JNSA標準化部会セミナー「IoTセキュリティ標準化の動向を知る」

IETFでも議論される Constrained IoTデバイスの ファームウェアアップデート

セコム　IS研究所
髙山　献（主務研究員）

自己紹介

信頼される安心を、社会へ。
SECOM

- 名前：髙山 献（たかやま けん）
- 経歴
  - 1991年　出生
  - 2019年　東京農工大学大学院修了
    - 専門はOSやVMMなどシステムソフトウェア
  - 2019年　セコム入社　IS研究所 所属
  - 2020年　**機密情報保護**と**データ利活用**を行うための基盤技術を研究
    - キーワード：TEE、Confidential Computing、…
  - 2020年　標準化団体IETFの109回会議出席、以降年3回参加
    - 標準規格の草稿を検証する立場、執筆する立場で参加
    - 主にTEEP, **SUIT**, **COSE**といったセキュリティ系のグループで活動中

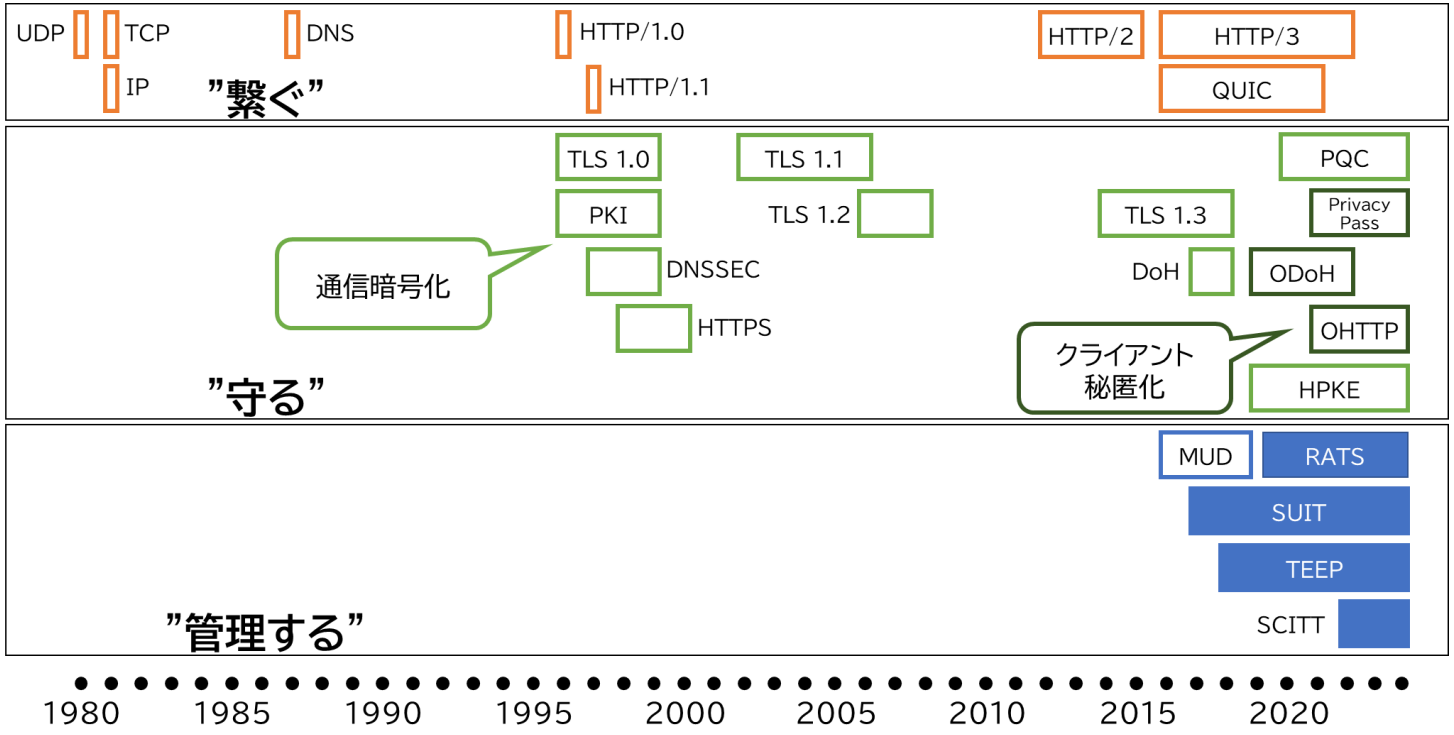
本日の内容

信頼される安心を、社会へ。
SECOM

- IETFにおけるIoTデバイスの位置づけ
  - IETFでは「モノのネットワーク」とも言える位置づけ
  - IETFで標準化されるプロトコルやフォーマットなどは、Internetに限らない
  - Constrained Nodes, Constrained Networks
- Constrained IoTデバイスの安全なファームウェアアップデート
  - SUIT: アップデート指示を安全に伝達する
  - JC-STARのセキュリティ要件との対応付け
  - COSE: セキュリティを担保するための暗号機能を提供する
  - CBOR: データサイズをコンパクトにする

IETFとは

- Internet Engineering Task Force
  - インターネットに関する技術標準を策定するボランティア団体
  - 標準ドキュメントは「RFC XXXX」とナンバリングされて発行される
  - 技術名や、その標準を策定したグループ名が通称として残る場合がある

