デバイスの製造と AWS IoT Core での X.509 証明書のプロビジョニング

2021 年 1 月

注意

お客様は、この文書に記載されている情報を独自に評価する責任を負うものとします。このドキュメントは、(a) 情報提供のみを目的としており、(b) AWS の現行製品とプラクティスを表したものであり、予告なしに変更されることがあり、(c) AWS およびその関連会社、サプライヤー、またはライセンサーからの契約義務や確約を意味するものではありません。AWS の製品やサービスは、明示または暗示を問わず、いかなる保証、表明、条件を伴うことなく「現状のまま」提供されます。お客様に対する AWS の責任は、AWS 契約により規定されます。本書は、AWS とお客様の間で行われるいかなる契約の一部でもなく、そのような契約の内容を変更するものでもありません。

目次

要約

このホワイトペーパーでは、一意の認証情報を使用した AWS IoT Core での IoT (モノのインターネット) デバイスのオンボーディングに焦点を当てています。証明書ベースの相互認証のために、デバイスを製造して、一意の X.509 証明書および秘密鍵をデバイスにプロビジョニングするためのさまざまなオプション、課題、および考慮事項について説明します。
このホワイトペーパーでは、各社のデバイスおよび製造プロセスの能力に基づいて、適切な AWS IoT プロビジョニングオプションに関するガイダンスをデバイスメーカーに提供します。Sigv4 およびカスタムオーソライザーの認証方法については扱いません。
このホワイトペーパーは、技術アーキテクト、IoT クラウドエンジニア、IoT セキュリティアーキテクト、組み込みエンジニアを対象としています。このホワイトペーパーでは、公開鍵基盤 (PKI) と Transport Layer Security (TLS) の基本的な概念と用語について読者が理解していることを前提としています。

注意: このホワイトペーパーに記載されているインスタンスおよびサポートバージョンの情報は、2021 年 1 月時点のものです。それ以降にお読みになられた場合は、情報が更新されている可能性がありますので、AWS IoT の製品ページまたは AWS IoT Core のドキュメントページにて最新情報をご確認ください。

はじめに

IoT の展開では、多くの場合セキュリティはデバイスユーザーとデバイスメーカーの両方にとって最大の懸念事項です。IoT デバイスから AWS IoT Core に転送中のデータを保護および暗号化するため、AWS IoT Core では X.509 証明書を使用した TLS ベースの相互認証がサポートされています。デバイスメーカーは、一意の秘密鍵と X.509 証明書を含む一意の認証情報を各デバイスにプロビジョニングする必要があります。さらに、デバイスメーカーは、デバイスごとに必要なクラウドリソースをアマゾン ウェブ サービス (AWS) でセットアップする必要もあります。
それらの一意の認証情報をプロビジョニングし、AWS IoT Core にオンボーディングする方法は、IoT デバイスの開発および製造のフェーズごとに異なります。デバイスメーカーは、IoT デバイスのライフサイクルにおいて、次のようにいくつかの点を考慮する必要があります。

- 顧客が所有する認証機関 (CA)、サードパーティー CA、または AWS IoT CA の使用
- セキュアエレメントなど、ハードウェアセキュリティモジュールの使用
- デバイスプロビジョニングプロセスをサポートするのに必要なクラウドリソース
- オンボーディング手順を実装するためのデバイスレベルのロジック

このホワイトペーパーでは、デバイス製造のサプライチェーンの複雑さについて説明し、各社のデバイスの能力と製造プロセスの制限事項に基づいて、これらの決定に関する推奨事項をデバイスメーカーに示します。

AWS IoT Core のリソース

デバイスが AWS IoT Core に接続して通信するには、AWS IoT Core に IoT Thing 、証明書、IoT ポリシーが必要です。

- **IoT Thing** — AWS では、デバイスを IoT Thing として Thing Registry に登録することを強くお勧めします。Thing とは、一意の名前と静的属性を含む物理デバイスを表すクラウドベースの表現です。

- **X.509 証明書** — Thing ごとに X.509 証明書のアタッチが必要です。証明書は、Thing ごとに一意である必要があります。X.509 証明書には、CA 発行者、公開鍵、有効期限などの公開情報が含まれています。公開鍵は、デバイスにのみ保持される秘密鍵を含む非対称キーペアの一部です。
- **IoT ポリシー** — IoT ポリシーは、デバイスが実行することを許可されたアクションを定義するドキュメントです。IoT ポリシーは、X.509 証明書にアタッチする必要があります。ポリシーは、ポリシー変数を使用して多くのデバイス間で共有できます。

デバイス上のリソース

デバイスが TLS ベースの相互認証を使用して AWS IoT Core に接続するには、Amazon Trust Services サーバー証明書、X.509 証明書、秘密鍵のほか、場合によってはデバイスのクライアント証明書の発行元 CA も使用してデバイスをプロビジョニングする必要があります。

- **X.509 証明書** — AWS に存在するものと同じ X.509 証明書がデバイスにも存在している必要があります。この証明書は、AWS IoT Core との TLS ハンドシェイク時に提示されます。
- **秘密鍵** — デバイスの秘密鍵は、X.509 証明書で提示される公開鍵と非対称的にペアになります。秘密鍵は、True Random Number Generator を使用してデバイスで生成されるのが理想的です。デバイスから外部に取り出すべきではありません。
- **署名者認証機関** — ジャストインタイムデバイスオンボーディングの場合、デバイスは X.509 証明書の発行元 CA を最初の TLS 接続で送信する必要があります。それ以降の接続では、発行元 CA 証明書は必要ありません。
- **サーバー証明書** — Amazon Trust Services (ATS) サーバー証明書は、デバイスが正規の AWS IoT ATS エンドポイントに接続していることを検証するために使用されます。

開発時のデバイスプロビジョニング

IoT プロジェクトの初期フェーズでは、開発およびテストの目的で、少数のデバイスが AWS IoT にプロビジョニングされます。このフェーズでは、開発者は多くの場合、セキュリティとスケーラビリティよりも利便性を選択します。

AWS IoT では、AWS マネジメントコンソール、AWS IoT コントロールプレーン API、または AWS Command Line Interface (AWS CLI) から、単一の Thing をプロビジョニングするために必要なすべてのリソースを作成できます。証明書は AWS CA によって発行され、秘密鍵は AWS クラウドで生成されます。証明書、秘密鍵、公開鍵は、開発者のローカルマシンにダウンロードされます。

ローカルマシンからデバイスに証明書と秘密鍵を手動でプロビジョニングするのは開発者です。証明書と秘密鍵は、デバイスのファイルシステムに追加することも、デバイスのファームウェアコードに直接書き込むこともできます。

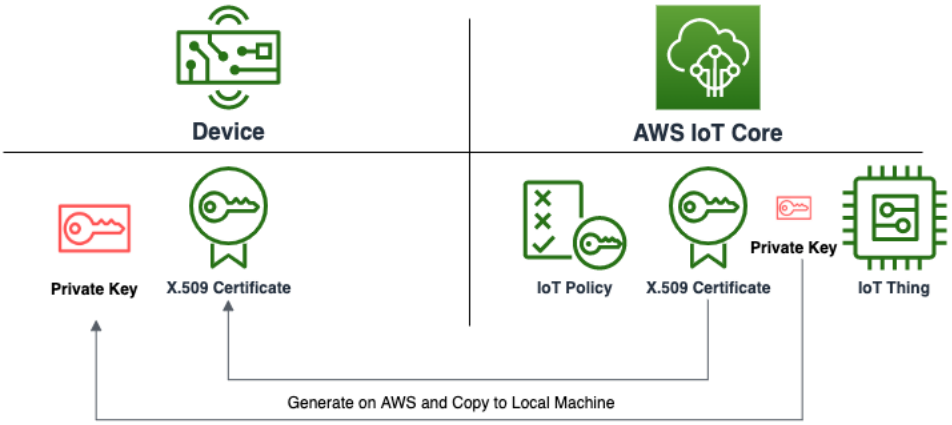

図 *1: AWS IoT Core* を使用して開発時にデバイスをプロビジョニングするプロセス

開発者のローカルマシンが、ソーシャルエンジニアリング、ユーザーエラー、または脆弱なパスワードが原因で侵害される可能性があるため、このプロセスを本番稼働環境で使用しないでください。

