このコンテンツは公開から3年以上経過しており内容が古い可能性があります
最新情報についてはサービス別資料もしくはサービスのドキュメントをご確認ください

[AWS Black Belt Online Seminar]
AWS Key Management Service

サービスカットシリーズ

Archived

松久 正幸
2018年11月21日

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

自己紹介

名前：松久 正幸 (まつひさ まさゆき)

所属：アマゾン ウェブ サービス ジャパン株式会社
シニア ソリューション アーキテクト

経歴：金融業界を長年担当
国内外のITプロジェクト、コンサルティング実施後AWSへ

好きな AWS サービス：Amazon VPC、AWS KMS

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

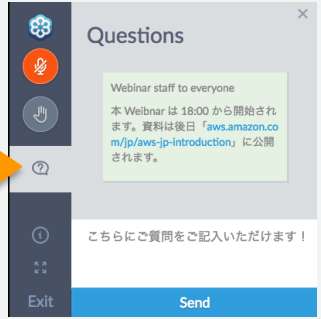
AWS Black Belt Online Seminar とは

「サービス別」「ソリューション別」「業種別」のそれぞれのテーマに分かれて、アマゾン ウェブ サービス ジャパン株式会社が主催するオンラインセミナーシリーズです.

質問を投げることができます！

- 書き込んだ質問は、主催者にしか見えません
- 今後のロードマップに関するご質問は
  お答えできませんのでご了承下さい

① 吹き出しをクリック
② 質問を入力
③ Sendをクリック



Twitter ハッシュタグは以下をご利用ください
#awsblackbelt

本動画および資料は後日サイトへアップいたします.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

内容についての注意点

- 本資料では2018年11月21日時点のサービス内容および価格についてご説明しています．最新の情報はAWS公式ウェブサイト(http://aws.amazon.com)にてご確認ください.

- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます.

- 価格は税抜表記となっています．日本居住者のお客様が東京リージョンを使用する場合、別途消費税をご請求させていただきます.

- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at http://aws.amazon.com/agreement/. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

4

このセミナーで得られること

本セミナーの内容

AWS Key Management Service (AWS KMS)の機能、内部構造、FIPS 140-2検定については、以下の３つの公式文書に記載されています。本セミナーではこれを要約してお伝えします。

AWS Key Management Service Cryptographic Details **[2018年8月更新]**
https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf
AWS Key Management Serviceのベストプラクティス **[日本語]**
https://d0.awsstatic.com/whitepapers/aws-kms-best-practices.pdf
FIPS 140-2 Non-Proprietary Security Policy AWS Key Management Service HSM(Hardware version 2.0, firmware version 1.3.6)**[2017/8/24]**
https://csrc.nist.gov/csrc/media/projects/cryptographic-module-validation-program/documents/security-policies/140sp3009.pdf

本セミナーのゴール

AWS KMSの使い方と鍵管理の仕組みを理解

AWSの鍵管理を正しくご利用いただくこと

社内のセキュリティ及びコンプライアンス部門に鍵管理の仕組みを伝えられること

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Agenda

- AWS KMS概要
  - 暗号化(encrypt)、鍵管理(KMI)、AWS KMS
- AWS KMSの暗号鍵管理機能
- AWS KMSの認可・監査
- AWS KMSの鍵を使用した暗号化/複合化
- AWS KMSの暗号詳細
- AWS KMSのTIPS
- まとめ

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS概要

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

暗号（Cryptographic)

平文(plaintext)を暗号化(encrypt)し暗号文(ciphertext)にする、またそれを複合化(decrypt)すること

暗号の技術は、アルゴリズム(algorithm)と鍵(key)

鍵の使い方によって、対称鍵暗号(symmetric cryptography)と、公開鍵暗号(public-key cryptography)または非対称暗号(asymmetric cryptography)に分類できる

暗号の常識として秘密の暗号アルゴリズムは使用してはいけない、公開された暗号アルゴリズムを使用する

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

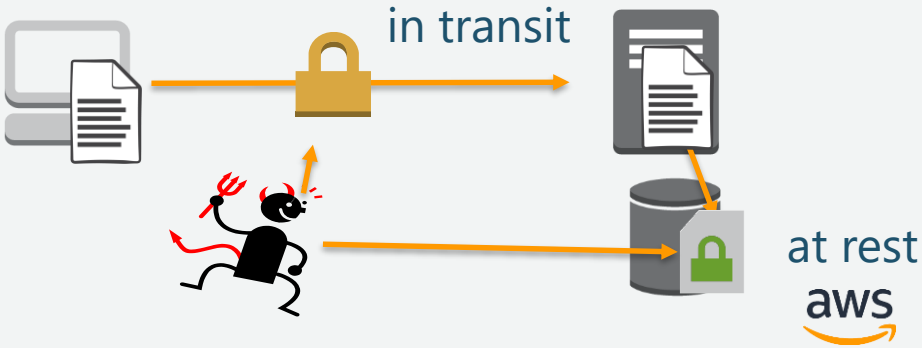
暗号化を利用したデータ保護

通信の暗号化(Data encryption in transit)

- 通信路の盗聴からデータを守る
  - SSL/TLS
  - IPSec
  - 無線LAN暗号化

保管データの暗号化(Data encryption at rest)

- 保管されたデータが意図しない第三者から読み出されるのを防ぐ
  - ファイル暗号化
  - 暗号化ファイルシステム
  - データベース暗号化
  - ブロックデバイス暗号化



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

暗号の鍵管理において考慮すべき問題

鍵はどこに保管されるのか？

- 自身が所有するハードウェア?
- AWSのハードウェア？

鍵はどこで使われるのか？

- 自分で管理するクライアントソフトウェアにて？
- AWS上で稼働するシステムにて？

誰が鍵を使えるのか？

- 許可をもつユーザーやアプリケーション?
- ユーザーが許可を与えたAWS上のアプリケーション？

鍵関連の**確かな**セキュリティを担保する仕組みがあるのか？

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Key Management Infrastructure

KMI(Key Management Infrastructure)とは？

- 暗号鍵の保管、鍵のアクセス制御等、鍵自身のセキュリティを管理するインフラストラクチャ

KMIは2つの機能から成り立つ

- **鍵の保管** - 鍵を安全に保管するストレージ
- **鍵の管理** – 鍵のライフサイクル管理、及び鍵の管理や利用に対する認証・認可・記録するアクセス制御

厳しいコンプライアンス適合要求の場合

- ハードウェアセキュリティモジュール(HSM)が利用される事が多い
- 鍵保管のための占有ストレージ、耐タンパー性、認可の無いユーザーからの保護

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWSにおける暗号化モデルとKMI

お客様が管理
AWSが管理

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Key Management Service (AWS KMS)

暗号鍵の作成、管理、運用のためのサービス

- 可用性、物理的セキュリティ、ハードウェアの管理をAWS が担当するマネージドサービス
- 暗号鍵を保存、暗号鍵を使用するための安全なロケーションを提供
- マスターキーはFIPS 140-2検定済暗号化モジュールによって保護

AWSサービスとの統合 (S3, EBS, Redshift, RDS, Snowball等)

SDKとの連携でお客様の独自アプリケーションデータも暗号化

AWS CloudTrailと連動したログの生成による組み込み型監査

全てのリージョンで利用可能

（※中国リージョン除く）

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSで使用する重要な用語

Customer Master Key(CMK)

- 暗号鍵のヒエラルキーの頂点に位置するKMS上のAES256ビットの鍵
- 暗号化された状態で可用性の高いストレージに保管される
- KMS内部のHSM上でのみ平文で存在し、最大4KBのデータを暗号化/複合化

Customer Data Key(CDK)

- 実際の暗号化対象オブジェクトの暗号化・複合化に使用されるAES256ビットの鍵
- KMSで生成され、CMKで暗号化された状態で保存される
- KMS、利用時のみCMKで複合化し、AWSのサービス内もしくはユーザのアプリケーションで使用する

Envelope Encryption

- マスターキーをデータ暗号化に直接利用するのではなく、マスターキーで暗号化した暗号キーで対象オブジェクトを暗号化/復号する。
- 暗号化したデータと、CMKで暗号化されたデータキーを一緒に保管する、つまりコンテンツに対する封筒（Envelop）
- 大量のデータの暗号化を実現

