

Chiffrez vos données avec AWS KMS et AWS CloudHSM

Julien Simon

Principal Technical Evangelist, AWS

julsimon@amazon.fr

@julsimon



Programme

Chiffrement et gestion des clés

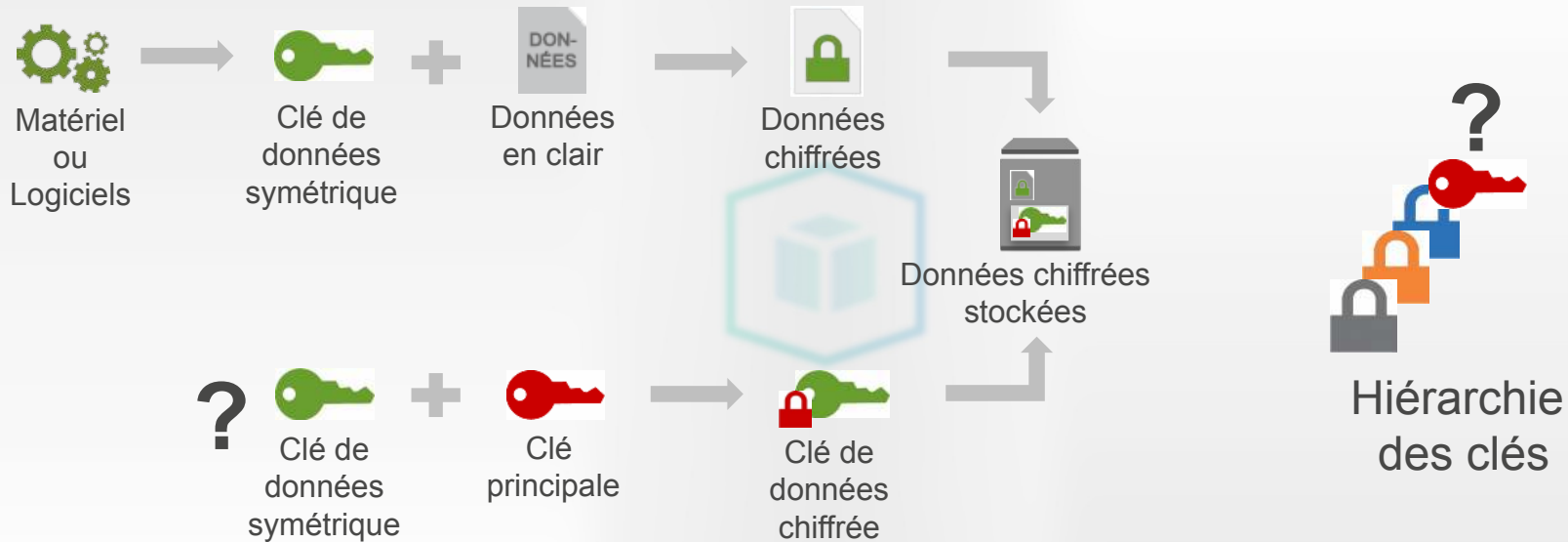
Comment AWS protège vos données avec le chiffrement

Comment AWS KMS simplifie le chiffrement

Alternatives à AWS KMS : AWS CloudHSM et solutions partenaires

Questions et réponses

La problématique du chiffrement



Problèmes à résoudre

Où sont stockées les clés ?

- Sur du matériel qui vous appartient ?
- Sur du matériel qui appartient au fournisseur de cloud ?

Où les clés sont-elles utilisées ?

- Sur un logiciel client que vous contrôlez ?
- Sur un logiciel de serveur que le fournisseur de cloud contrôle ?

Qui peut utiliser les clés ?

- Les utilisateurs et les applications qui en ont l'autorisation ?
- Les applications du fournisseur de cloud auxquelles vous avez donné l'autorisation ?

Quelles sont les assurances d'une utilisation appropriée des clés ?