現場にデバイスを設置しないため、セキュリティリスクを抑えられます。デバイスは、開発者によ

り、ラボ環境で厳重に管理されます。侵害されたデバイスを再プログラミングしたり、証明書を無効にしたりすることもできます。

AWS クラウドですべてのリソースを作成し、必要なキーをデバイスにコピーするプロセスは手動であり、時間がかかります。開発者が独自のファームウェアとファイルを各デバイスに書き込む必要があります。このシナリオでは、ほとんどの操作はラボ環境で実行されますが、IoT プロジェクトのパイロットフェーズまたは本番稼働フェーズに入る際にプロセスをスケーリングすることはできません。IoT プロジェクトをスケーリングする場合は通常、より複雑なサプライチェーンが存在します。

デバイス製造サプライチェーン

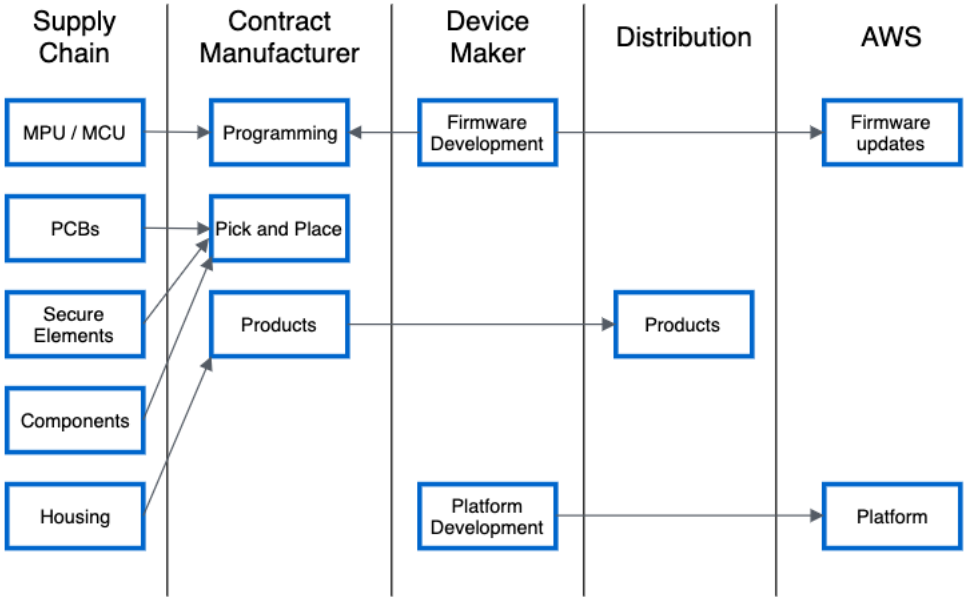

図 *2: IoT* デバイスの製造プロセス

IoT プロジェクトが開発フェーズから本番稼働フェーズに移行すると、デバイスメーカーから顧客までのサプライチェーンが構築されます。

デバイスメーカーは、デバイスのコンポーネント、物理仕様、機能の仕様を定めます。製品の設計、製品のハードウェアとソフトウェアの開発、AWS クラウドアプリケーションの開発と保守、テンプ

レート、ポリシー、リソースのプロビジョニング、製品の販売とマーケティングを行うのはデバイスメーカーです。ほとんどのデバイスメーカーは、大量の製品を物理的に製造する設備を持たないため、この製造を受託製造業者にアウトソーシングする必要があります。

プロトタイピングの段階では、デバイスメーカーは多品種少量契約製造を利用して、少量のエンジニアリングサンプルをすばやく生成できます。プロトタイピングから本番稼働フェーズに移行する際、スケールメリットを活かすために大量受託製造業者を利用します。

少品種大量受託製造業者は、大量のデバイスを製造するための生産ライン、ツール、およびプロセスを備えています。受託製造業者は、デバイスメーカーから提供された仕様に従ってデバイスを構築します。この仕様には、プリント回路基板 (PCB) のスキーム、部品表、デバイスに書き込まれるファームウェアが含まれます。受託製造業者は PCB にコンポーネントを配置し (ピックアンドプレース)、デバイスメーカーから提供されたファームウェアと認証情報をプロセッサに書き込み、デバイスを最終製品にパッケージ化します。その後、販売するために製品を流通チャネルに提供します。受託製造業者は、個々のコンポーネントを調達して、サプライチェーンから最終製品を構築する必要があります。

契約製造サプライチェーンに含まれるベンダーは、デバイスで使用される個々のコンポーネントを製造します。コンポーネントの例としては、マイクロコントローラー、プロセッサ、セキュアエレメント、接続モジュールなどがあります。コンポーネントベンダーは、契約製造元までのサプライチェーンを実現するために、ディストリビューターに依存している場合があります。

デバイスの製造後、在庫レベルやエンジニアリングサンプルを維持するためにデバイスメーカーに送付されることがあります。デバイスを販売してエンドカスタマーまでのサプライチェーンを担うディストリビューターにもデバイスが送付されることがあります。

デバイス製造サプライチェーンにおけるプロビジョニングのタイミング

デバイス製造サプライチェーンを作成するとき、デバイスメーカーは、デバイスが一意の X.509 証明書と秘密鍵を受け取るタイミングを指定する必要があります。

デバイスメーカーは、ファームウェアに含まれる一意の認証情報を使用してデバイスをプロビジョ

ニングすることを選択できます。このシナリオでは、コードに X.509 証明書と秘密鍵を含むデバイスごとに一意のファームウェアが生成されます。一意のファームウェアイメージはそれぞれ受託製造業者に送られます。受託製造業者には、製造時に各ファームウェアイメージをデバイスのプロセッサに書き込める必要があります。このシナリオでは、サプライチェーンに対する権限が最も高いのはデバイスメーカーであり、AWS IoT に各デバイスを事前登録することができます。このため、受託製造業者から見るとかなり複雑になります。

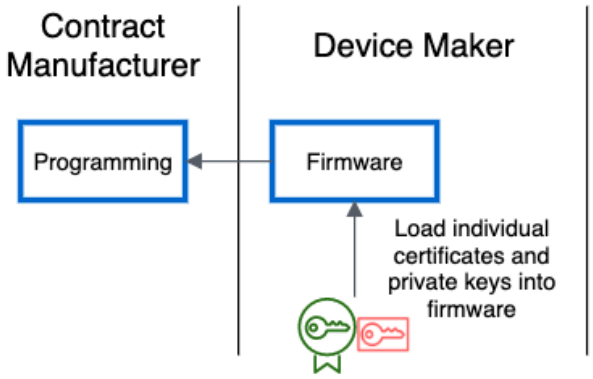

図 3: ファームウェアへの認証情報のプロビジョニング

受託製造業者は、製造プロセス中にカスタマイズを実行できることがあります。このシナリオでは、ゴールデンイメージと呼ばれる単一のファームウェアイメージが受託製造業者に提供され、製造業者がデバイスに認証情報を提供する責任を負います。ファームウェアには、デバイスが Secure Shell (SSH) やネットワークファイルシステム (NFS) などのオープンインターフェイス経由またはシリアル接続経由で認証情報を受け入れ、それらの認証情報をデバイス上のセキュアな場所に保存することを可能にする追加のロジックが必要です。セキュリティ認証情報は製造環境で利用され、PKI は受託製造業者によって処理されます。プロビジョニングプロセスは、信頼できる人がセキュアな環境で実行することが重要です。