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSにおける暗号鍵のヒエラルキー

2-Tierの暗号鍵ヒエラルキー

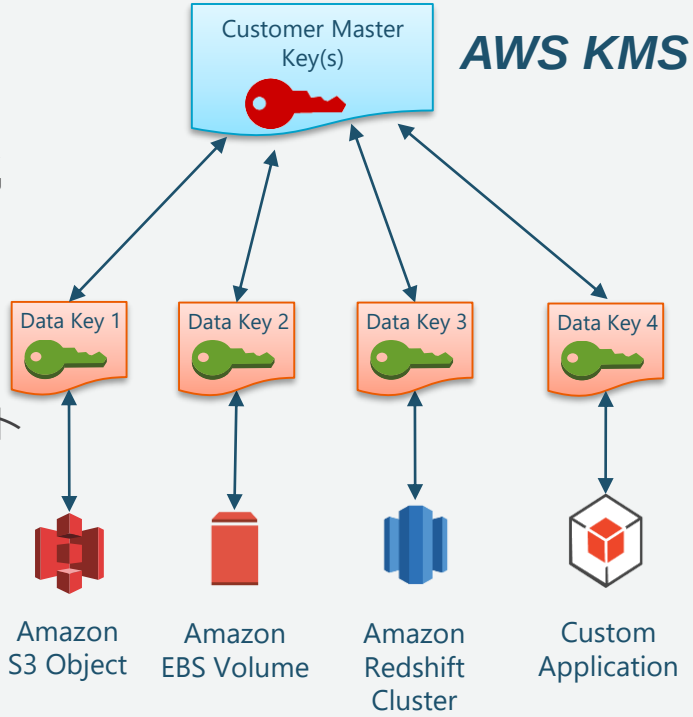
個別のデータキーによるユーザーデータの暗号化

AWS KMS マスターキーによるデータキーの暗号化

Envelope Encryptionを利用

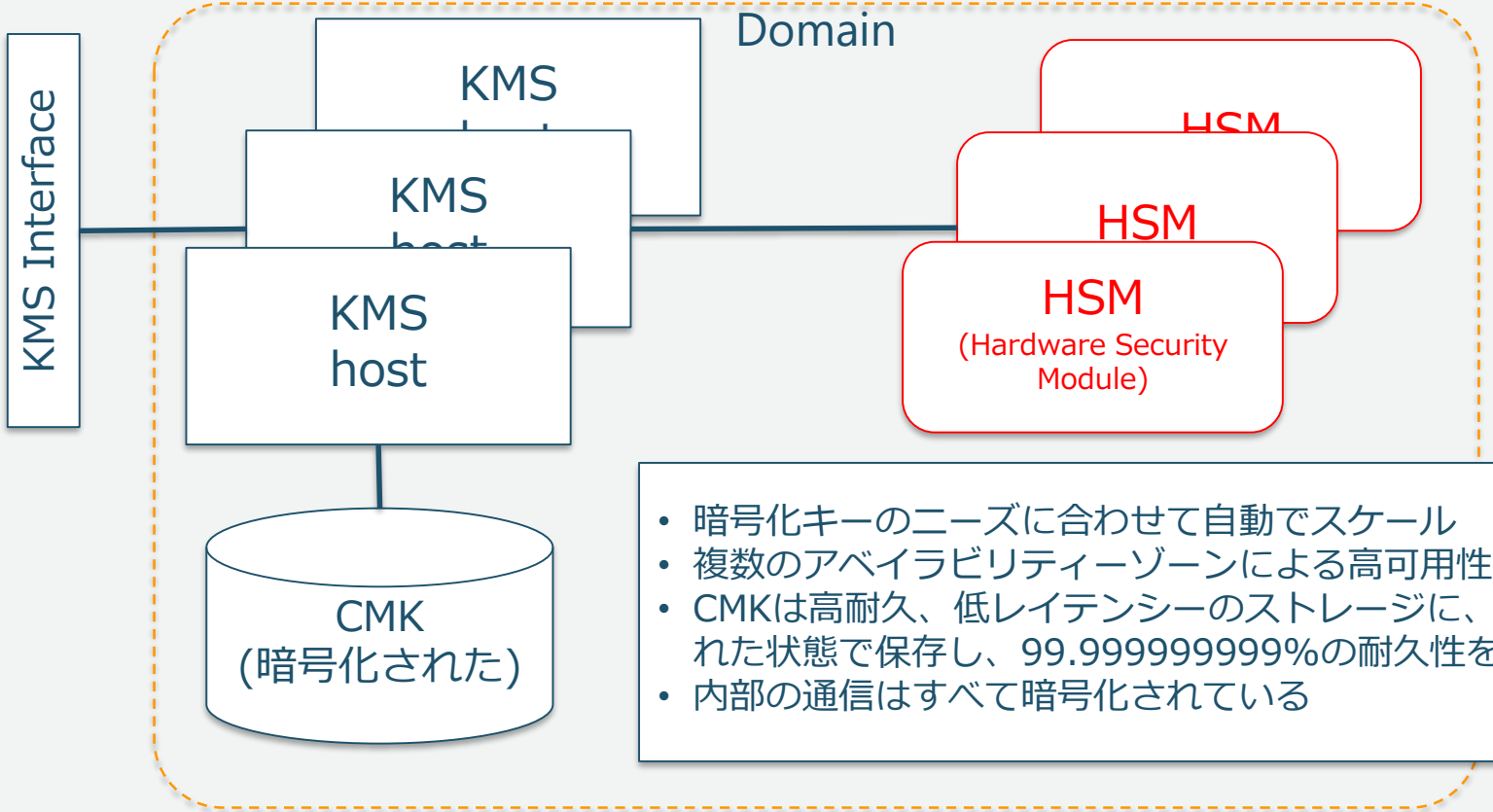
Envelope Encryptionの利点

- データキーの漏洩リスクを限定化
- ラージデータを暗号化する場合のパフォーマンスメリット
- 少数のマスターキーを管理することで管理性を向上
- 鍵利用に関する中央集中アクセスと監査



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

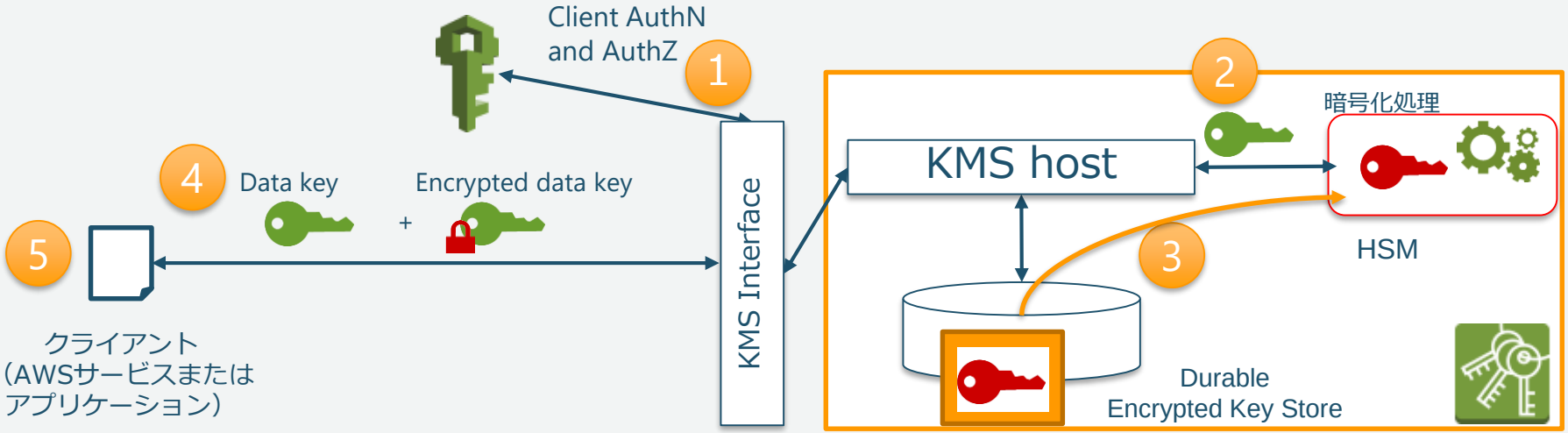
AWS KMSのアーキテクチャー



- 暗号化キーのニーズに合わせて自動でスケール
- 複数のアベイラビリティーゾーンによる高可用性
- CMKは高耐久、低レイテンシーのストレージに、暗号化された状態で保存し、99.999999999%の耐久性を実現
- 内部の通信はすべて暗号化されている

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSの基本動作



1. クライアントは AWSアカウント内のCMKのIDを渡して`kms:GenerateDataKey` を呼び出す。このリクエストは呼び出したユーザーの権限と鍵のアクセス権限の両方に基づき認証する
2. KMSがユニークなCDKを生成
3. 暗号化されたCMKを高耐久ストレージから取り出し、HSMの中でCMKを複合化
4. 複合化されたCMKを使用しCDKを暗号化し、平文のCDKと暗号化されたCDKをクライアントに返す、利用後は平文のCMKは消える
5. クライアントは、平文のCDKを使ってデータを暗号化し、データとともに暗号化されたCDKを保存する