Méthodes de chiffrement

Chiffrement côté client

Client-side encryption

- Clés gérées par le client

Chiffrement côté serveur

Server-side encryption, i.e. SSE

- S3 : Clés fournies par le client (SSE-C)

Chiffrement côté client

Clés gérées par le client

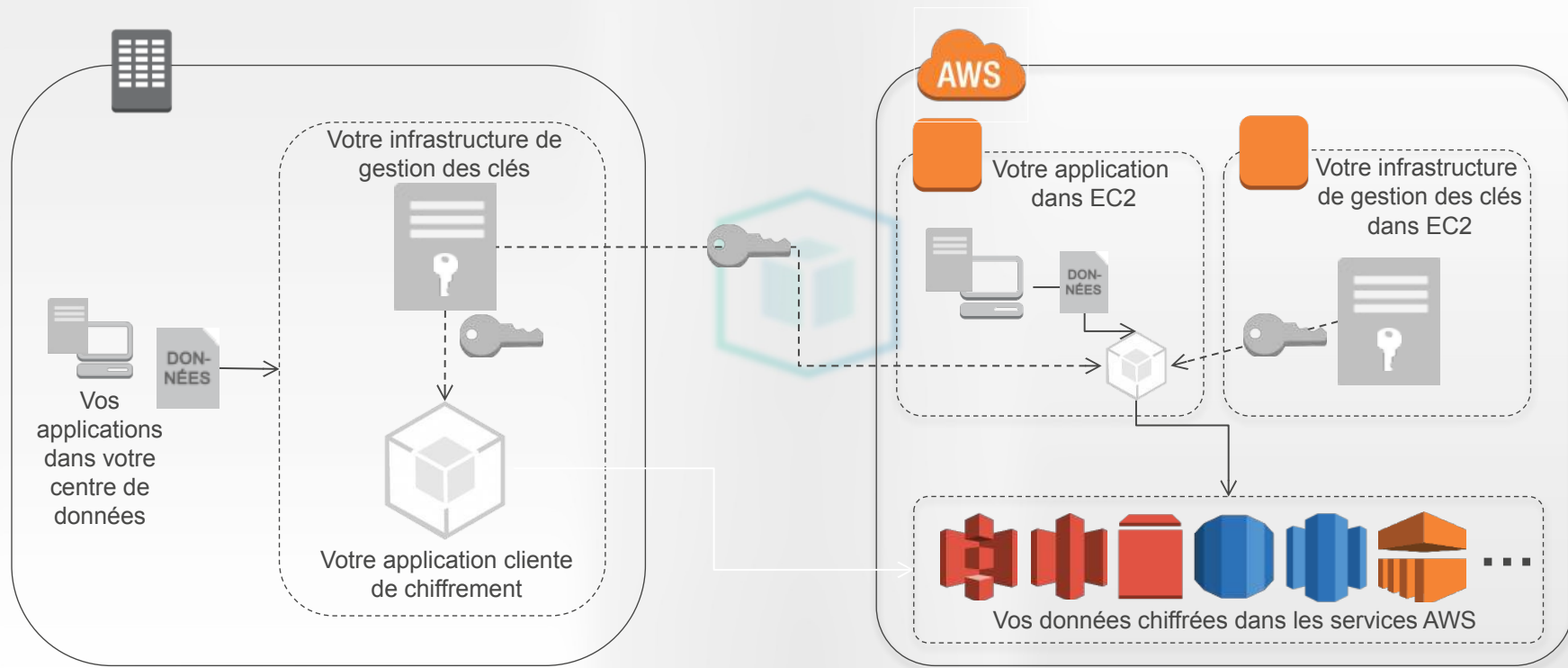


Clé principale gérée par le client

- Le code client obtient auprès de son infrastructure une clé de chiffrement (*Client-side Master Key*, CMK).
- Le code utilise la CMK et ses propres primitives crypto pour chiffrer les données.
- Les données chiffrées sont envoyées à AWS.



Clé principale gérée par le client



Chiffrement côté serveur

Clés fournies par le client (SSE-C)

