飯塚 将太
アマゾン ウェブ サービス ジャパン
IoT ソリューションアーキテクト
2020/2/13作成

AWS INNOVATE 2020 オンラインカンファレンス

「IoTにおけるセキュリティ考慮事項とAWSを活用した対策のご紹介」のクイズおよび解答

AWS INNOVATE 2020のセッションの視聴およびアンケートにご記入頂きありがとうございます。本資料が「IoTにおけるセキュリティ考慮事項とAWSを活用した対策のご紹介」で出題されたクイズの解答になります。

問題１：AWS IoT Coreに接続するため、デバイス証明書を使用して認証することにしました。各デバイスへ個別の証明書をプロビジョニングするためにはどのような手段があるでしょうか？

解答1: セッション内では、IoTデバイスの認証におけるベストプラクティスの一つとして「**デバイスに個別の認証情報を割り当てる**」と紹介させていただきました。セッション時間の関係上、具体的なその方法についてはセッション内でご紹介できなかったため、ここで説明します。

今回は問題にある通り、AWS IoTを利用する際、**各デバイスへ個別の証明書**をプロビジョニングする手段について紹介します。AWS IoTではお客様の要件やニーズにお応えするため、以下のようないくつかの手段をご用意しています。

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/x509-client-certs.html

1) AWS IoT に証明書と秘密鍵を発行してもらう方法
2) 秘密鍵と CSR をデバイス側で作成し、CSR を AWS IoT に送信し署名してもらうことで証明書を発行する方法
3) 自ら発行したデバイス証明書を事前に AWS IoT に登録しておく方法
4) 自ら発行したデバイス証明書を初回接続時に AWS IoT に送信し、JITR によってデバイス証明書を登録する方法

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

飯塚 将太
アマゾン ウェブ サービス ジャパン
IoT ソリューションアーキテクト
2020/2/13作成

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/device-certs-your-own.html#auto-register-device-cert

5) 自ら発行したデバイス証明書を初回接続時に AWS IoT に送信し、JITP のプロビジョニングテンプレートでデバイス証明書を登録およびポリシーの作成、デバイス登録をする方法

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/jit-provisioning.html

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/provision-template.html

また、それぞれの手段の特徴や差異を以下の表にまとめます。

個別証明書のプロビジョニング手段	1)	2)	3)	4)	5)
証明書の発行	AWS IoT*1		お客様*2		
証明書の有効期限	指定できない		任意に設定可能		
秘密鍵の発行	AWS IoT	任意の場所			
AWS IoTへの証明書の事前登録	必要			不要	

*1 https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/device-certs-create.html

*2 https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/device-certs-your-own.html

まず、デバイス証明書を利用するためには、証明書を発行し署名する認証局（以下CA）が必要です。AWS IoTでは、**AWS IoTが持つCA**や**お客様が用意したCA**を利用することができます。そのため、AWS IoTが発行した証明書やお客様が用意した証明書の両方を利用することができます。独自にCAを用意する場合には、そのCA証明書を事前にAWS IoTへ送信し、CAを登録しておく必要があります。

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/device-certs-your-own.html#register-CA-cert

また、1)の手段では秘密鍵の発行をAWS IoTに任せることができます。しかしAWS IoTから秘密鍵をダウンロードしなければならないため、秘密鍵がインターネット経路を通ります。一方でその他の手段では、デバイスもしくは生産や出荷設備の中で秘密鍵を作成することができるため、秘密鍵の経路を制御できます。

このように、デバイスに個別の認証情報を割り当てるにあたっては、デバイス証明書を発行するCAや証明書を事前に登録するかなどの違いがあります。これらの手段は、お客様のセキュリティポリシーや、

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

飯塚 将太
アマゾン ウェブ サービス ジャパン
IoT ソリューションアーキテクト
2020/2/13作成

生産や出荷プロセスでの制約、CAを運営できるかなどの条件によって決まります。要件や制約から適切な手段をお選びください。

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

飯塚 将太
アマゾン ウェブ サービス ジャパン
IoT ソリューションアーキテクト
2020/2/13作成

問題２：AWS IoT Coreに送信されたデータをクラウド上に暗号化してS3に保存したいです。どのようなアーキテクチャで実現できるでしょうか？

解答 2: まず、Amazon S3 で保管時のデータを保護するには、暗号化する箇所によって 2 つの手段を選択できます。
https://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/UsingEncryption.html

サーバー側で暗号化するか、クライアント側で暗号化するかとなります。サーバー側の暗号化では、オブジェクトを S3 に保存する前に暗号化し、オブジェクトをダウンロードするときに復号するよう S3 にリクエストします。
https://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/serv-side-encryption.html

クライアント側の暗号化では、クライアント側でデータを暗号化し、暗号化したデータをS3にアップロードします。この場合、暗号化プロセス、暗号化キー、関連ツールはお客様が管理してください。

https://docs.aws.amazon.com/ja_jp/AmazonS3/latest/dev/UsingClientSideEncryption.html