復号：復号の際は、クライアントがデータに含まれる暗号化されたデータキーをKMSに渡す。暗号化されたデータキーは復号のために必要

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS　できること/できないこと

できること

暗号鍵の生成と安全な保管
鍵利用の権限管理
鍵利用の監査
対称鍵暗号
最大4KBのデータ暗号化
AWSサービスとのインテグレーション
鍵のインポート

(現時点で)できないこと

シングルテナント
非対称鍵暗号
4KBを超えるデータの直接的な暗号化
鍵のエクスポート

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSとインテグレーションされているAWSサービス

Alexa for Business*	Amazon Glacier	Amazon WorkMail	AWS Snowball
Amazon Athena	Amazon Kinesis Data Streams	Amazon WorkSpaces	AWS Snowmobile
Amazon Aurora	Amazon Kinesis Firehose	AWS Certificate Manager*	AWS Snowball Edge
Amazon CloudWatch Logs	Amazon Kinesis Video Streams	AWS Cloud9*	AWS Storage Gateway
Amazon Comprehend*	Amazon Lex	AWS CloudTrail	AWS X-Ray
Amazon Connect	Amazon Lightsail*	AWS CodeBuild	
Amazon DynamoDB*	Amazon Simple Email Service (SES)	AWS CodeCommit*	
Amazon DynamoDB Accelerator (DAX)*	Amazon Simple Queue Service (SQS)	AWS CodeDeploy	
Amazon EBS	Amazon Neptune	AWS CodePipeline	
Amazon EFS	Amazon Relational Database Service (RDS)	AWS Database Migration Service	
Amazon Elastic Transcoder	Amazon Redshift	AWS Lambda	
Amazon Elasticsearch Service	Amazon SageMaker	AWS Secrets Manager	
Amazon EMR	Amazon S3	AWS Systems Manager	

*AWS マネージド KMS キーのみをサポートします。

https://aws.amazon.com/jp/kms/details/

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

主要なAWS KMS API

鍵管理用API

- CreateKey, CreateAlias
- DisableKey
- EnableKeyRotation
- PutKeyPolicy
- ListKeys
- DescribeKey

データAPI

- Encrypt
- Decrypt
- GenerateDataKey*

AWS KMSを操作するAPIは35個（2018年11月現在）

http://docs.aws.amazon.com/kms/latest/APIReference/Welcome.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSの暗号鍵管理機能

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSの暗号鍵管理機能

- 鍵の生成
  - 単一のエイリアスおよび説明を付けたCMKの作成
- 鍵のアクセス管理
  - 鍵を管理するIAM ユーザーおよびロールの定義
  - 鍵を使用できるIAM ユーザーおよびロールの定義
  - キーポリシーによるリソースベースの管理
- CMKの無効化/有効化/削除
- 1 年ごとの自動キーローテーションの有効化
- CMKの持ち込み、Bring Your Own Key (BYOK)

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

CMKの作成

事前のポイント

アクセス方法

- AWSマネージメントコンソールのサービス一覧からアクセス（'19 10月以前はIAMの管理画面からアクセス、下記URL参照）

リージョンに注意

- キーを作るリージョンを確認

AWSマネージドキーの存在

- Amazon EBS等のキーなど
- そのリージョンでそのサービスが使われたタイミングで作成

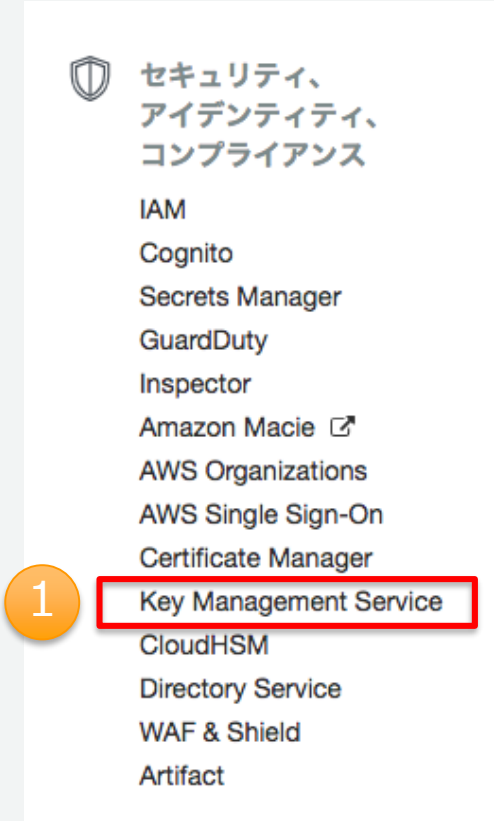
CMK作成手順概要

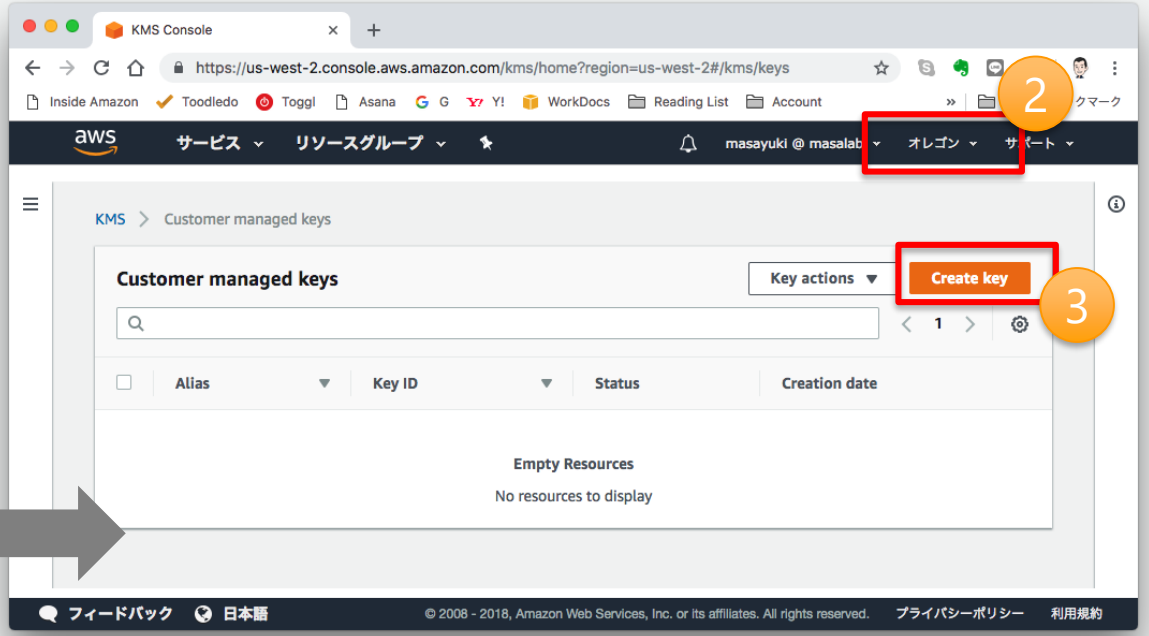
1. エイリアスと説明の作成
2. タグの追加
3. 鍵の管理アクセス許可の定義
4. 鍵の使用アクセス許可の定義
5. キーポリシーのプレビュー

https://aws.amazon.com/jp/about-aws/whats-new/2018/11/aws-key-management-service-has-a-new-console-experience/

© 2020, Amazon Web Services, Inc or its Affiliates. All rights reserved.

AWS管理コンソールからのCMK作成

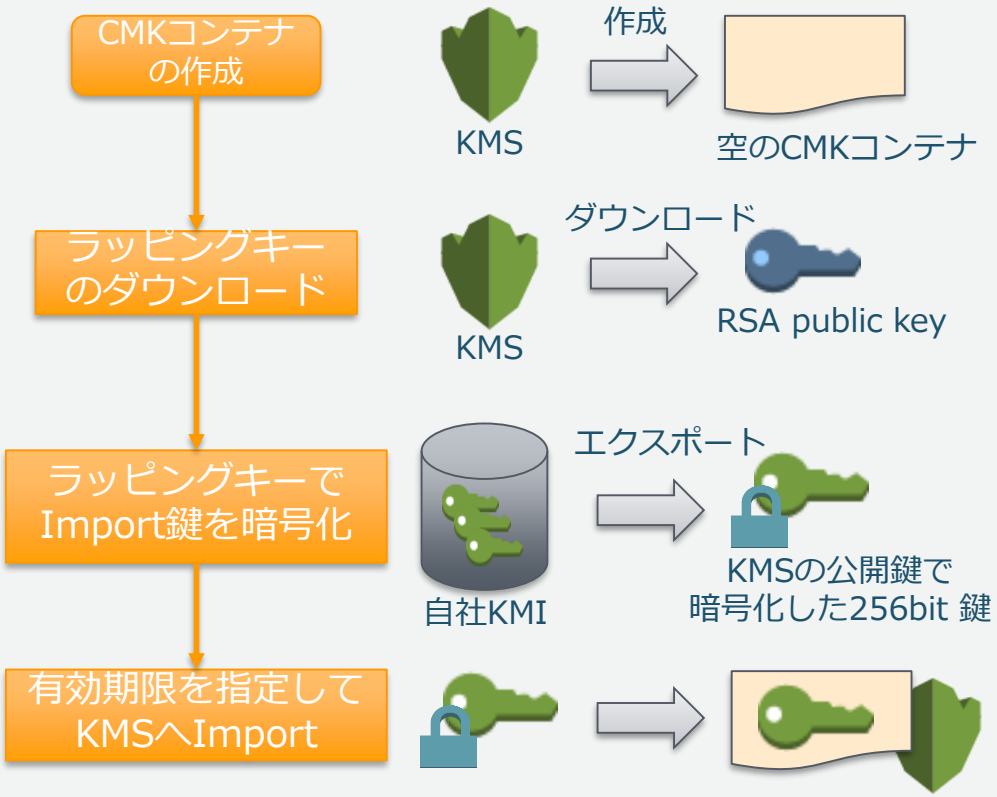




© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

鍵のインポート - Bring Your Own Keys(BYOK)