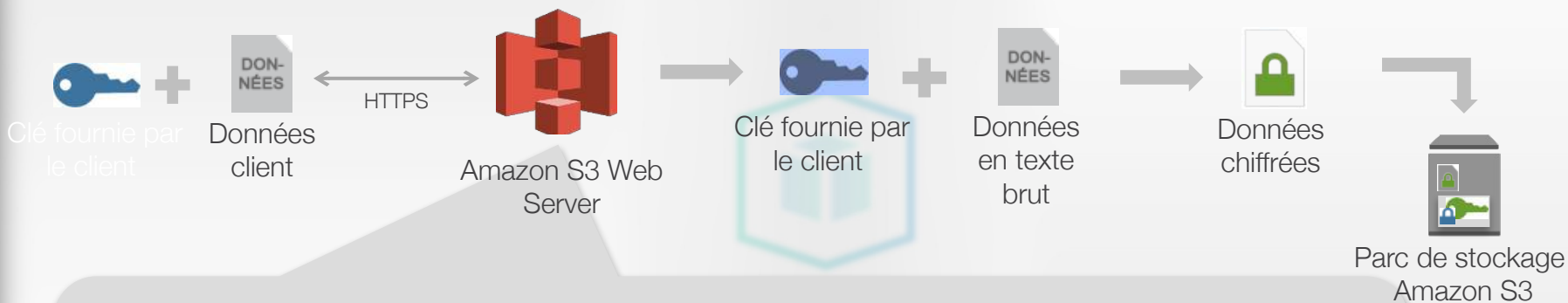
Clés fournies par le client (SSE-C)

- Les données au repos dans S3 sont **chiffrées par le service lui-même**. Les métadonnées ne sont pas chiffrées.
- Chaque objet est chiffré avec **AES-256** et une **clé fournie par le client** :
 - programmatiquement avec un SDK AWS
 - par API en ajoutant les en-têtes HTTP
 - *x-amz-server-side-encryption-customer-algorithm* (AES-256)
 - *x-amz-server-side-encryption-customer-key*
 - *x-amz-server-side-encryption-customer-key-MD5*



Chiffrement côté serveur dans AWS

Server-Side Encryption with Customer-Provided Encryption Keys (SSE-C)



La clé est utilisée au niveau du serveur Web S3, puis supprimée

Le client doit fournir la même clé lors du téléchargement afin de permettre à S3 de déchiffrer les données

Le client peut donc gérer ses clés, mais...

Où sont stockées les clés ?

- Sur du matériel qui vous appartient ?
- Sur du matériel qui appartient au fournisseur de cloud ?

Où les clés sont-elles utilisées ?

- Sur un logiciel client que vous contrôlez ?
- Sur un logiciel de serveur que le fournisseur de cloud contrôle ?

Qui peut utiliser les clés ?

- Les utilisateurs et les applications qui en ont l'autorisation ?
- Les applications du fournisseur de cloud auxquelles vous avez donné l'autorisation ?

Quelles sont les assurances d'une utilisation appropriée des clés ?

L'objectif de KMS est de résoudre
ces problèmes à votre place

AWS Key Management Service

- KMS simplifie la **création**, le **contrôle**, la **rotation** et l'**utilisation** des clés de chiffrement dans vos applications
- KMS est intégré :
 - **Côté serveur** : S3, EBS, Snowball, RDS, Redshift, DMS, CodeCommit, EMR, Firehose, SES, Elastic Transcoder, WorkMail, WorkSpaces
 - **Côté client** : SDKs, Clients de chiffrement S3 & DynamoDB
 - **AWS CloudTrail** fournit des informations d'audit sur l'utilisation des clés.
- KMS est disponible dans **toutes les régions** à l'exception de la Chine.
- Vous pouvez également **importer** vos propres clés
<https://aws.amazon.com/fr/blogs/aws/new-bring-your-own-keys-with-aws-key-management-service/>

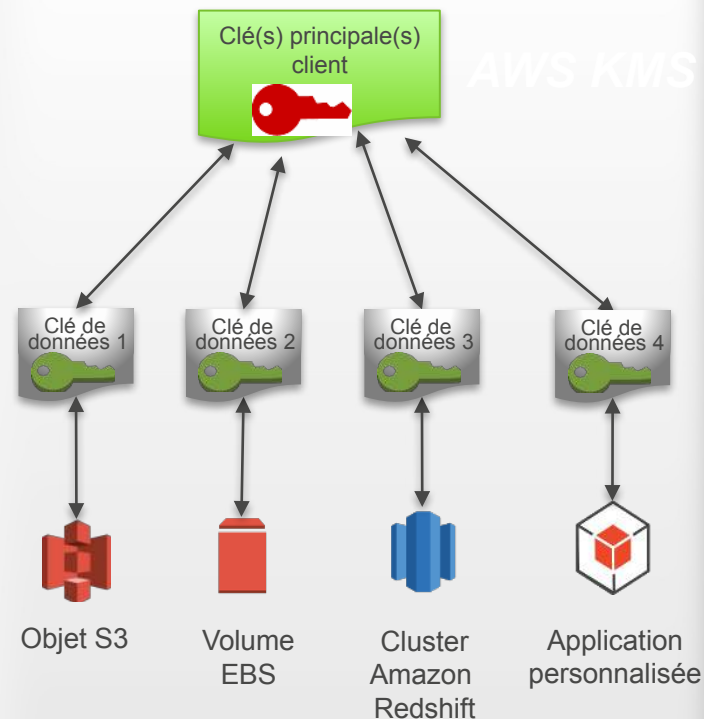
Comment les services AWS s'intègrent-ils à KMS ?