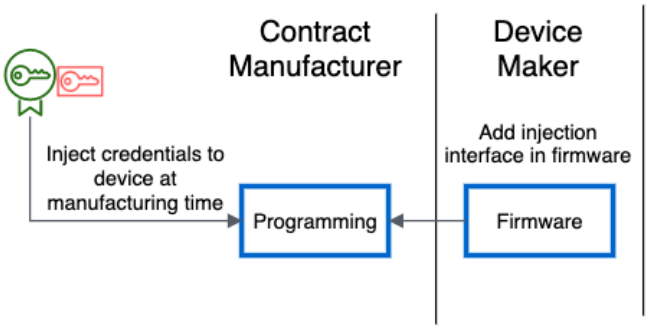

図 *4:* 製造時における認証情報の書き込み

製造時に各デバイスにカスタマイズを取り込むと、製造されるデバイスごとにそのカスタマイズを追跡する必要があるため、各デバイスを製造するための貴重な時間が奪われたり、ロジスティクスのオーバーヘッドが増えたりする可能性があります。これにより、受託製造業者がデバイスメーカーに請求するユニットあたりのコストが増加します。

デバイスメーカーは、セキュアエレメントやトラステッドプラットフォームモジュール (TPM) などのハードウェアセキュリティモジュール (HSM) をデバイスに追加することを選択できます。ハードウェアセキュリティモジュールベンダーは、ベンダーのセキュアな施設で秘密鍵を生成し、X.509 証明書に署名するプロセスを用意しています。これにより、デバイスメーカーは HSM と通信するロジックを使用して、単一の標準ファームウェアイメージを受託製造業者に提供できるようになります。HSM を PCB に配置する責任を負うのは受託製造業者ですが、認証情報は開示されず、認証情報をプロビジョニングするために追加のプロセスは必要ありません。

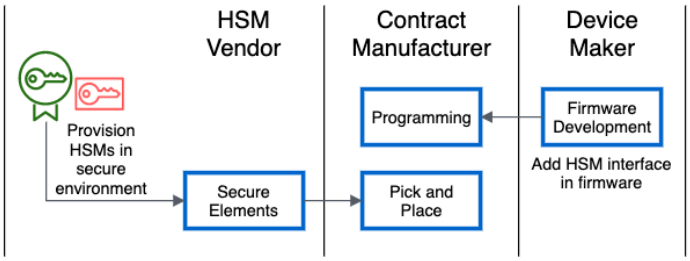

図 *5:* ハードウェアセキュリティモジュールのベンダーによるセキュアエレメントへの認証情報のプロビジョニング

アプリケーションと認証情報のセキュアな書き込みなど、付加価値の高いサービスがサードパーティーによって提供されています。デバイスメーカーは、信頼できるディストリビューターやサプ

ライチェーン内のサードパーティーを使用して、デバイスに認証情報をプロビジョニングできます。これは、製造委託先で機器が組み立てられる前のサプライチェーンで起こる場合もあれば、最終的な顧客への販売を履行するためにサプライチェーンで機器が生産された後に起こる場合もあります。

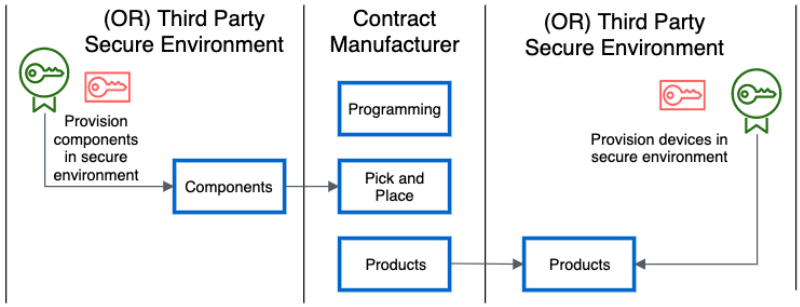

図 *6:* サードパーティーベンダーによるセキュアな環境での認証情報のプロビジョニング

サプライチェーンでプロビジョニングされないデバイス

製造業者、コンポーネントベンダー、またはディストリビューターは、カスタマイズサービスを提供する場合の最小注文数を設けている場合があります。少量生産のデバイスでは、カスタマイズや事前プロビジョニングされたハードウェアセキュリティモジュールの導入にコストがかかる可能性があります。コンシューマー製品では、デバイスをプロビジョニングする場所が製造時にわからないことがあります。このようなシナリオでは、デバイスは一意の認証情報なしで製造を完了します。デバイスは、フリートプロビジョニングと呼ばれるサービスを使用して、販売されて現場に設置された後にプロビジョニングできます。

認証機関と信頼チェーンの管理

認証機関は X.509 証明書を発行して署名します。 デバイスメーカーは、AWS IoT を CA として使用するか、独自の CA を使用するか、サードパーティーの CA を使用するかを決定する必要があります。

AWS IoT では、クラウドにおいて X.509 証明書と秘密鍵を生成できます。X.509 証明書は AWS IoT CA によって署名され、作成時にデバイスメーカーの AWS IoT レジストリに事前登録されます。作成後、デバイスメーカーは証明書と秘密鍵をダウンロードし、製造時に証明書と秘密鍵をデバイ