- お客様所有のKMIで鍵を生成し、その鍵のコピーをAWS KMSにインポート
- インポートされた鍵は、AWS KMSで生成された鍵と同様に、AWS KMSと統合されたAWSのサービスで、または独自のアプリケーションで使用可能
- 鍵の有効期限指定が可能
- 鍵の削除、再インポートが可能
- 運用負荷とインポートの利点を十分に考慮する必要あり



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

BYOK利用時の考慮点

- 鍵のエクスポートは不可。
- インポートする鍵は256ビットの対称鍵のみ。
- 鍵のインポートは2つのRSA PKCS#1スキームのいずれかを使用して保護される。
- インポートした鍵の自動ローテーションは不可。随時手動でローテーションする。
- インポート後はAWS KMS生成のCMKと同等の可用性であるが、耐久性は異なり、リージョン障害などの場合に自動復旧されない。
- 有効期限が過ぎた鍵はKMSによって削除される。AWS KMS生成のCMKは7日～30日の待機時間があるのに対して、即時に削除。インポートの再実施で復旧可能。
- 上記の耐久性と削除（特にオペレーションミスによる削除）の対応として、鍵を必ず保管しておくことが運用の前提。
- 同じ鍵を複数リージョンにインポートすることは可能。
  - 但し、あるリージョンのKMSで暗号化したデータ(データキー)を別リージョンのKMSで復号することはキーIDが違うため出来ない。

詳細)https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/importing-keys.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSの認可・監査

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Key Policy

暗号鍵に対するリソースベースのパーミッション
暗号鍵作成時に定義
AWS管理コンソール、CLI/SDKにより変更可能

- GetKeyPolicy：ポリシーの取得
- PutKeyPolicy：ポリシーの設定

暗号鍵に対するKey Policyの他に、IAM UserやIAM Roleで設定されているPolicyでもアクセス制御が可能

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

暗号鍵のデフォルトポリシー例 (1/2)

```
{
 "Id": "key-consolepolicy-2",
 "Version": "2012-10-17",
 "Statement": [
   {
    "Sid": "Enable IAM User Permissions",
    "Effect": "Allow",
    "Principal": {
     "AWS": [
      "arn:aws:iam::336580663xxx:root"
     ]
    },
    "Action": "kms:*",
    "Resource": "*"
   },
```

AWS rootアカウントを許可するdefault policy

```
   {
    "Sid": "Allow access for Key Administrators",
    "Effect": "Allow",
    "Principal": {
     "AWS": [
      "arn:aws:iam::336580663xxx:user/SuperUser"
     ]
    },
    "Action": [
     "kms:Create*",
     "kms:Describe*",
     "kms:Enable*",
     "kms:List*",
     "kms:Put*",
     "kms:Update*",
     "kms:Revoke*",
     "kms:Disable*",
     "kms:Get*",
     "kms:Delete*"
    ],
    "Resource": "*"
   },
```

暗号鍵管理者向けポリシー

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

暗号鍵のデフォルトポリシー例 (2/2)

```
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
    "Principal": {
      "AWS":
[        "arn:aws:iam::336580663xxx:user/Dev","arn:aws:iam::336580663xxx:role/EC2_Admin","arn:aws:iam::912412960xxx:root"
      ]
    },
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:ReEncrypt*",
      "kms:GenerateDataKey*",
      "kms:DescribeKey"
    ],
    "Resource": "*"
  },
```

暗号鍵利用者向けポリシー

```
{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {

"AWS":["arn:aws:iam::33658066XXXX:user/Dev","arn:aws:iam::33658066XXXX:role/EC2_Admin","arn:aws:iam::91241296XXXX:root"
      ]
    },
    "Action": [
      "kms:CreateGrant",
      "kms:ListGrants",
      "kms:RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
      "Bool": {
        "kms:GrantIsForAWSResource": true
      }
    }
  }
]
}
```

KMS と統合された AWS サービスが、キーを使用するための設定

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Policyの評価

暗号鍵に付与されたリソースベースのパーミッションと、IAM User/Roleに付与したユーザーベースのパーミッションの両方を評価

暗号鍵に付与されるDefault Policyはrootアカウントに対して全ての権限を持つ

KMSのアクセス権の評価は通常のIAMのPolicy評価と同じロジック

- アクセス権限に“Allow”の条件があった場合、アクセス許可
- ただしアクセス権限に１つでも“Deny”の条件があった場合、アクセス拒否(明示的なDeny)
- デフォルトDeny < Allow < 明示的なDeny

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

許可(Grants)

- AWS KMSではキーポリシーと許可(Grants）という2つのリソースベースのアクセスコントロール機能をサポート
- 許可を利用すると、CMKの使用をクライアントによって他のAWSプリンシパルに委任することができる
- より細かいアクセス権の設定が可能
- 利用される5つの要素
  - Key　　: CMKのID
  - GranteePrincipal　　: パーミッションを受ける対象
  - Operations　　: 許可されるオペレーション
  - Constraints　　: Grantsが有効になるConditionを定義
  - RetiringPrincipal　　: Ｇrantsをretireさせるこのとのできる対象

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

許可(Grants)の使用例

```
# Create a grant

key_id = 'arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-
1234567890ab'
grantee_principal = 'arn:aws:iam::111122223333:user/Alice'
operation = 'Encrypt'

response = kms_client.create_grant(
    KeyId=key_id,
    GranteePrincipal=grantee_principal,
    Operations=operation
)
```

https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/programming-grants.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Encryption Context

暗号化機能を利用する際にKMSに渡すことのできるKey/Valueペア

暗号化の際に指定すると、復号化の際にも同一の値が必要とされる

- 追加認証データ（ADD: Additional Authenticated Data)として利用可能

CloudTrailのログに平文で出力されるため、Encryption Contextに機微情報を利用しない

GrantsのConstraintsの中で以下の2つをサポート

- EncryptionContextSubset：EncryptionContext中に含まれていればよい
- EncryptionContextEquals：EncryptionContextが完全に一致した場合

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Constraintsの例

EncryptionContextSubset {“Department”:”Finance”, “classification”:”critcal”}

{“Department”:”Finance”, “classification”:”critical”, “customer”:”12345”} -> OK
{“Department”:” Finance” } -> NG

EncryptionContextEquals {“Department”:”Finance”, “classification”:”critical”}

{“Department”:”Finance”, “classification”:”critical”} -> OK
{“Department”:” Finance”, “classification”:”critical”, “customer”:”12345”} -> NG

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Grantsの利用例 – Amazon EBSのボリュームID指定

Amazon EBS暗号化を行う場合、ボリュームをアタッチした際のログ


```
"eventName":"CreateGrant",
"awsRegion":"us-east-1",
"sourceIPAddress":"AWS Internal",
"userAgent":"AWS Internal",
"requestParameters":{
        "retiringPrincipal":"137640147550",
        "constraints":{
                "encryptionContextSubset":{
                        "aws:ebs:id":"vol-9a98axxx“
                }
        },
        "operations":["Decrypt"],
        "granteePrincipal":"33658066xxxx:aws:ec2-infrastructure:i-5439cxxx",
        "keyId":"arn:aws:kms:us-east-1:33658066xxxx:key/6f712a31-88ba-4975-a4bf-0a87faxxxx“
},
```


© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

CloudTrailによるKMSのログ集約

- CloudTrailによりKMSの管理コンソールでの操作やSDKでの操作で利用したAPIのログを取得
  - CreateAlias
  - CreateGrant
  - CreateKey
  - Decrypt
  - DeleteAlias
  - DescribeKey
  - DisableKey
  - EnableKey
  - Encrypt
  - GenerateDataKey
  - GenerateDataKeyWithoutPlaintext
  - GenerateRandom
  - GetKeyPolicy
  - ListAliases
  - ListGrants
  - ReEncrypt

CloudTrailにより取得できるログ
- 誰が
- いつ
- どのAPIを
- どのAWSリソースから
- どのAWSリソースに対して

http://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/logging-using-cloudtrail.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

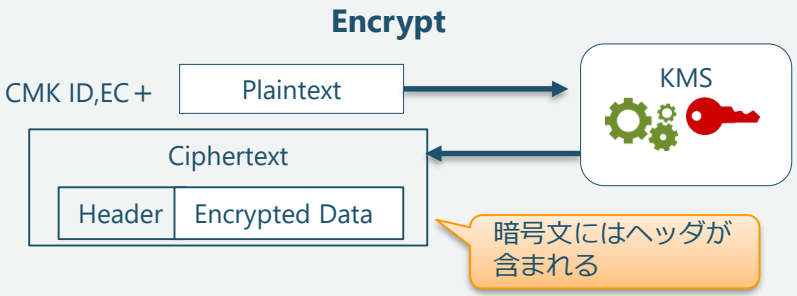
AWS KMSの鍵を使用した暗号化/複合化

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

鍵利用のAPI

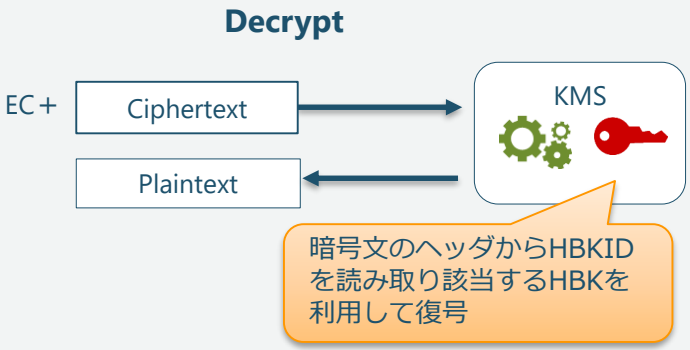
Encrypt API

- データを暗号化するためのAPI
  - 鍵データの暗号化用
- ４KBまでの平文データに対応
- 生成される暗号文にはヘッダが付与される
  - HBK ID、Encryption Contextなどの情報
- 元データおよび生成された暗号文はAWS内には保持されない



Decrypt API

- データを復号するためのAPI
- CMKの指定は不要
  - 暗号文のヘッダから該当するCMKを特定
- 暗号文および平文はAWS内には保持されない



GenerateDataKey

- ユーザーがデータの暗号化に利用するCDKを生成
- 平文の鍵と、Encrytpt APIで暗号化された鍵を返す

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSを使用した暗号処理

データキーを作成

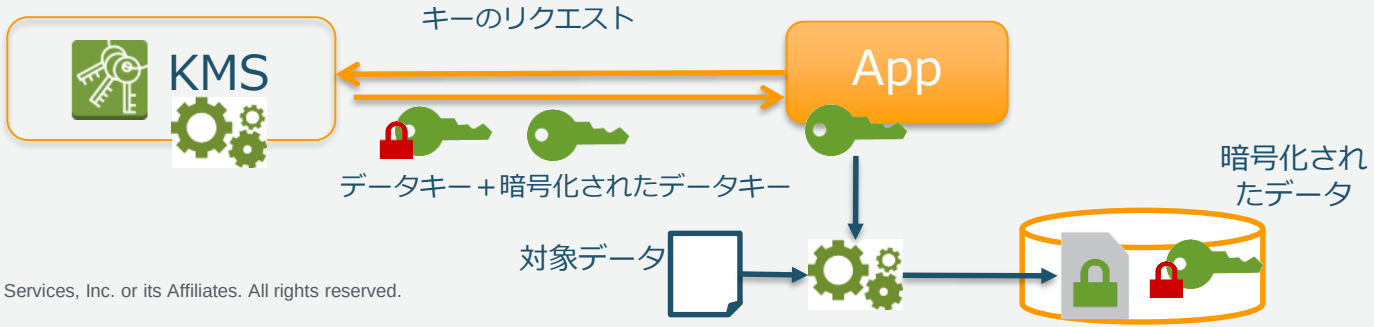
- KMSはデータキーと暗号化されたデータキーの両方を返す

データキーはアプリケーションのメモリ上に配置し、データの暗号化に利用

- データキーの管理はアプリケーション側
  - 平文のデータキーは、決してキーをディスクにおいてはならない
  - 暗号化処理が終了したら即座に削除する

暗号化されたデータキーは暗号化されたデータと共に保存する

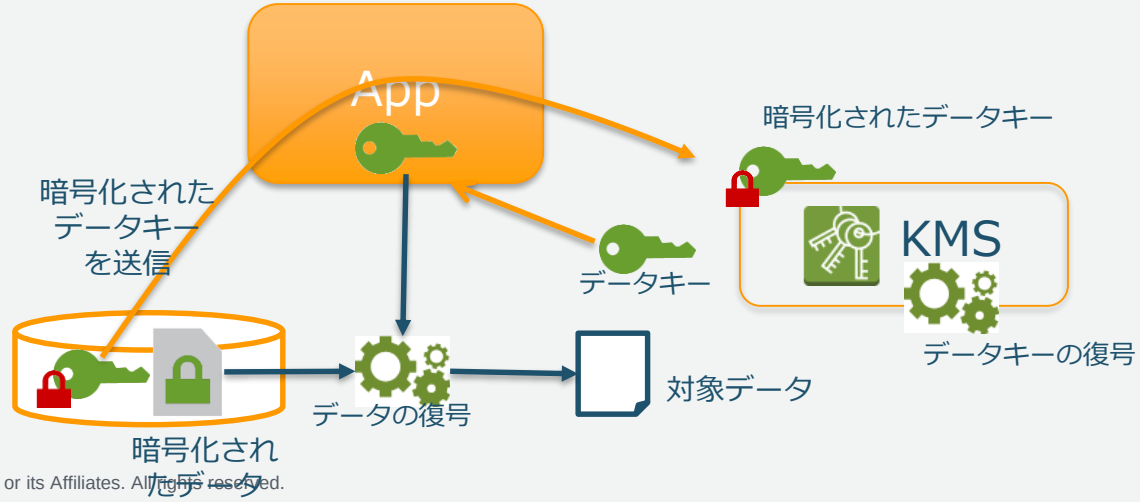
- 管理を容易にするため



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSを使った複合処理

- アプリケーションから暗号化されたデータキーをKMSに送信
- KMSは該当するマスターキーを利用してデータキーを復号し、アプリケーションに返す
- アプリケーションでデータキーを使ってデータを復号
  - 暗号化同様、平文のデータきーの取り扱いに注意



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSを利用したデータ暗号の選択肢

Client-Side Encryption

- ユーザーアプリケーションでのデータ暗号化にKMSを利用
  - 各種AWS SDKを利用
- より上位のSDKやクライアントを利用すれば、Envelope Encryptionを容易にハンドル可能
  - AWS Encryption SDK
  - Amazon S3 Client
    - AWS SDK for Java, .NET, Rubyでサポート
  - Amazon EMR File System (EMRFS)
  - Client-side Encryption for Amazon DynamoDB
    - https://github.com/awslabs/aws-dynamodb-encryption-java

Server-Side Encryption

- AWSの各種サービスとインテグレーションされている
- AWSサービスでデータが受信された後にサービスがKMSを利用してデータを暗号化

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Encryption SDK

AWSが提供するクライアントサイド暗号化のための暗号化ライブラリ

- ”マスターキープロバイダー”を定義して利用するためのAPIと、トップレベルの鍵もしくはデータを暗号化するための複数の鍵へのインターフェースを提供
- データ暗号化キー(DEK)の追跡と保護
- 鍵アクセスのための低レベルの暗号化処理をライブラリ内で実施

トップレベルのマスターキーを指定すれば、SDKが残りの作業を実施

- SDKが低レベルの暗号化処理とトップレベルのマスターキーを結びつける

鍵プロバイダーを抽象化

- 複数の鍵を単一のマスターキープロバイダーにまとめて利用
- 複数の鍵プロバイダーを利用した暗号化(AWS KMSとAWS CloudHSMの同時利用など)

JavaとPythonのための本番稼働に耐えるうる、かつKMSのようなキー生成機能の参照実装が添付されています。

Javaの実装はJava Cryptography Architecture(JCA/JCE)をサポートし、AWS CloudHSMやPKCS#11デバイスもサポートします

https://github.com/awslabs/aws-encryption-sdk-java
https://aws.amazon.com/blogs/security/new-aws-encryption-sdk-for-python-simplifies-multiple-master-key-encryption/