Hiérarchie de clés à deux niveaux

- Les **clés principales**, stockées dans KMS, chiffrent les clés de données.
- Les **clés de données** chiffrent les données client.

Avantages :

- Minimiser **l'impact** de la compromission d'une clé de données
- Réduire **le nombre de clés** à gérer
- Faciliter **l'audit** de l'activité des clés en centralisant leur gestion



Pourquoi confier vos clés à AWS ?

- KMS est conçu de manière à ce que **personne** ne puisse accéder à vos clés principales.
- KMS ne stocke **jamais** les clés principales en clair.
- KMS n'exporte **jamais** les clés principales.
- KMS ne stocke **jamais** les clés de données.
- KMS **sépare les responsabilités** entre les systèmes qui manipulent les clés principales et ceux qui manipulent les clés de données.
- KMS est utilisé et audité par les organisations **les plus exigeantes** en matière de sécurité.
- KMS est **certifié** : SOC 1/2/3, PCI DSS Level 1, ISO 27017, ISO27018, ISO 9001, FIPS 140-2 en cours

Utilisation des clés principales

L'utilisation des clés principales est soumise aux **permissions** accordées par **IAM**.

Par exemple :

- Clé accessible uniquement par **un groupe d'utilisateurs**
- Clé utilisée pour le chiffrement et le déchiffrement par un groupe d'instances **EC2** possédant le **rôle adéquat**
- Clé utilisée par **l'application A pour chiffrer** les données et par **l'application B pour déchiffrer** les données

Console KMS

Rechercher sur IAM

Tableau de bord

Groupes

Utilisateurs

Rôles

Stratégies

Fournisseurs d'identité

Paramètres du compte

Rapport sur les informations d'identification

Clés de chiffrement


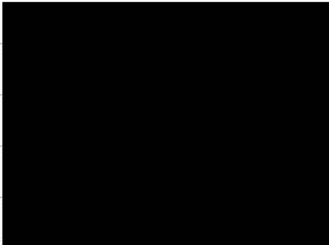




Créer une clé

Actions de clé

5 résultats affichés

Filtre : USA Est (Virginie du Nord)

Q Filtre

<input type="checkbox"/>	Alias	ID de clé	Statut	Date de création
	aws/s3		Activé	2016-04-12 11:22 UTC+0100
	aws/codecommit		Activé	2016-02-08 22:35 UTC+0100
	aws/acm		Activé	2016-10-05 13:15 UTC+0100
	aws/ebs		Activé	2016-04-20 10:52 UTC+0100
	aws/lambda		Activé	2016-12-04 14:48 UTC+0100

L'identifiant des clés n'est pas secret, mais par principe, évitez de le révéler...

API KMS

KMS peut être utilisé en **ligne de commande** ou via l'un des **SDKs** AWS.

Exemples d'API de gestion

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

Exemple d'API de données

- Encrypt
- Decrypt
- ReEncrypt
- GenerateDataKey



32 API au total

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

Génération et utilisation des clés de données



1. L'application passe la **référence de la clé principale** à utiliser et demande **une clé de données**.
2. La demande client est **vérifiée par IAM**.
3. Une nouvelle clé de données est **générée aléatoirement** par KMS.
4. La clé de données (en clair et chiffrée) est **envoyée au client**.
5. Le client utilise la clé de données pour **chiffrer** les données, puis il la **supprime**.
6. La clé de données chiffrée est **stockée** par le client : elle sera envoyée à KMS pour le **déchiffrement**.

Auditer l'utilisation des clés principales avec CloudTrail

"EventName": "DecryptResult",

Cette API KMS a été appelée...

