Service Name

AWS Key Management Service

Service Overview

AWS Key Management Service (KMS) is a fully managed cryptographic service that enables customers to create and control encryption keys used to protect their data. The service operates using FIPS 140-3 Security Level 3 validated hardware security modules (HSMs) and provides centralized key management across AWS services and applications. KMS offers traditional key management services integrated with AWS, handling infrastructure management including availability, physical security, logical access control, and maintenance. Keys are protected within HSMs and never leave the service unencrypted, ensuring maximum security for data protection both at rest and in transit.

Key Use Cases

- Data encryption at rest: Protecting sensitive data stored in AWS services like S3, EBS, RDS, and DynamoDB using customer-managed or AWS-managed keys
- Client-side encryption: Using envelope encryption with AWS Encryption SDK to protect data before sending it to AWS services
- Digital signing and verification: Creating asymmetric keys for signing documents, APIs, and code, enabling authentication and integrity verification
- Compliance and regulatory requirements: Meeting standards like PCI DSS, HIPAA, and GDPR through centralized key management and comprehensive audit logging
- Multi-region data protection: Using multi-region keys to encrypt data across multiple AWS regions while maintaining consistent access controls

Features

- **Symmetric Encryption Keys**: 256-bit AES keys for encrypting and decrypting data up to 4 KB directly or for envelope encryption of larger datasets
- **Asymmetric Keys**: RSA, ECC, ML-DSA, and SM2 key pairs for encryption/decryption, digital signing, and key agreement operations
- **HMAC Keys**: Symmetric keys for generating and verifying hash-based message authentication codes for data integrity
- **Hardware Security Modules**: FIPS 140-3 Security Level 3 validated HSMs protecting keys with tamper-evident and tamper-resistant hardware
- **Automatic Key Rotation**: Annual automatic rotation for AWS-managed keys and configurable rotation for customer-managed keys
- **Multi-Region Keys**: Keys that can be replicated across multiple AWS regions for global applications and disaster recovery

- **Custom Key Stores**: Integration with AWS CloudHSM clusters or external key managers for additional control and compliance
- **Grants and Key Policies**: Fine-grained access controls using IAM policies, key policies, and temporary grants for delegation
- **CloudTrail Integration**: Comprehensive logging of all key usage and management operations for audit and compliance
- **Post-Quantum Cryptography**: Support for ML-DSA quantum-resistant digital signature algorithms for future-proof security

Value Proposition

AWS KMS provides unparalleled security through FIPS 140-3 validated HSMs while eliminating the complexity and cost of managing cryptographic infrastructure. Unlike self-managed solutions, KMS offers seamless integration with over 100 AWS services, automatic scaling, and built-in high availability across multiple regions:

- The service reduces operational overhead by handling key lifecycle management, rotation, and backup automatically
- KMS provides superior cost efficiency with pay-per-use pricing, no upfront hardware costs, and a generous free tier
- The centralized audit trail through CloudTrail integration simplifies compliance reporting, while envelope encryption enables protection of unlimited data volumes without performance impact

Service Integration

AWS KMS integrates natively with over 100 AWS services for seamless encryption. Core integrations include Amazon S3 for object-level encryption, Amazon EBS for volume encryption, Amazon RDS for database encryption, and Amazon DynamoDB for table-level protection:

- Container services like Amazon ECS and EKS use KMS for secrets management, while Amazon CloudTrail, AWS Config, and AWS Systems Manager Parameter Store rely on KMS for log and configuration encryption
- Analytics services including Amazon Redshift, Amazon EMR, and Amazon Athena leverage KMS for data warehouse and big data encryption
- The AWS Encryption SDK enables client-side encryption across applications, and services like AWS Lambda, Amazon API Gateway, and AWS Step Functions use KMS for serverless application security

Onboarding and Offboarding

Getting started with AWS KMS requires minimal setup through the AWS Management Console, CLI, or SDKs. Customers begin by creating their first KMS key, selecting between symmetric encryption, asymmetric signing, or HMAC keys based on use case:

- The service provides comprehensive SDK examples in Java, Python, and PHP for common operations like key creation, encryption, and policy management
- AWS offers hands-on labs, documentation, and getting started guides to accelerate adoption
- Integration with existing AWS services is typically achieved by specifying a KMS key ARN in service configurations
- The AWS KMS console provides intuitive key management, policy configuration, and monitoring capabilities for both technical and non-technical users

Getting Started Guide:
https://docs.aws.amazon.com/kms/latest/developerguide/example_kms_Scenario_Basics_section.html

Data Removal

AWS KMS supports secure key deletion with a mandatory 7-30 day waiting period to prevent accidental data loss. During this period, keys cannot be used for cryptographic operations but can be recovered if needed. For immediate cessation of key usage, customers can disable keys rather than delete them. When deleting imported key material, the key becomes immediately unusable but can be restored by re-importing the same material. Multi-region keys require deletion from the primary region to remove all replicas. Customer data encrypted with KMS keys remains accessible until the key is permanently deleted.