そして、AWS IoT Coreに転送されたデータをこのS3バケットに保存するためには、以下の図のようなアーキテクチャを実装します。ルールエンジンをAWS IoT Coreで作成し、その中のルールアクションでAmazon Kinesis Data Firehoseを指定し設定します。またFirehoseでは、S3のバケットを指定します。

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/iot-rules.html
https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/iot-rule-actions.html
https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/iot-rule-actions.html#firehose-rule

ルールアクションでは、S3を指定することもできます。

https://docs.aws.amazon.com/ja_jp/iot/latest/developerguide/iot-rule-actions.html#s3-rule

しかしこの場合、AWS IoT Coreに送信されたメッセージごとにファイルがS3上に作成されます。IoTの場合、大量の細かなメッセージが送信されてくることが想定されるため、こういったデータの場合、S3に大量の小さなファイルが作成されてしまうことになります。そのようにファイルが小さく細かく分

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

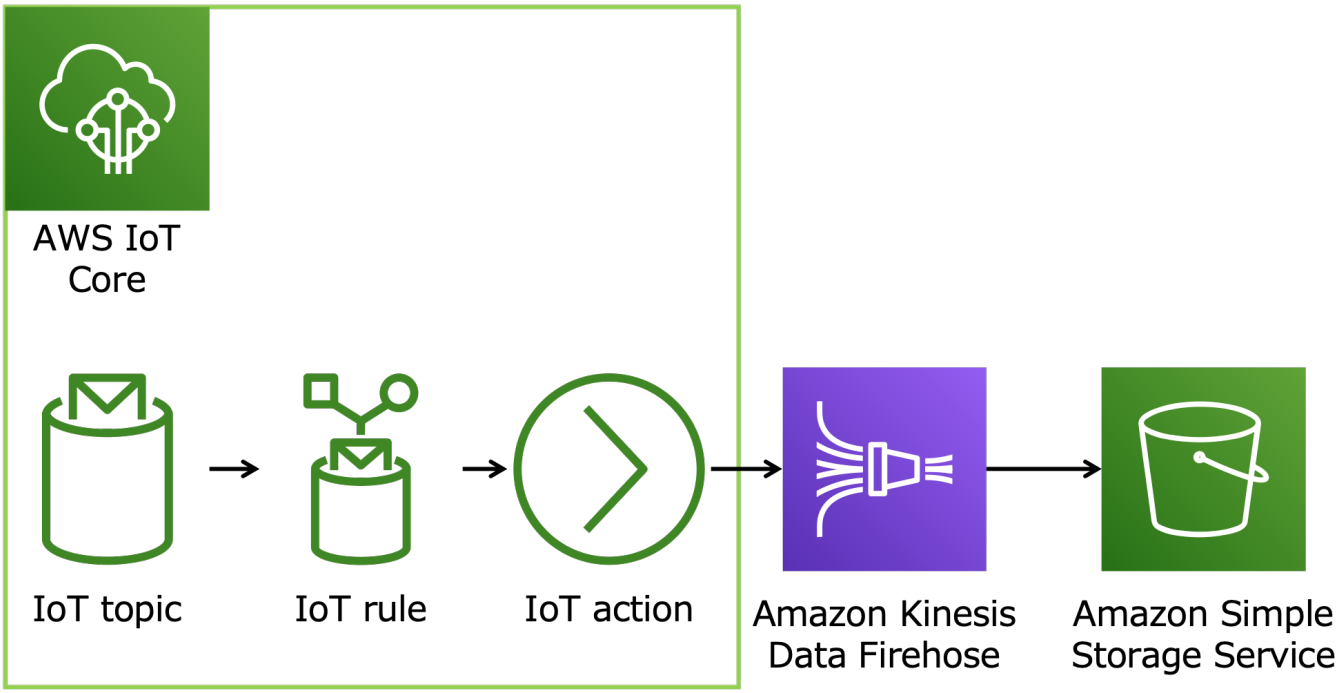
飯塚 将太
アマゾン ウェブ サービス ジャパン
IoT ソリューションアーキテクト
2020/2/13作成

割されていると、大量データの分析時にパフォーマンスを得られない可能性があります。そういった場合では、いくつかのメッセージのまとまりごとにファイルが作成され保存されていた方が、分析時にパフォーマンスを得られやすいです。図で示したアーキテクチャのように、FirehoseをAWS IoT CoreとS3の間に挟むことにより、Firehoseがメッセージを一定のまとまりごとにファイルとしてデータをS3へ保存してくれます。したがって、分析時のパフォーマンスの面も考慮すると、このような構成にすることを推奨します。

また、Firehoseでもサーバー側暗号化を利用できます。そのため暗号化オプションを有効化することで、一時的にFirehoseに保存されるデータも暗号化することが可能です。

https://docs.aws.amazon.com/ja_jp/firehose/latest/dev/encryption.html

以上から、デバイスからS3までEnd-to-Endでデータを暗号化して保護することができます。このようなアーキテクチャを実装いただくことで、IoTデータやログをクラウドへ送信し、安全にS3へ保存することができます。



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