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS SDKを利用したClient-Side Encryption

```
//クライアント設定
//USE ROLES FOR EC2
AWSKMSClient kms = new AWSKMSClient();
kms.setRegion("us-west-2");
```

```
//暗号化
String keyId = "alias/my-application-key";
String plaintext = "Hello World!";
Map<String,String> ec = Collections.singletonMap("dept","finance");
GenerateDataKeyRequest req = new GenerateDataKeyRequest()
                                .withEncryptionContext(ec)
                                .withKeyId(keyId)
                                .withKeySpec("AES-256");
DecryptResult res = kms.generateDataKey(req);
ByteBuffer plaintextKey = res.getPlaintext();
ByteBuffer encryptedKey = res.getCiphertextBlob();

//plaintextKeyを利用してデータを暗号化
・・・
```

KMSのマスターキー

データキーを生成

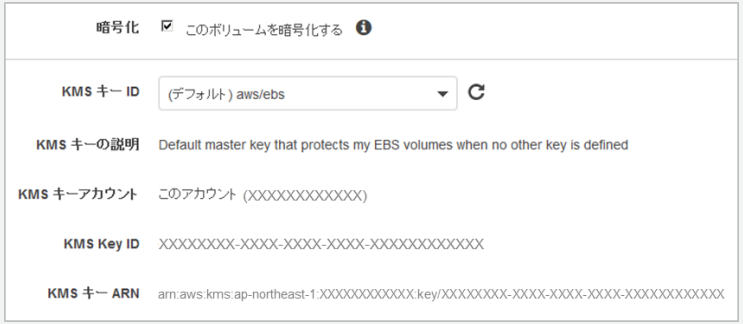
Envelope Encryptionは自分でハンドルする必要がある

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWSサービスでのSSE(Server-Side Encryption) with KMS

Amazon EBSの例

- Amazon EBSのAWSマネジメントコンソール



- 各AWSサービスでCMKを指定するのみ
- Envelope Encryptionのハンドルはサービス側で行われる

- CLI

```
create-volume [--dry-run | --no-dry-run] [--size <value>] [--snapshot-id <value>] --availability-zone <value> [--volume-type <value>] [--iops <value>] [--encrypted | --no-encrypted] [--kms-key-id <value>] [--cli-input-json <value>] [--generate-cli-skeleton]
```

© 2018, Amazon W

AWSサービスでのSSE(Server-Side Encryption) with KMS

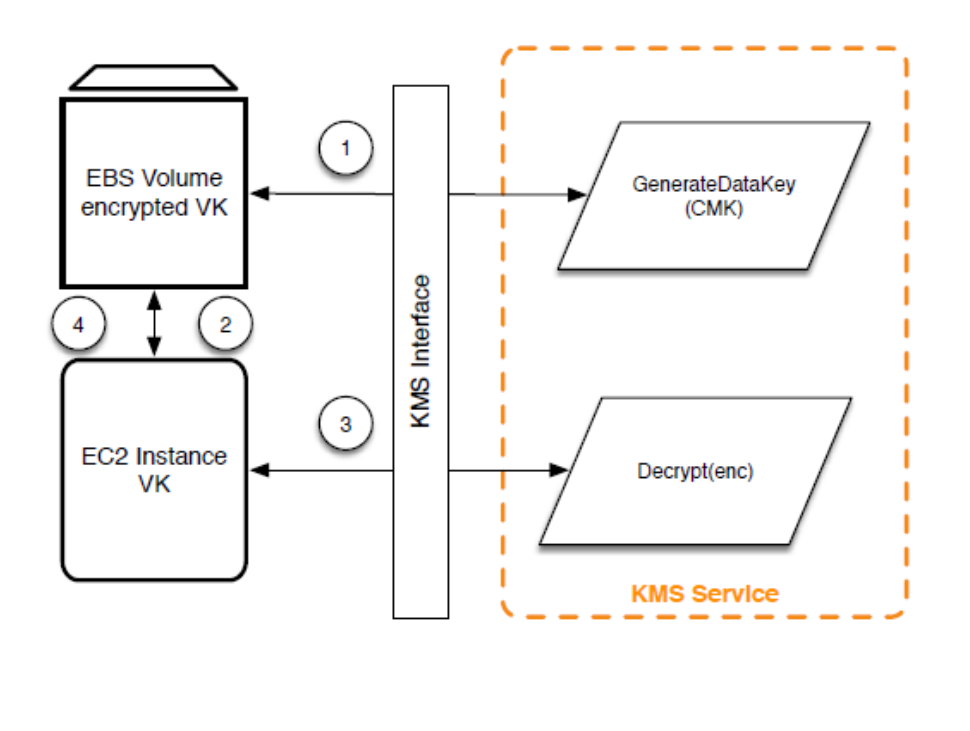
Amazon EBSの例

1.EBSがTLSセッションで暗号化されたボリュームキーをKMSから取得、KMSは指定されたキーへのアクセス権をチェックし、暗号化されたデータキーをEBSに送信(EBSは次回のアタッチに備え、暗号化されたデータキーをボリュームのメタデータに保存)

2.EBSボリュームがマウントされると、EC2がボリュームのメタデータから暗号化されたボリュームキーを取得

3.KMSへの復号リクエストが生成されEC2インスタンスが稼働するホストが復号されたボリュームキーを受け取る

4.ボリュームキーはアタッチされたEBSボリュームに対する全ての入出力データの暗号化/復号に利用される



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSの暗号化詳細

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS Key Management Service HSM

FIPS 140-2 Section Title	Validation Level
Cryptographic Module Specification	3
Cryptographic Module Ports and Interface	2
Roles, Services, and Authentication	3
Finite State Model	2
Physical Security	3
Operational Environment	N/A
Cryptographic Key Management	2
Electromagnetic Interface / Electromagnetic Campatibility	2
Self-Tests	2
Design Assrance	3
Mitigation of Other Attacks	N/A
Overall Level	**2**

詳細 https://csrc.nist.gov/projects/cryptographic-module-validation-program/Certificate/3139

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

FIPS 140-2認定のポイント

暗号アルゴリズム

- AES (FIPS 197, SP800-38A)、鍵長128ビット、256ビット

暗号鍵の生成

- DRBG (SP 800-90A) – ランダムビットジェネレーター
- NDRNG – DRBGに対するSeed、384ビットのエントロピー
- Recommendation for Block Cipher Modes of Operations : Galois/Counter Mode(GCM) and GMAC (SP 800-38D)

https://aws.amazon.com/jp/about-aws/whats-new/2018/11/aws-key-management-service-has-a-new-console-experience/

© 2022, Amazon Web Services, Inc or its Affiliates. All rights reserved.

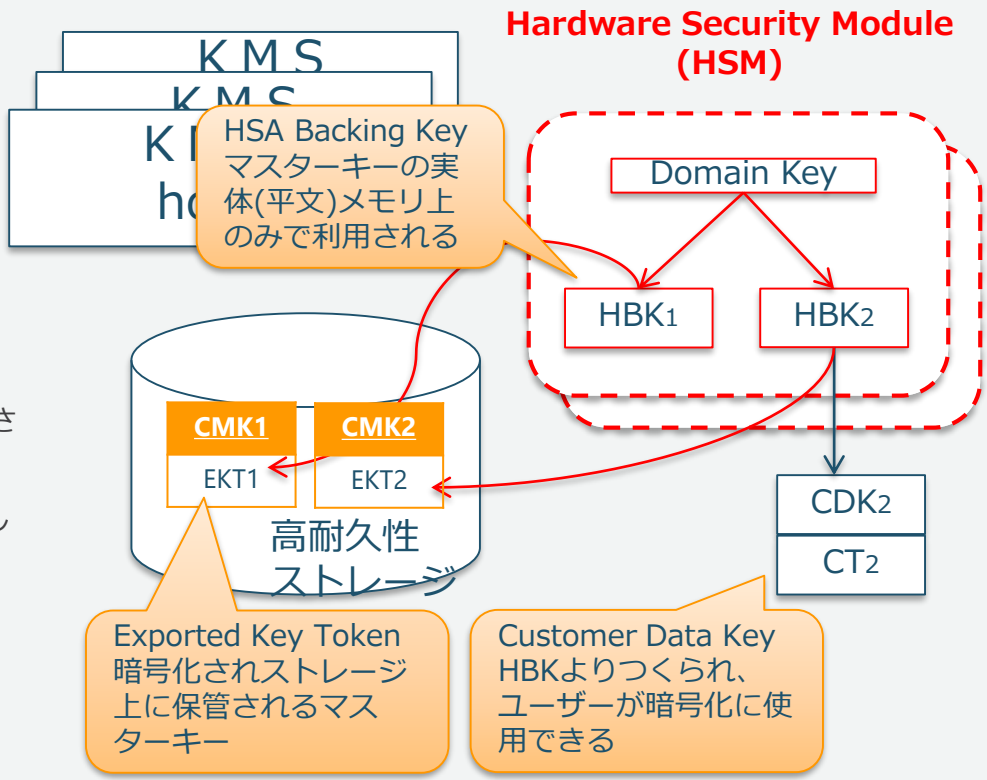
AWS KMSの鍵管理

KMS内部では以下の鍵データが存在する

- Domain Key(DK)
  - 管理用の鍵
  - リージョン内の全てのHSMが**メモリ上**に保持
  - 日次でローテーション
- HSM Backing Key(HBK)
  - CMKの実体となる平文の鍵データ
  - CMK生成時にHSMで生成、もしくはBYOK
  - **HSMの揮発性メモリ上でのみ平文で存在**
  - 平文のままHSMから取り出し不可
- Exported Key Token(EKT)
  - DKで暗号化されたHBK、つまりCMKは暗号化されている
  - KMS内の高耐久性ストレージに保管される
  - CMK利用時にHSM上で復号化しHBKを取り出し