"EventTime": "2014-08-18T18:13:07Z",

... à ce moment-là

"RequestParameters":

"{\"keyId\": \"2b42x313-1911-4e2a-8321-6b67324025eb\"}\", ... en référence à cette clé

"EncryptionContext": "volumeid-23657",

... pour protéger cette ressource AWS

"SourceIPAddress": "46.23.143.114",

... à partir de cette adresse IP

"UserIdentity":

"{\"arn\": \"arn:aws:iam::957787256530:user/User123\"} ... par cet utilisateur AWS

Tarification de KMS

- \$1 par clé par mois
- \$0,03 par 10 000 appels d'API
- Niveau d'usage gratuit : 20 000 appels par mois

Méthodes de chiffrement

Chiffrement côté client

Client-side encryption

- Clés gérées par le client
- Clés gérées par AWS KMS

Chiffrement côté serveur

Server-side encryption, i.e. SSE

- S3 : clés fournies par le client (SSE-C)
- Clés gérées par AWS KMS (SSE-KMS)

Chiffrement côté client

Clés gérées par KMS

Clés gérées par KMS

- Le code client passe l'identifiant d'une clé KMS au SDK AWS (par exemple à *AmazonS3EncryptionClient* pour S3).
- Le SDK reçoit une clé de données, avec laquelle il chiffre les données, puis les envoie à AWS.
- La clé chiffrée est stockée avec les données. Elle sera utilisée pour le déchiffrement.



Chiffrement côté serveur

Clés gérées par Amazon KMS (SSE-KMS)

Chiffrement côté serveur dans AWS

Amazon EBS

Consol

e

Encryption ⓘ ☒ Encrypt this volume

Master Key ⓘ

Key Details

Description	This key protects critical data in my account
Account	This account [REDACTED]
KMS Key ID	[REDACTED] 33d40cac295

Cancel Create

CL

```
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]
```

Chiffrement côté serveur dans AWS

Amazon Redshift

✓

✓

○

|

CLUSTER DETAILS NODE CONFIGURATION **ADDITIONAL CONFIGURATION** REVIEW

Provide the optional additional configuration details below.



Cluster Parameter Group Parameter group to associate with this cluster.

Encrypt Database

☐ None ☒ KMS ☐ HSM

[Learn more about database encryption](#)

Master Key

Description

Protects critical data in my applications

Account

This account ()

KMS Key ID

-ca8a1a92204f

Pour S3, il existe une solution
encore plus simple

Méthodes de chiffrement

Chiffrement côté client

Client-side encryption

- Clés gérées par le client
- Clés gérées par AWS KMS

Chiffrement côté serveur

Server-side encryption, i.e. SSE

- S3 : clés fournies par le client (SSE-C)
- Clés gérées par AWS KMS (SSE-KMS)
- S3 : clés gérées par Amazon S3 (SSE-S3)

Chiffrement côté serveur

Clés gérées par Amazon S3 (SSE-S3)

Clés gérées par Amazon S3 (SSE-S3)

- Les données au repos dans S3 sont chiffrées **par le service lui-même**. Les métadonnées ne sont pas chiffrées.
- Chaque objet est chiffré avec **AES-256** et une **clé unique**.
- Les clés sont elles-mêmes **chiffrées** avec une clé principale qui change **régulièrement**.
- Le chiffrement peut se faire :
 - dans la console
 - programmatically avec un SDK AWS
 - par API en ajoutant l'en-tête HTTP `x-amz-server-side-encryption`



Exemple – SSE-S3 avec la console AWS

Définir les détails Annuler

Charger vers : [Tous les compartiments](#) / [jsimon-public](#)

Détails : définissez des détails supplémentaires pour tous les objets que vous chargez. Vous pouvez choisir entre Stockage standard, [stockage RRS](#) et [Standard - accès peu fréquent](#). Vous pouvez également décider ou non de [chiffrer vos fichiers](#).

☐ Utiliser le stockage standard ☐ Utiliser le stockage standard - accès peu fréquent ☐ Utiliser le stockage RRS

☒ Utiliser le chiffrement côté serveur [En savoir plus](#)

☒ **Utiliser la clé principale de service Amazon S3.**
S3 déchiffre l'objet pour quiconque détient l'autorisation d'accéder à cet objet.

☐ **Utiliser une clé principale AWS Key Management Service**
S3 déchiffre l'objet pour quiconque détient l'autorisation d'accéder à cet objet ainsi que l'autorisation d'utiliser la clé principale.

