.microsoft.com/en-us/azure/dms/<br><br>Azure Database Migration Service<br>https://learn.microsoft.com/en-us/azure/dms/<br><br>Azure Data Factory<br>https://learn.microsoft.com/en-us/azure/data-factory/ | Azure Data Box (physical transfer)<br>https://learn.microsoft.com/en-us/azure/databox/<br><br>Azure ExpressRoute (dedicated network)<br>https://learn.microsoft.com/en-us/azure/expressroute/expressroute-introduction<br><br>Azure Storage Account (object storage)<br>https://azure.microsoft.com/en-us/pricing/details/storage/blobs/ |
| **Google Cloud Platform (GCP)** | Cloud Migrate<br>https://cloud.google.com/architecture/migration-to-gcp-getting-started<br><br>Cloud Dataflow<br>https://cloud.google.com/dataflow/docs | GCP Transfer Appliance (physical transfer)<br>https://cloud.google.com/storage-transfer-service<br><br>Google Cloud Interconnect (dedicated network)<br>https://cloud.google.com/network-connectivity/docs/interconnect/concepts/overview<br><br>Google Cloud Storage Bucket (object storage)<br>https://cloud.google.com/storage/pricing |
| **IBM Cloud** | IBM Cloud Database Migration Service (to various cloud targets)<br>https://www.ibm.com/docs/en/db2/11.5?topic=db2-migration-service<br><br>IBM DataStage (data integration platform)<br>https://www.ibm.com/products/datastage | IBM Aspera for Cloud (high-speed transfer)<br>https://www.ibm.com/downloads/cas/WAOOY7R2 |

# Third-Party Providers Migration Tools

| Vendor | Functionality | Key Features |
|---|---|---|
| **Informatica Cloud Data Integration (CDI)** | Integrates data from various sources https://www.informatica.com/products/cloud-data-integration.html | 1. Supports on-premises, cloud, and hybrid environments<br>2. Offers data cleansing, transformation, and mapping capabilities<br>3. Provides pre-built connectors for various applications and databases |
| **Talend Open Studio (Open-Source Option)** | Open-source ETL platform https://www.talend.com/products/talend-open-studio/ | 1. Enables data extraction, transformation, and loading for data migration<br>2. Offers a visual interface for building data pipelines<br>3. Integrates with various cloud platforms and data sources |
| **Xplenty** | Cloud-based data integration platform https://try.xplenty.com/software-testing-help/ | 1. Simplifies data movement between diverse sources and cloud platforms<br>2. Provides a user-friendly interface for data migration workflows<br>3. Offers scheduling and automation capabilities for data pipelines |
| **Flyway (Open-Source Option)** | Open-source database version control tool https://www.red-gate.com/products/flyway/ | 1. Manages database schema changes during cloud migrations<br>2. Ensures consistency and repeatability across database environments<br>3. Provides rollback capabilities for managing potential migration issues |
| **RiverMeadow** | Cloud migration and management platform https://www.rivermeadow.com/ | 1. Offers end-to-end managed services for data and application migration to the cloud<br>2. Supports various cloud platforms like AWS, Azure, and GCP<br>3. Provides tools for data security and compliance management |
| **CloudEndure Migration** | Cloud migration platform https://aws.amazon.com/marketplace/seller-profile?id=e54656a2-d5f7-45b7-af6d-9d06ac25b203 | 1. Specializes in migrating workloads to AWS and Azure<br>2. Enables continuous data replication to minimize downtime during migration<br>3. Offers automated rollback and recovery capabilities |