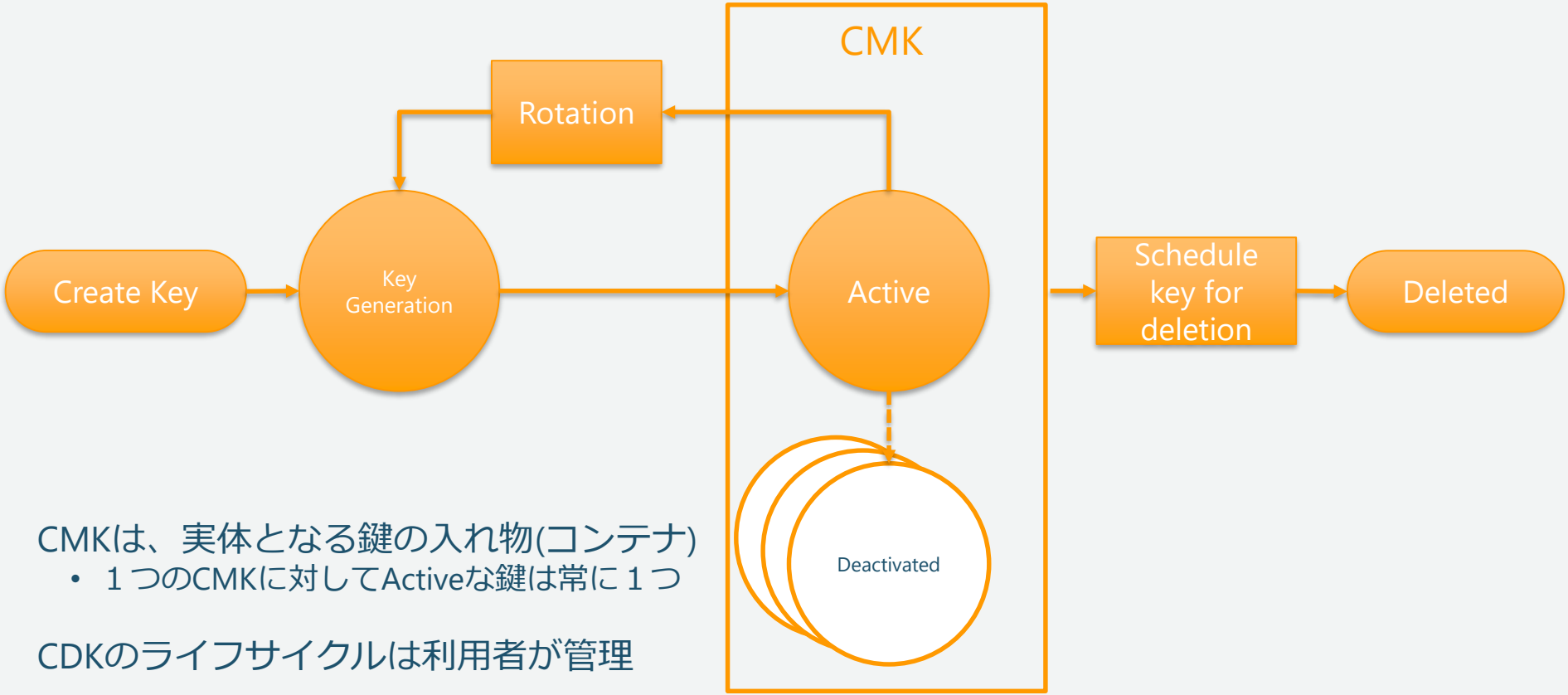
CMKは論理キー(鍵のコンテナ)

- HBK、EKTが関連付けられている
- ARNつまりキーIDは実際には実体であるHBKを指している



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

CMK(カスタマーマスターキー)のライフサイクル



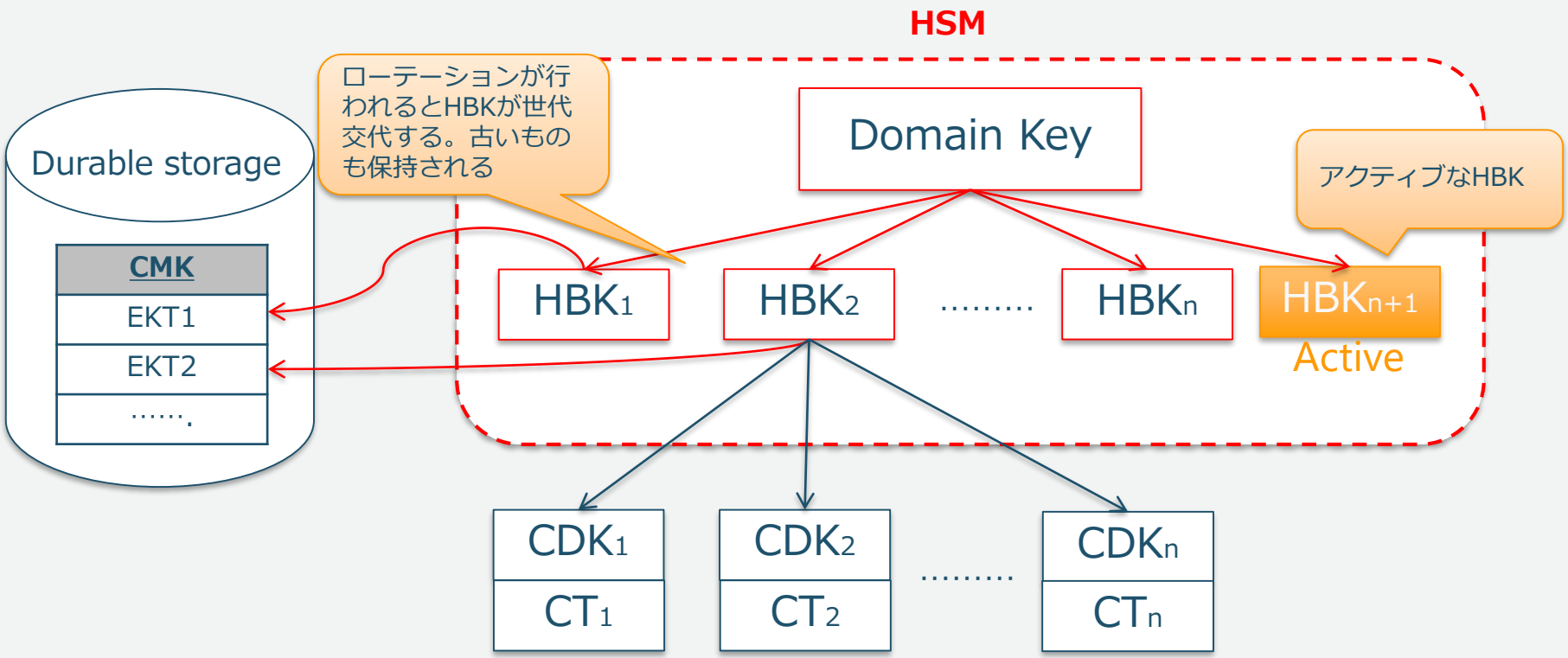
CMKは、実体となる鍵の入れ物(コンテナ)

- １つのCMKに対してActiveな鍵は常に１つ

CDKのライフサイクルは利用者が管理

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

ローテーションの仕組み



© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSのTIPS

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSのベストプラクティス(1/2)

- 認証と認可
  - IAMポリシーとキーポリシーのデザイン
  - 最小権限と権限の分離(Least Privilege / Separation of Duties)
    - クロスアカウントのアクセス、ルーツアカウントの考慮
    - PutKeyPolicyに対する権限
    - Encryption Context
  - MFAの利用の検討