< Sélectionner les fichiers Définir des autorisations > Commencer le chargement Annuler

Exemple – SSE-S3 avec le SDK Java

```
File file = new File(uploadFileName);
PutObjectRequest putRequest = new PutObjectRequest(bucketName, keyName, file);

// Request server-side encryption.
ObjectMetadata objectMetadata = new ObjectMetadata();
objectMetadata.setSSEAlgorithm(ObjectMetadata.AES_256_SERVER_SIDE_ENCRYPTION);

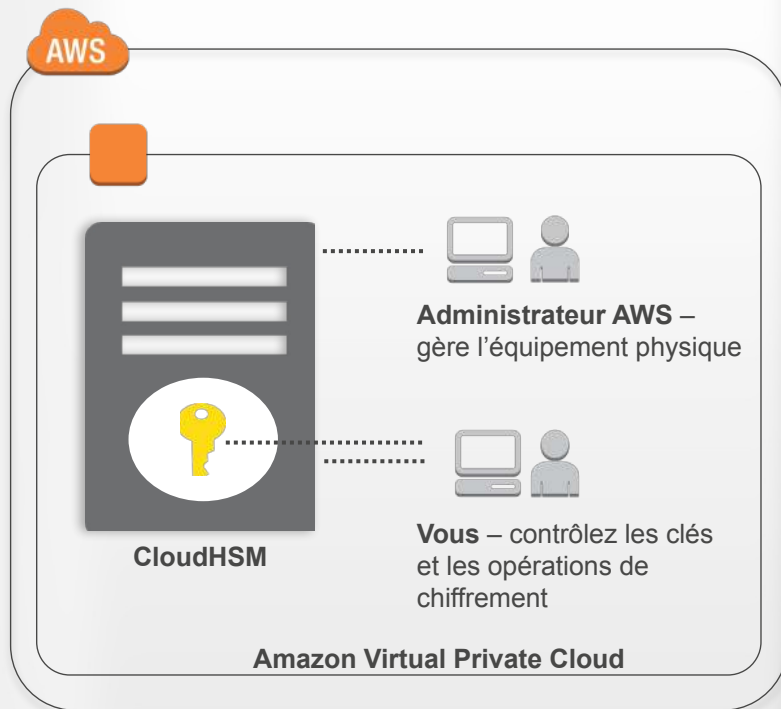
putRequest.setMetadata(objectMetadata);

PutObjectResult response = s3client.putObject(putRequest);
System.out.println("Uploaded object encryption status is " +
    response.getSSEAlgorithm());
```

Certaines organisations ont besoin
d'un niveau de sécurité
encore plus élevé

AWS CloudHSM

- Equipement SafeNet Luna SA
 - Hébergé par AWS
 - Dans **votre VPC**
 - Réservé à **votre usage exclusif**
- Chiffrement **symétrique** ou **asymétrique**
- Vous seul avez accès aux **clés** et aux **actions** effectuées sur les clés



AWS CloudHSM

Disponible dans onze régions

- GovCloud, USA Est (Virginie du Nord, Ohio), USA Ouest (Oregon, Californie), Canada, UE (Irlande, Francfort) et Asie-Pacifique (Sydney, Tokyo, Singapour).

Conformité


- PCI DSS, SOC, FIPS 140-2, Common Criteria EAL4+

Cas d'utilisation

- Chiffrement pour Amazon Redshift et RDS Oracle (TDE)
- Intégration avec un logiciel tiers
- Intégration avec votre propre application



Tarification de CloudHSM

- \$5,000 à la mise en œuvre
 - \$1,88 / heure (eu-west-1)
 - Le tarif varie selon la région
 - \$21,500 la première année
 - \$16,500 les années suivantes
- 

Comparaison entre CloudHSM et KMS

AWS CloudHSM

- Le plus haut niveau de sécurité
- Vous contrôlez totalement vos clés et l'application qui les utilise
- Applications prises en charge :
 - Votre logiciel personnalisé
 - Un logiciel tiers
 - Amazon Redshift, RDS Oracle

AWS KMS

- Solution de gestion des clés hautement disponible, durable et auditable
- Chiffrez aisément vos données avec les services AWS et dans vos applications en fonction des stratégies IAM définies
- Applications prises en charge :
 - Votre logiciel personnalisé (AWS SDK)
 - Services AWS (S3, EBS, RDS, Amazon Redshift, WorkMail, Elastic Transcoder, etc.)