スに配布する必要があります。

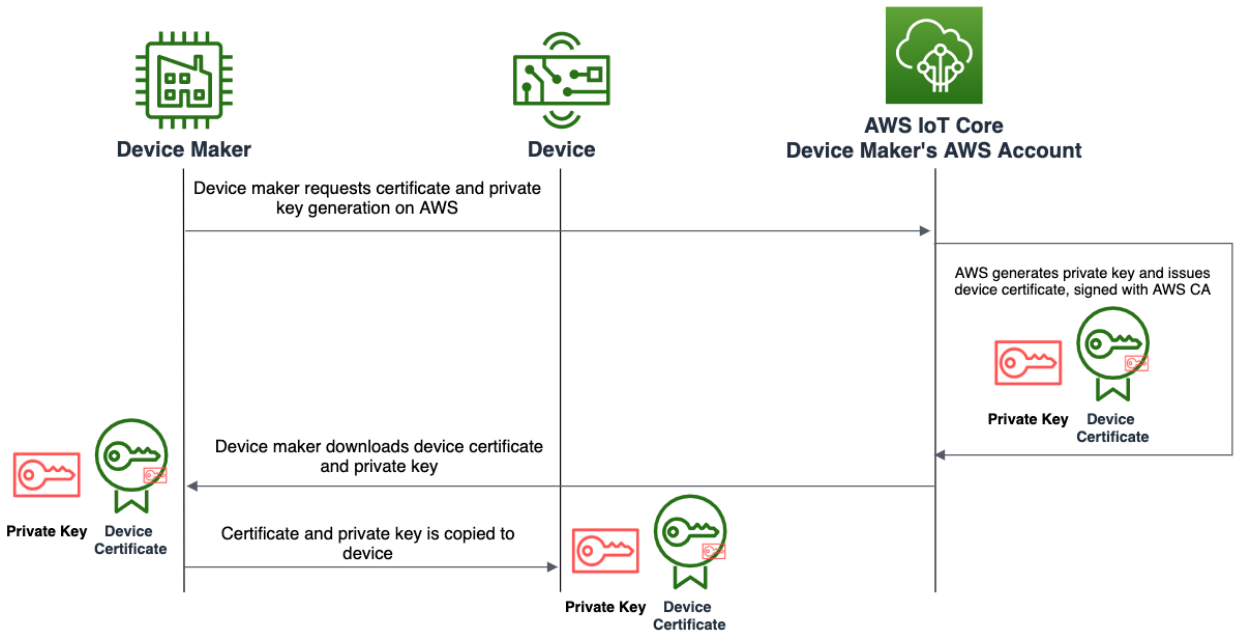

図 *7: AWS* で生成された *X.509* 証明書と秘密鍵

オンボーディングされたデバイスにすでに秘密鍵がある場合は、証明書署名リクエストを AWS に送信して、デバイスの秘密鍵を外部に晒すことなく証明書に署名できます。

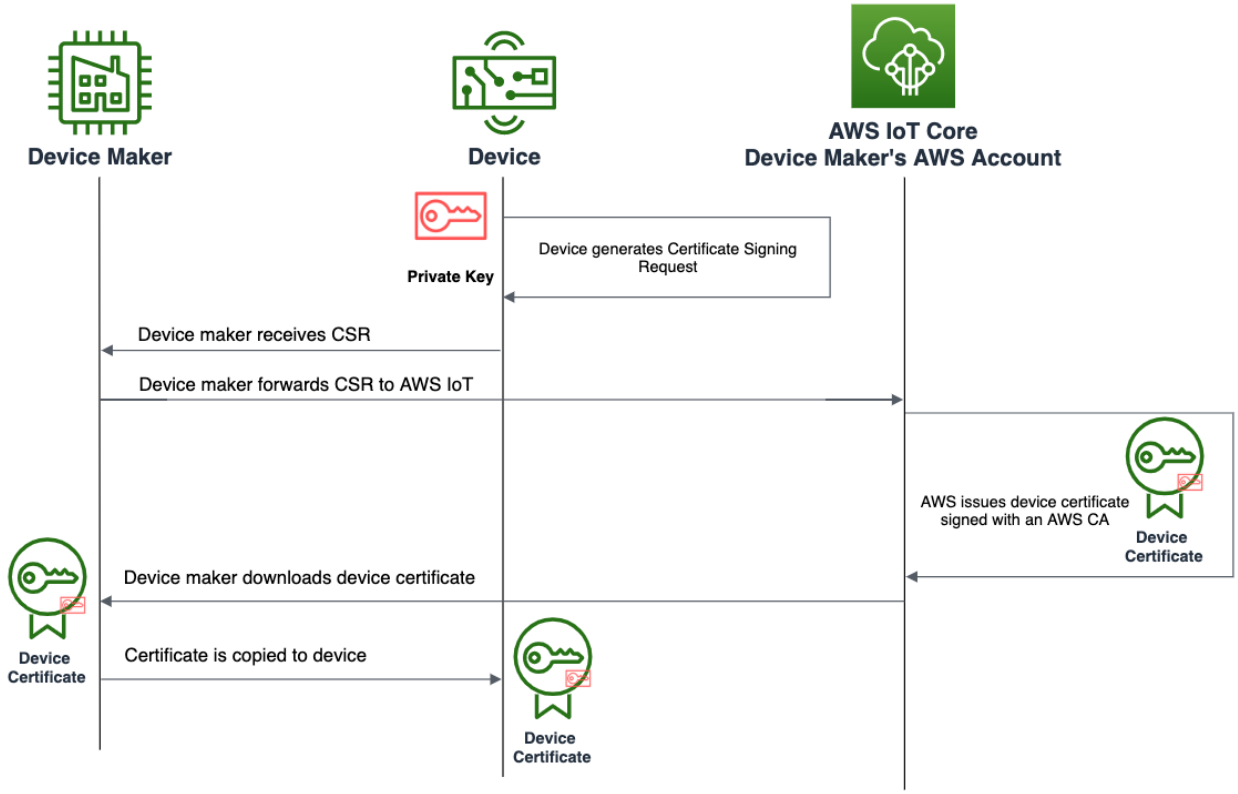

図 *8:* デバイスから *AWS* に行われる証明書署名リクエスト

証明書と秘密鍵は、各デバイスのファームウェアに含めるか、デバイスに配布するために受託製造業者に提供する必要があります。AWS CA は AWS 責任共有モデルで保護されているため、デバイスメーカーは CA に独自のコントロールを実装する必要はありません。デバイスメーカーは、AWS アカウントのユーザーに認可ポリシーを付与し、認可されたユーザーのみが新しい証明書を生成できるようにします。

デバイスメーカーが CA と公開鍵基盤を管理する必要がある場合、AWS IoT は顧客が所有の CA を使用するオプションを提供します。デバイスはセキュアなネットワークチャネルを介して CA と直接やり取りし、製造プロセス中に証明書署名リクエストを作成します。デバイスが CA に直接アクセスできない場合、証明書と秘密鍵を事前に生成してデバイスのファームウェア、ハードウェアセキュリティモジュールに書き込むか、製造プロセス中にセキュアなローカル接続を介して配布できます。デバイスが初めて接続する前に証明書を AWS IoT に登録する必要があります。

大企業は通常、独自の自己署名ルート CA を持っています。自己署名 CA を使用すると、公開鍵

インフラストラクチャに対する最高レベルの柔軟性とコントロールが実現します。CA の侵害を防ぐには、キー署名セレモニーや物理的なアクセス制御など、厳格なセキュリティプロトコルを採用する必要があります。認証機関が自己署名されている場合、通常は 1 つ以上の中間署名者証明書のチェーンがあります。これにより、証明書インフラストラクチャの他の部分に影響を与えずに中間証明書を取り消すことができるため、デバイスメーカーは証明書失効リストをより厳密にコントロールできます。デバイス証明書の発行元署名者 CA は、AWS IoT に登録されている必要があります。

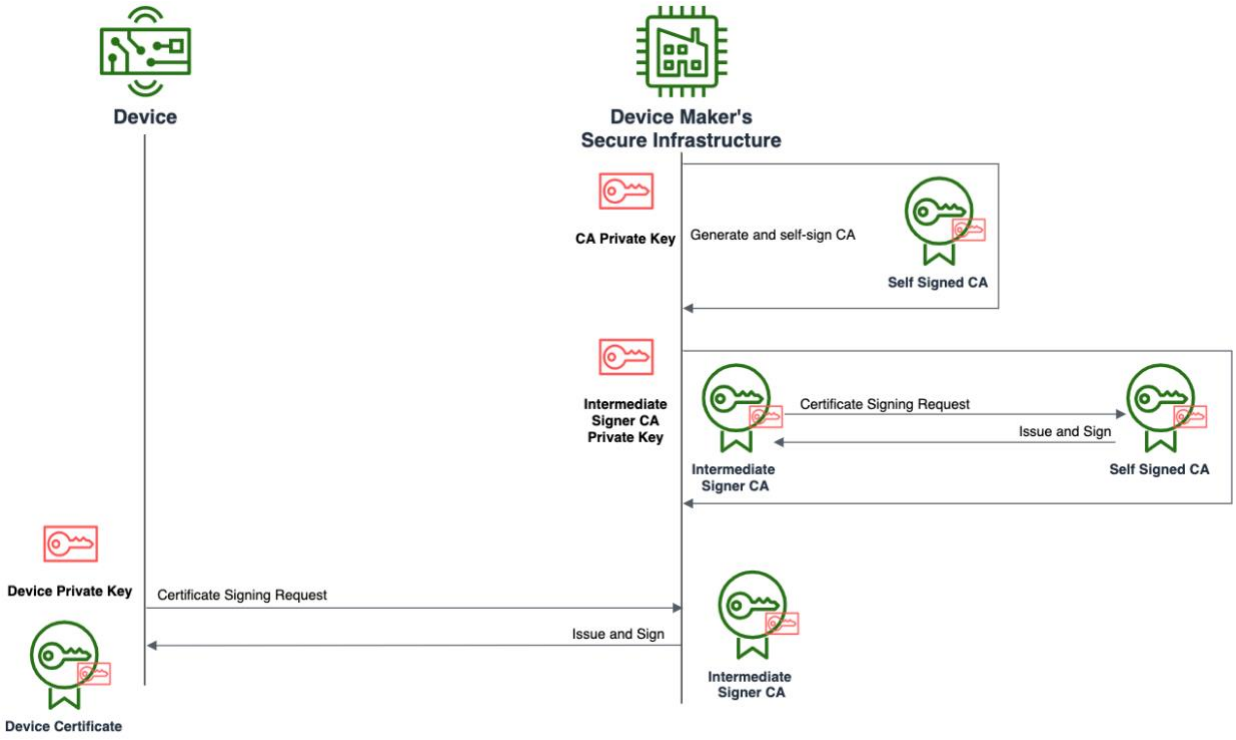

図 *9:* 自己署名認証機関と中間署名者 *CA* インフラストラクチャ

AWS Certificate Manager (ACM) は、クラウドで X.509 証明書を生成して署名できる AWS のマネージドサービスです。ACM の柔軟性により、顧客は独自の CA を持ち込み、AWS で証明書署名操作を実行できます。AWS は、ACM サービス上の CA が保持されている物理インフラストラクチャを保護します。デバイスメーカーには、自分のアカウントで ACM サービスにアクセスできるユーザーに対して適切なポリシーを定める責任があります。