- 発見的統制
  - CloudTrailの利用

- パフォーマンスとサービス制限
  - 1秒あたりのAPIオペレーションのリクエスト数に注意

https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/limits.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMSのベストプラクティス(2/2)

- インフラストラクチャのセキュリティ
  - カスタマーマネージド型鍵の利用
  - 鍵をどの単位で分割するかの検討（AWSアカウントの体系、データクラス、インシデント対応から考える）

	AWS マネージド型 CMK	カスタマーマネージド型 CMK
作成	顧客のために AWS が生成する	顧客が生成する
ローテーション	3 年に 1 回自動的	1 年に 1 回自動的にオプトインで、またはオンデマンドで手動で
削除	削除できません	削除できます
使用範囲	特定の AWS サービスに制限	KMS/IAM ポリシーで管理
キーアクセスポリシー	AWS マネージ	カスタマーマネージ
ユーザーアクセス管理	IAM ポリシー	IAM ポリシー

- インシデント対応の明確化と自動化
  - AWS KMSセキュリティの自動化
  - CMKの無効化と削除、特に削除は慎重に

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS利用TIPS

直接暗号化/復号できるデータは**4KB**まで

- Envelope Encryption方式での利用を推奨

APIリクエストのスロットリングに注意

- 暗号化EBSボリュームを複数もつインスタンスの同時起動や、大量オブジェクトのS3へのUpload/Download等に注意
- 必要に応じて上限緩和申請
  - https://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/limits.html

対称鍵暗号方式のみサポート

- 公開鍵暗号方式では利用できない

リージョン間での鍵の共有は出来ない

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS利用TIPS：EBS

サポートされているインスタンスタイプに注意

ボリュームの生成時にCMKを指定して暗号化

- 暗号化されていないボリュームの直接の暗号化、暗号化ボリュームの直接の非暗号化は不可
  - 暗号化されていないボリュームのスナップショットをコピーする時に暗号化することは可能
  - 暗号化されたスナップショットをコピーする時に異なるCMKを指定して再暗号化することは可能

暗号化スナップショットを他のAWSアカウントへ共有することが可能

- CMKに対するアクセス権も付与する必要がある
- デフォルトキーで暗号化されたスナップショットは共有不可
- 共有されたスナップショットからボリュームを作成する場合は一旦コピーが必要

KMSのCMKで暗号化されたスナップショットはリージョン間コピー不可

1インスタンスにアタッチできる同じCMKで暗号化したボリュームは**30個**まで

http://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/EBSEncryption.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS利用TIPS : S3

KMSで暗号化されたオブジェクトに対する操作はVersion4署名が必要

オブジェクトの暗号化に利用されたデータキーもオブジェクトと共に暗号化されて保管される

レスポンスに含まれるETagはオブジェクトのMD5ではない

バケットポリシーでSSE-KMSの強制化が可能

- s3:PutObjectに対して、“s3:x-amz-server-side-encryption”:“aws:kms”の条件を利用

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS 利用TIPS : RDS(1/2)

ストレージ暗号化で利用

新規インスタンスの作成時に暗号化指定

暗号化可能なインスタンスタイプに注意

暗号化されたインスタンスの暗号化解除は不可

暗号化されていないスナップショットをコピーして暗号化することは可能(Auroraでは不要)

リードレプリカも同じキーを利用して暗号化する必要がある

暗号化されていないバックアップ/スナップショットを暗号化インスタンスにリストアすることは出来ない

Auroraでは暗号化されていないスナップショットから暗号化クラスターに復元可能

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

AWS KMS利用TIPS：RDS(2/2)

暗号化されたMySQLスナップショットをAuroraに復元することはできない

KMSで暗号化されたスナップショットのリージョン間コピーは不可

リージョン間レプリケーションも不可

暗号化スナップショットを他のAWSアカウントへ共有することが可能

- CMKに対するアクセス権限を付与する必要がある
- 共有されたスナップショットからインスタンスを作成する場合は一旦コピーが必要

  https://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/USER_ShareSnapshot.html

不用意なキーのDisableは避ける

- キーをDisableするとTerminalステートになり、リカバリー不能となる
- バックアップからのリストアが必要になる

  http://docs.aws.amazon.com/ja_jp/AmazonRDS/latest/UserGuide/Overview.Encryption.html

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

KMSのセキュリティ(1/3)

鍵の管理はお客様が実施

マスターキーには誰もアクセスできない

- 幅広い堅牢化技術を使用してマスターキーを保護するように設計
  - プレーンテキストのマスターキーをディスクに保存しない
  - メモリ上に存続させ続けない
  - デバイスに接続できるシステムを制限するなど
- SOCレポートを参照

KMSサービス内でソフトウェアをアップデートするためのアクセスは、複数段階の承認プロセスによって管理

- Amazonの独立したグループによって監査およびレビューを実施

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

KMSのセキュリティ(2/3)

キーは作成されたリージョンにのみ保存

- 他のリージョンに移動できない

キーのローテーションが可能

- １年毎に自動ローテーション（インポートされたキーは手動のみ）

AWS CloudTrailによる管理操作、暗号化操作の記録

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

KMSのセキュリティ(3/3)第三者認証

AWS Service Organization Controls レポート

- SOC1、SOC2レポートはAWS Artifactでダウンロードしてください。

PCI DSS レベル 1

- AWS の PCI DSS 準拠サービスの詳細については、PCI DSS のよくある質問を参照してください。

ISO 27017

- AWS の ISO 27017 準拠サービスの詳細については、ISO-27017 のよくある質問を参照してください。

ISO 27018

- AWS の ISO 27018 準拠サービスの詳細については、ISO-27018 のよくある質問を参照してください。

ISO 9001

- AWS の ISO 9001 準拠サービスの詳細については、ISO-9001 のよくある質問を参照してください。

FIPS 140-2

- 詳細については、FIPS 140-2 Modules In Process List で確認できます。

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

KMSの料金 http://aws.amazon.com/jp/kms/pricing/

全リージョンで共通

1USD / key version / 月

- KMSで生成、インポートにかかわらない
- ローテーションを有効にした場合、料金は更新した各バージョンにつき月間**1USD**
- 課金対象外のkey
  - AWS管理のサービスデフォルトキー
  - 削除を予定されたCMK
    - 待機期間中に削除をキャンセルした場合、削除を予定されなかったものとして課金
  - GenerateDataKey / GenerateDataKeyWithoutPlaintext で生成されたCDK

0.03USD per 10,000 APIリクエスト (GovCloudでは、0.04USD)

- 20,000 APIリクエスト / 月 の無償利用枠 (全リージョン合計)

※価格は2018年11月現在のものです。CloudTrailを有効にしてAPIアクセスログを取得する場合は別途S3,SNSの料金がかかります。

© 2018, Am

まとめ

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

まとめ

- AWS KMSは、データを保護するための暗号化キーの一元管理を可能にする低コストなマネージドサービスです
- AWSサービスとの統合により容易にデータの暗号化を行うことができますし、アプリケーション上のデータの暗号化にも利用することができます
- 暗号鍵のセキュアな管理を可能にし、CloudTrailとの統合による組み込み型の監査対応機能も利用できます

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

参考情報

AWS Key Management Service Developer Guide

- http://docs.aws.amazon.com/ja_jp/kms/latest/developerguide/overview.html

AWS Key Management Service API Reference

- http://docs.aws.amazon.com/ja_jp/kms/latest/APIReference/Welcome.html

AWS Key Management Service FAQ

- http://aws.amazon.com/jp/kms/faqs/

AWS Key Management Service Pricing

- http://aws.amazon.com/jp/kms/pricing/

AWS Key Management Service Cryptographic Details **[2018年8月更新]**

- https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf

AWS Key Management Serviceのベストプラクティス **[日本語]**

- https://d0.awsstatic.com/whitepapers/aws-kms-best-practices.pdf

FIPS 140-2 Non-Proprietary Security Policy AWS Key Management Service HSM(Hardware version 2.0, firmware version 1.3.6)**[2017/8/24]**

- https://csrc.nist.gov/csrc/media/projects/cryptographic-module-validation-program/documents/security-policies/140sp3009.pdf

© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Q&A

お答えできなかったご質問については

AWS Japan Blog
「https://aws.amazon.com/jp/blogs/news/」にて
資料公開と併せて、後日掲載します.

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

70

AWS Well-Architected 個別技術相談会

毎週”W-A個別技術相談会”を実施中

- AWSのソリューションアーキテクト(SA)に対策などを相談することも可能

- 申込みはイベント告知サイトから

(https://aws.amazon.com/jp/about-aws/events/)

AWS イベント で[検索]

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

ご視聴ありがとうございました

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive

© 2018, Amazon Web Services, Inc. or its Affiliates. All rights reserved.