Solutions partenaires sur AWS Marketplace

- Testez en toute liberté avant d'acheter
- Paiement horaire, mensuel ou annuel
- Frais ajoutés à la facture AWS
- *Bring Your Own License*



Comparaison entre les options de gestion des clés

	AWS Key Management Service
Emplacement de génération et de stockage des clés	AWS
Emplacement d'utilisation des clés	AWS ou vos applications
Méthode de contrôle de l'utilisation des clés	Stratégie que vous définissez, appliquée dans AWS
Responsabilité pour les performances/ l'évolution	AWS
Intégration aux services AWS ?	Oui
Modèle de tarification	Par clé/utilisation

Comparaison entre les options de gestion des clés

	AWS Key Management Service	AWS CloudHSM
Emplacement de génération et de stockage des clés	AWS	Dans AWS, sur un HSM que vous contrôlez
Emplacement d'utilisation des clés	AWS ou vos applications	AWS ou vos applications
Méthode de contrôle de l'utilisation des clés	Stratégie que vous définissez, appliquée dans AWS	Code client + API SafeNet
Responsabilité pour les performances/ l'évolution	AWS	Vous
Intégration aux services AWS ?	Oui	Limitée
Modèle de tarification	Par clé/utilisation	Par heure

Comparaison entre les options de gestion des clés

	AWS Key Management Service	AWS CloudHSM	Solutions partenaires AWS Marketplace
Emplacement de génération et de stockage des clés	AWS	Dans AWS, sur un HSM que vous contrôlez	Votre réseau ou dans AWS
Emplacement d'utilisation des clés	AWS ou vos applications	AWS ou vos applications	Votre réseau ou votre instance EC2
Méthode de contrôle de l'utilisation des clés	Stratégie que vous définissez, appliquée dans AWS	Code client + API SafeNet	Gestion propre au fournisseur
Responsabilité pour les performances/ l'évolution	AWS	Vous	Vous
Intégration aux services AWS ?	Oui	Limitée	Limitée
Modèle de tarification	Par clé/utilisation	Par heure	Par heure/par an

Comparaison entre les options de gestion des clés

	AWS Key Management Service	AWS CloudHSM	Solutions partenaires AWS Marketplace	DIY
Emplacement de génération et de stockage des clés	AWS	Dans AWS, sur un HSM que vous contrôlez	Votre réseau ou dans AWS	Votre réseau ou dans AWS
Emplacement d'utilisation des clés	Services AWS ou vos applications	AWS ou vos applications	Votre réseau ou votre instance EC2	Votre réseau ou votre instance EC2
Méthode de contrôle de l'utilisation des clés	Stratégie que vous définissez, appliquée dans AWS	Code client + API SafeNet	Gestion propre au fournisseur	Fichiers de configuration, gestion propre au client
Responsabilité pour les performances/ l'évolution	AWS	Vous	Vous	Vous
Intégration aux services AWS ?	Oui	Limitée	Limitée	Limitée
Modèle de tarification	Par clé/utilisation	Par heure	Par heure/par an	Variable

Ressources

AWS Key Management Service <https://aws.amazon.com/kms>

AWS CloudHSM <https://aws.amazon.com/cloudhsm/>

Blog sécurité AWS <http://blogs.aws.amazon.com/security>

AWS re:Invent 2016: Encryption: It Was the Best of Controls, It Was the Worst of Controls (SAC306) <https://www.youtube.com/watch?v=zmMpgblhCpw>

Livre blanc sur les détails cryptographiques d'AWS Key Management Service <https://d0.awsstatic.com/whitepapers/KMS-Cryptographic-Details.pdf>

Merci !

Julien Simon

Principal Technical Evangelist, AWS

julsimon@amazon.fr

@julsimon

Lundi

- Bonnes pratiques d'authentification avec AWS IAM
- Chiffrez vos données avec AWS

Mardi

- Fireside chat avec Matthieu Bouthors et Julien Simon
- Re:Invent update 1

Mercredi

- Deep dive : Amazon Virtual Private Cloud
- Bonnes pratiques anti-DDoS

Jeudi

- Re:Invent update 2
- Gérez les incidents de sécurité avec AWS CloudTrail

Vendredi

- Automatisez vos audits de sécurité avec Amazon Inspector
- Bonnes pratiques de sécurité sur AWS