デバイスメーカーが独自の CA を保持しない一方で、自社のアセットの公開鍵インフラストラク

チャを制御したい場合、サードパーティーの CA サービスが利用できます。それらの CA サービス企業は、デバイスメーカーの仕様に合わせてカスタマイズされたデバイスメーカー用の中間署名者 CA を生成するか、独自のルート CA から証明書に署名できます。サードパーティーの CA によって、デバイスメーカーが X.509 証明書を生成して署名できるようになりますが、CA の物理的なセキュリティはサードパーティーが管理します。ハードウェアセキュリティモジュールのベンダーは通常、受託製造業者への出荷前にモジュールを事前にプロビジョニングするためにこのサービスを提供しています。

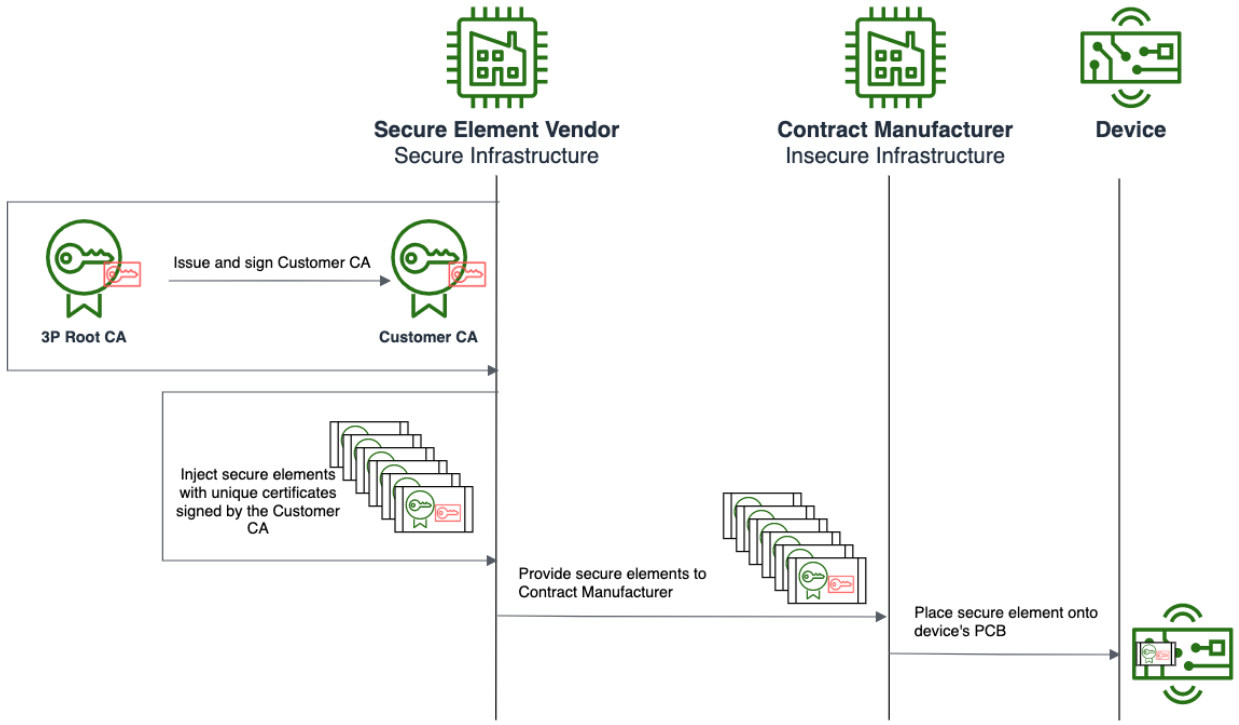

図 *10:* サードパーティー認証機関とハードウェアセキュリティモジュール

AWS IoT プロビジョニングオプション

AWS IoT では、デバイスの機能や、エンドカスタマーに販売される前に一意の X.509 証明書と秘密鍵がデバイスに存在するかどうかに基づいて、多数のデバイスをプロビジョニングおよびオンボーディングするオプションを用意しています。

デバイスメーカーが製造時または流通時にデバイスに一意の認証情報をプロビジョニングすること

が製造チェーンで許可されている場合、デバイスメーカーはジャストインタイムプロビジョニング、ジャストインタイム登録、またはマルチアカウント登録を使用できます。
デバイスがエンドカスタマーに販売される前にデバイスに認証情報を配布できない場合、デバイスメーカーはフリートプロビジョニングを使用してデバイスをオンボードできます。

ジャストインタイムプロビジョニング (JITP)

JITP を使用するデバイスには、AWS IoT にオンボーディングする前に証明書と秘密鍵が存在します。証明書は、顧客の指定した CA で署名されている必要があり、その CA は AWS IoT に登録されている必要があります。顧客は、プロビジョニングする前にデバイスがどのアカウントに接続するかを把握している必要があります。

セットアップ

デバイスは JITP を使用して AWS IoT に接続し、登録された CA に対して証明書の署名が検証されます。検証後、プロビジョニングテンプレートによって Thing と証明書が登録され、デバイスにポリシーが割り当てられます。デバイスメーカーは、署名者 CA を登録し、CA にプロビジョニングテンプレートを添付します。

デバイスロジック

デバイスが AWS IoT Core に初めて接続するとき、デバイス証明書と AWS IoT に登録されている署名者 CA を TLS ハンドシェイク中に送信する必要があります。初回の接続時は TLS ハンドシェイクに失敗します。これは、証明書が AWS IoT アカウントに事前に登録されていないために発生します。デバイスによって提供される証明書は、プロビジョニングプロセス中に AWS IoT に登録され、アクティベートされます。デバイスには、短時間で AWS IoT に再接続するためのロジックが必要です。プロビジョニング操作が成功すると、デバイスは AWS IoT に正常に接続されます。