Backup/Restore and Disaster Recovery

AWS KMS provides built-in disaster recovery through automatic replication of keys and policies across multiple Availability Zones within a region. Multi-region keys enable cross-region replication for global disaster recovery scenarios. The service maintains durable storage of key metadata and policies with 99.999999999% durability. Key rotation creates new key versions while maintaining access to previous versions for decrypting older data. AWS Backup integrates with KMS to encrypt backup data, and customers can use customer-managed keys with AWS Backup's logically air-gapped vaults for additional protection.

Backup Capabilities

AWS KMS keys are automatically backed up across multiple Availability Zones with high durability storage. The service maintains historical key versions through automatic and on-demand rotation, enabling recovery of data encrypted with previous key versions:

- Multi-region keys provide cross-region backup capabilities for disaster recovery
- AWS KMS integrates with AWS Backup service, allowing customers to use customer-managed keys to encrypt backup vaults and recovery points

- Key metadata, policies, and grants are automatically replicated and maintained by the service infrastructure

Disaster Recovery

AWS KMS supports comprehensive disaster recovery through multi-region key replication and automatic failover capabilities. The service maintains keys across multiple Availability Zones within regions and supports cross-region replication via multi-region keys:

- Integration with AWS Application Recovery Controller enables automated disaster recovery orchestration
- KMS provides consistent global endpoints and automatic scaling during regional failures
- The service's distributed architecture ensures continued operation even during significant infrastructure events, with sub-second recovery times for most operations

Recovery Objectives

AWS KMS enables customers to achieve aggressive RPO and RTO objectives through multi-region keys and real-time replication. Multi-region keys provide near-zero RPO by maintaining synchronized copies across regions. RTO targets of minutes can be achieved through automatic failover mechanisms and globally distributed endpoints. The service's high availability architecture with 99.9% SLA ensures minimal downtime. Integration with AWS Application Recovery Controller automates recovery processes, while envelope encryption patterns enable rapid recovery of large datasets without re-encryption.

Service Constraints

Service Quotas: https://docs.aws.amazon.com/general/latest/gr/kms.html

FAQ: https://aws.amazon.com/kms/faqs/

Technical Requirements

Service Documentation: https://docs.aws.amazon.com/kms/latest/developerguide/

User Guide: https://docs.aws.amazon.com/kms/latest/developerguide/overview.html

API Reference: https://docs.aws.amazon.com/kms/latest/APIReference/

Pricing Overview

Please see the AWS UK G Cloud 15 Pricing Document accompanying this service in the Digital Marketplace.

Public Pricing: https://aws.amazon.com/kms/pricing/

Cost Optimization

AWS KMS offers several unique cost optimization opportunities: utilize the 20,000 free requests per month across all regions; implement data key caching with the AWS Encryption SDK to reduce API calls by up to 99%; use Amazon S3 bucket keys to reduce KMS requests for S3 encryption by sharing data keys across objects; leverage AWS-managed keys when customer control isn't required as they incur no monthly charges; optimize by using a single KMS key across multiple AWS services rather than creating separate keys; consider geographic placement as cross-region key usage incurs additional charges; implement request batching and connection pooling to minimize API overhead.

Savings Considerations

Primary cost savings strategies include leveraging AWS-managed keys for standard use cases eliminating $1/month per key charges; implementing AWS Encryption SDK with data key caching reducing request volumes by 90%+; using S3 bucket keys sharing encryption keys across objects reducing per-object KMS calls; consolidating multiple applications under fewer KMS keys while maintaining security boundaries; utilizing the generous free tier of 20,000 requests monthly; optimizing request patterns through batching and caching; avoiding unnecessary key rotations beyond security requirements; properly sizing custom key store usage; and monitoring CloudTrail costs associated with KMS API logging to optimize audit trail expenses.