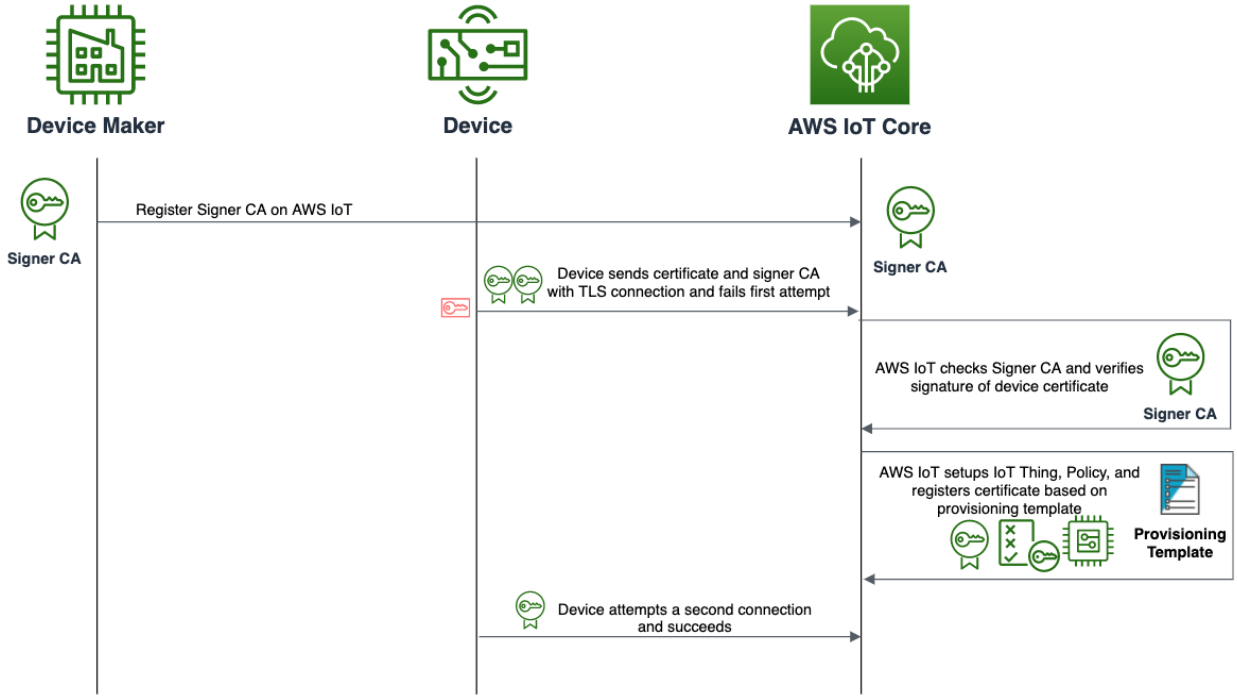

図 *11:* ジャストインタイムプロビジョニングプロセス

ジャストインタイム登録 (JITR)

デバイスを AWS IoT に登録するときに追加のカスタムロジックが必要な場合、JITR を使用する必要があります。JITP と同様、証明書と秘密鍵はオンボーディング前にデバイスに存在する必要があり、デバイスのオンボーディングの前に署名者 CA を AWS アカウントに登録する必要があります。

セットアップ

デバイスが初めて AWS IoT に接続するとき、認証機関に対して証明書の署名が検証され、証明書が非アクティブ状態で登録されます。AWS IoT は、MQTT トピック ‘$aws/events/certificates/registered/<caCertificateID>’ でライフサイクルイベントを生成します。デバイスメーカーは、そのトピックでメッセージが発行されるたびに AWS Lambda 関数をトリガーする IoT ルールをセットアップします。

Lambda 関数は、許可リストまたは証明書失効リストに対するセカンダリ検証、クラウドプラットフォーム上の特定のユーザーへのデバイス登録、追加のオンボーディングワークフローのトリガー

などのアクションを実行できます。Lambda 関数は、追加の検証が完了した後、証明書の状態をアクティブに設定します。さらに、Lambda 関数は IoT Thing も登録し、ポリシーをセットアップする必要があります。

デバイスロジック

JITP と同様、デバイスは AWS IoT への初回の接続時は接続できません。初めて接続が失敗した後 AWS IoT に再接続するロジックがデバイスに含まれている必要があります。Lambda 関数が証明書をアクティベートすると、2 回目のデバイス接続は成功します。

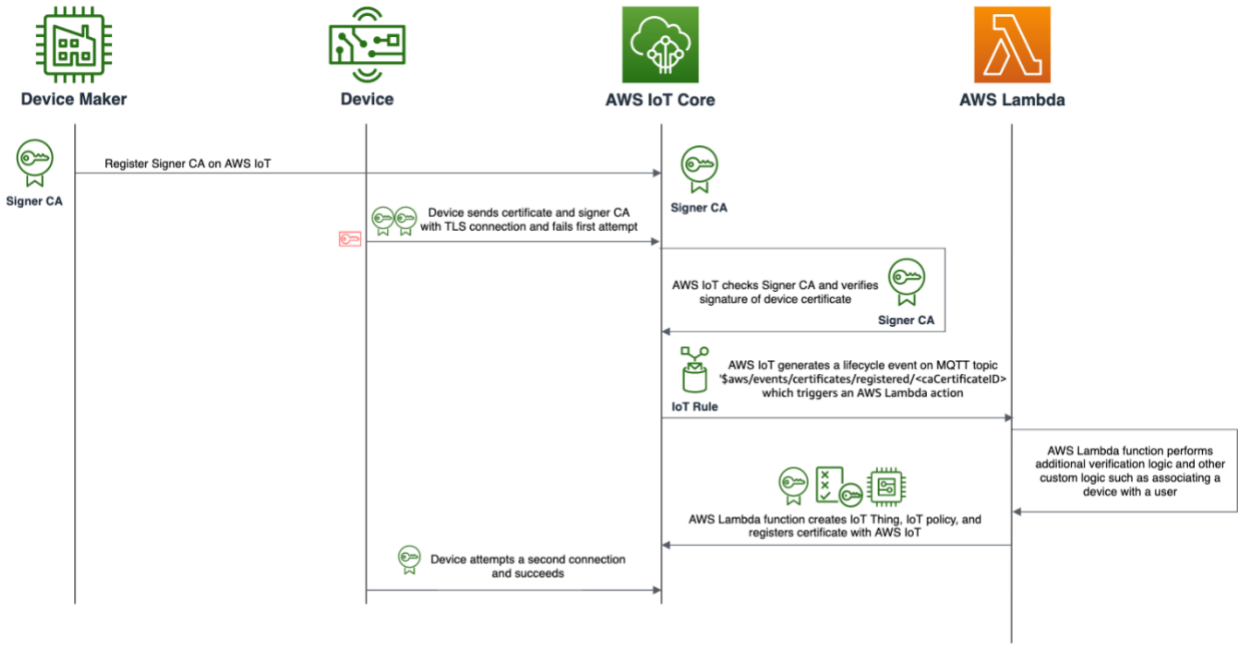

図 *12:* ジャストインタイム登録プロセス

ジャストインタイムプロビジョニングおよび登録のユースケース

ジャストインタイムプロビジョニングとジャストインタイム登録は、デバイスがプロビジョニングされる AWS アカウントとリージョンをデバイスメーカーが製造時に把握している場合に使用されます。デバイスが初回の接続を試みる前に、デバイスがオンボーディングされるアカウントに CA を登録する必要があります。ジャストインタイムプロセスでは 1 回限りのセットアップが必要で、数百万台のデバイスにスケーリングできます。

マルチアカウント登録 (MAR)

MAR を使用して AWS IoT Core にプロビジョニングされるデバイスには、オンボーディング前にデバイスに存在する一意の証明書と秘密鍵が必要です。証明書は CA で署名されますが、その CA を AWS IoT に登録する必要はありません。証明書は、デバイスメーカーがアクセスできるどのリージョンおよびアカウントにも登録できます。

セットアップ

デバイスメーカーは、AWS でデバイスごとに必要なリソースを手動でセットアップする必要があります。リソースは、デバイスが初めて AWS IoT に接続する前にセットアップされます。
一部のベンダーは、秘密鍵と証明書が事前にプロビジョニングされたハードウェアセキュリティモジュールを用意しています。証明書はベンダー独自の CA で署名されており、ベンダーは証明書のマニフェストをデバイスメーカーに提供します。デバイスメーカーには、AWS マネジメントコンソール、AWS IoT コントロールプレーン API、または AWS CLI を使用して、各アカウントおよびリージョンに証明書を登録する責任があります。

デバイスロジック

デバイスの TLS スタックは、サービス名インジケータ (SNI) 拡張をサポートしている必要があり、AWS IoT Core エンドポイントは SNI 文字列で渡されます。MAR プロビジョニングを実行するのに必要な追加のデバイスロジックはありません。

ユースケース

MAR は、デバイスメーカーがデバイスのプロビジョニング先の柔軟性を必要としている場合に使用されます。デバイスメーカーは、サンドボックス、テスト、本番稼働用に複数の AWS アカウントを使用することがあります。証明書は各 AWS アカウントに事前登録でき、デバイスはライフサイクルを通じてさまざまなアカウントに接続できます。デバイスは全世界で販売でき、証明書は複数の AWS リージョンおよび同じリージョン内の複数の AWS アカウントに登録できます。MAR を使用すると、デバイスメーカーは IoT サービスプロバイダーにパブリック X.509 証明書を提供で

きます。これにより、サービスプロバイダーのアカウントにデバイスをオンボーディングしたり、デバイスが複数のアカウントに接続して各エンドポイントにデータを送信したりできるようになります。MAR は、デバイスメーカーが署名認証機関を所有しておらず、CA を AWS IoT に登録できない場合にも使用されます。

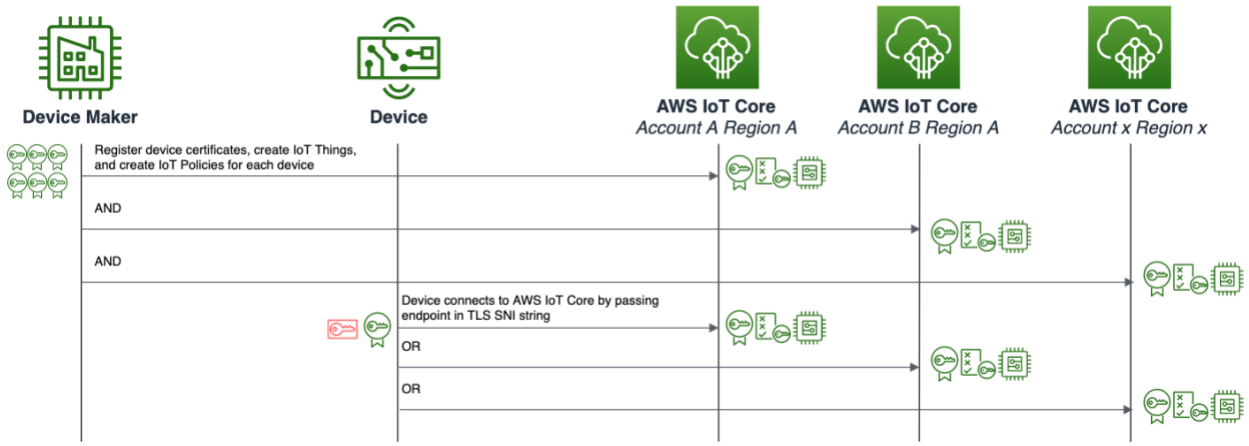

図 *13:* マルチアカウント登録 *(MAR)* プロセス

フリートプロビジョニング

技術的な制限、コスト、アプリケーション固有の制限により、製造時に一意の認証情報をデバイスにプロビジョニングできないケースが多くあります。製造業者がカスタマイズしていないデバイスは、一意の認証情報なしで顧客に販売されます。フリートプロビジョニングには、エンドカスタマーに配布された後に一意の認証情報をデバイスにプロビジョニングする 2 つの方法 (信頼されたユーザーとクレーム) が用意されています。

信頼されたユーザーによるフリートプロビジョニング

AWS IoT には、モバイルアプリケーションが一時的な証明書と秘密鍵を生成できるようにするアプリケーションプログラミングインターフェイス (API) が用意されています。デバイスは、一意の認証情報なしで製造施設から出荷されるため、信頼されたユーザーのみが自身の一意の認証情報を使

用してデバイスをプロビジョニングできます。
インストーラはモバイルアプリケーションを使用して、AWS で認証します。また、信頼されたユーザー API を使用して、5 分間有効な一時的な X.509 証明書と秘密鍵を受け取ります。認証情報は、モバイルアプリケーションを使用してデバイスに配布されます。デバイスは AWS IoT に接続し、一時的な認証情報を、AWS CA によって署名された一意の X.509 証明書と秘密鍵に交換します。このワークフローでは、Thing の名前、ポリシー、証明書などの AWS リソースが AWS アカウントでセットアップされます。

セットアップ

信頼されたユーザーフローを使用するデバイスメーカーは、信頼されたユーザー API を実行するモバイルアプリケーションを開発および管理する必要があります。フリートプロビジョニングテンプレートは、デバイスメーカーが AWS IoT でセットアップおよび管理する必要があります。オプションで、AWS Lambda 関数を使用してプロビジョニングプロセス中に追加の認証ステップを提供することもできます。

デバイスロジック

デバイスは、Bluetooth Low Energy、WiFi、USB などのセキュアな接続を介して一時的な認証情報を受け入れることができる必要があります。デバイスは、フリートプロビジョニング MQTT トピックの発行とサブスクライブに必要なロジックを実装し、永続的な認証情報を受け入れて、セキュアな保存領域に認証情報を書き込む必要があります。

ユースケース

製造サプライチェーンに認証情報が開示されることはありません。高度なセキュリティが必要な場合、製造チェーンが信頼されていない場合、または製造チェーンでデバイスをプロビジョニングできない場合は、信頼されたユーザーによるフリートプロビジョニングが推奨されます。

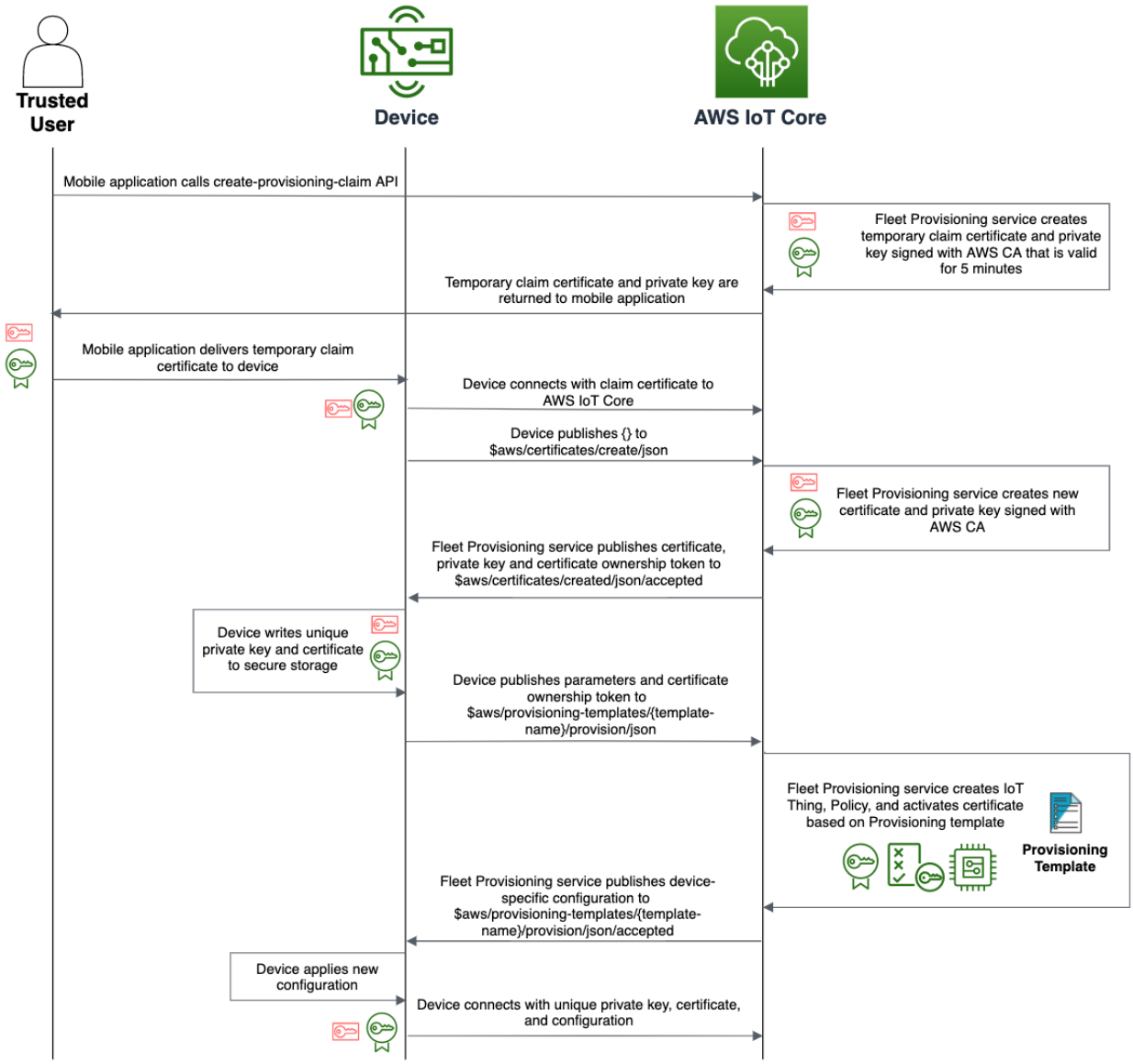

図 *14:* 信頼されたユーザーによるフリートプロビジョニングプロセス

クレームによるフリートプロビジョニング

デバイスによっては、セキュアな転送を介して認証情報を受け入れることができず、製造時にデバイスをカスタマイズする機能が製造サプライチェーンに備わっていないことがあります。AWS IoT には、これらのデバイスがデプロイ時に一意の認証情報を受け取るためのパスが用意されています。

デバイスメーカーは、各デバイスのファームウェアに共有クレーム証明書を含める必要があります。このクレーム証明書は、デバイスのバッチごとに一意である必要があります。クレーム証明書を含むファームウェアは、受託製造業者が書き込みます。カスタマイズを実行する必要はありません。デバイスは、AWS IoT との接続を初めて確立するとき、AWS 認証機関によって署名された一意の X.509 証明書と秘密鍵のクレーム証明書を交換します。また、デバイスは、フリートプロビジョニングサービスが許可リストに対して検証するために使用できるプロビジョニングリクエストとともに、シリアル番号や埋め込みハードウェアシークレットなどの一意のトークンを送信する必要があります。

セットアップ

クレームによるフリートプロビジョニングを使用するデバイスメーカーは、フリートプロビジョニングテンプレートと、追加の検証ロジックを持つ AWS Lambda 関数を管理する必要があります。クレーム証明書は、不正使用を防ぐために保護し、頻繁に監査する必要があります。

デバイスロジック

デバイスは、フリートプロビジョニング MQTT トピックの発行とサブスクライブに必要なロジックを実装し、永続的な認証情報を受け入れて、セキュアな保存領域に認証情報を書き込む必要があります。

ユースケース

クレームによるフリートプロビジョニングは、デバイスで一意の認証情報をプロビジョニングできず、製造サプライチェーンが受託製造業者からエンドカスタマーまで信頼できる個人によって保護されている場合にのみ使用してください。サードパーティーのディストリビューターなどのセキュアでないチャネルを通過するデバイスには、クレームによるフリートプロビジョニングを使用しないでください。

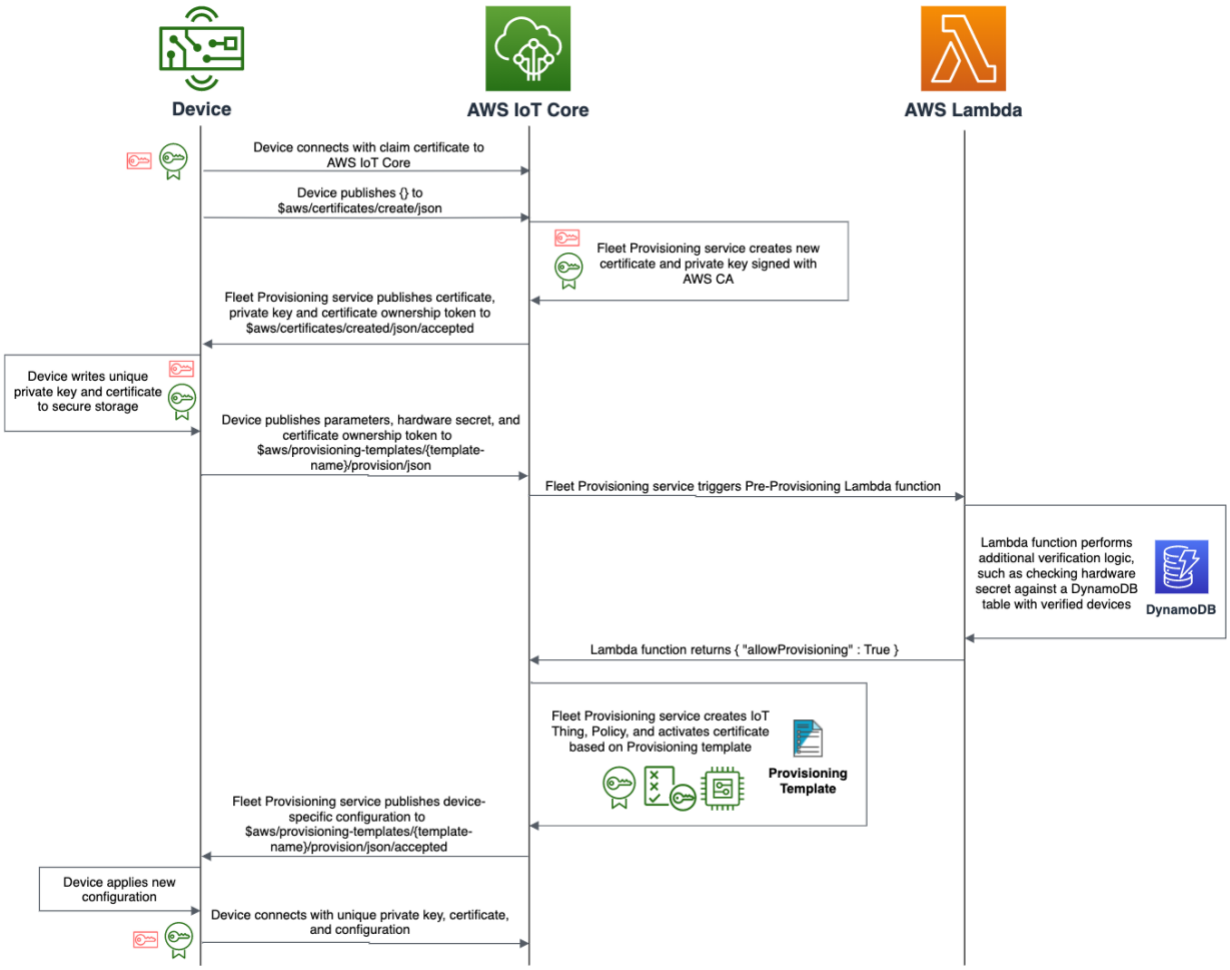

図 *15:* クレームによるフリートプロビジョニングプロセス

まとめ

IoT プロジェクトを概念実証 (PoC) から大規模なデプロイに移行するプロセスは複雑です。デバイスメーカーは、デバイスの機能、セキュリティのレベル、部品表、製品および運用コストに根本的な影響を与える複数の決定を行う必要があります。 製造およびデプロイの決定については、製造とデプロイのコストに影響を与えるため、開発プロセスの早い段階で考慮する必要があります。

デバイスメーカーは、IoT プロジェクトを開始する際に以下の点を考慮する必要があります。

- プロジェクトの公開鍵インフラストラクチャを管理するのは誰ですか?

- どのような機能をデバイスに組み込む必要がありますか?
- 受託製造業者が製造時にカスタマイズする必要があるのはどのような機能ですか?

このホワイトペーパーでは、デバイスメーカーがこれらの質問に答える際に利用できる AWS IoT のオプションについて概要を説明しました。デバイスの機能や製造プロセス、製造サプライチェーンの信頼レベル、セキュリティ要件にかかわらず、AWS IoT では、デバイスメーカーがデバイスを大規模かつセキュアにオンボーディングできるオプションを用意しています。

寄稿者

この文書の寄稿者は次のとおりです。

- IoT、シニアパートナーソリューションアーキテクト、David Walters
- IoT、シニアパートナーソリューションアーキテクト、Michael Schy
- IoT、シニアパートナーソリューションアーキテクト、Ashok Bhaskar
- IoT、プリンシパルパートナーソリューションアーキテクト、Tim Mattison
- AWS IoT Core、シニアプロダクトマネージャー、Alok Jha

ドキュメントの改訂

日付	説明
2021 年 1 月	初版発行