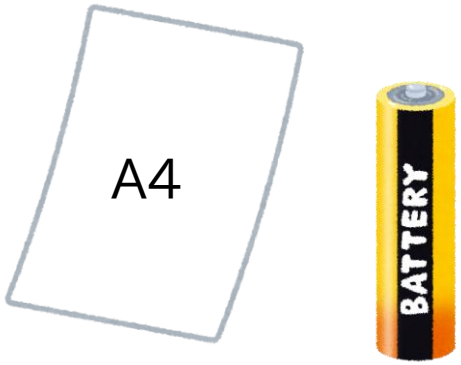


発表者は
このあたりの標準化に
携わっています

IETFの標準化対象とその広がり（抜粋）

標準化するとは

- 標準化する ＝ 良いものの基準をみんなで確認すること
  - 作る人や使う人が、どういったものが良いものかを合意する「ものさし」を作る
  - 言葉を定義することもある
- 標準化される過程は様々
  - デジュール標準　法律や条約に基づく団体が合議して標準を作る
  - フォーラム標準　業界団体などで協議して標準を作る ← IETFが該当
  - デファクト標準　市場の淘汰によって生き残ったものが標準になる



デジュール標準によるもの

フォーラム標準によるもの

デファクト標準によるもの

IETFにおけるIoTとは

- The Internet of Things
  - The Internet of Things (IoT) is **the network of physical objects or ”things”** embedded with electronics, software, sensors, actuators, and connectivity
  - to enable objects to exchange data with the manufacturer, operator, and/or other connected devices.

https://www.ietf.org/technologies/iot/

- モノのインターネットとは（Copilot訳）
  - 電子機器、ソフトウェア、センサー、アクチュエーター、および接続機能を組み込んだ**物理的なオブジェクトや「モノ」のネットワークのこと**です。
  - これにより、オブジェクトは製造者、運用者、および/または他の接続されたデバイスとデータを交換することができます。

IETFにおけるIoTとは

- The Internet of Things

IETFにおけるIoTは、インターネットへの接続機能を持つ
（Internet Protocolに対応する）モノに限定されていません。
デバイスやネットワークの性能が低い、大量にあり管理が大変になる、
といったIoT特有の課題を解決できる標準技術の策定が続いています。

- モノのインターネットとは（Copilot訳）
  - 電子機器、ソフトウェア、センサー、アクチュエーター、および接続機能を組み込んだ**物理的なオブジェクトや「モノ」のネットワークのこと**です。
  - これにより、オブジェクトは製造者、運用者、および/または他の接続されたデバイスとデータを交換することができます。

RFC 7228: Constrained〇〇の用語整理

信頼される安心を、社会へ。 SECOM

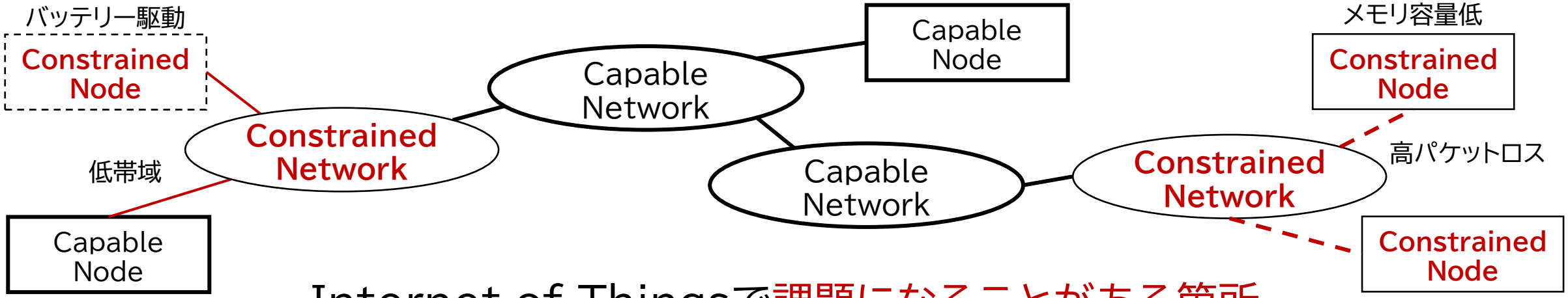
- Constrained Node（制約のあるノード）
  - データの処理・格納・伝送性能が低い、UIが限定的、…
- **Constrained Device**（制約のあるデバイス）
  - Nodeのうち特にデバイス単体の性能に着目する場合
- Constrained Network（制約のあるネットワーク）
  - 低帯域、高パケットロス、…

分類	アップデート可否
F0	不可、廃棄のみ
F1	可、上書き、要再起動
F2	可、実行中に適用可能、要再起動

（例）アップデートに着目した分類

分類	RAMサイズ	FLASHサイズ
Class 0	<< 10 KiB	<< 100 KiB
Class 1	10 KiB程度	100 KiB程度
Class 2	50 KiB程度	250 KiB程度

（例）メモリサイズに着目した分類



Internet of Thingsで課題になることがある箇所

RFC 7228: Constrained〇〇の用語整理

信頼される安心を、社会へ。 SECOM

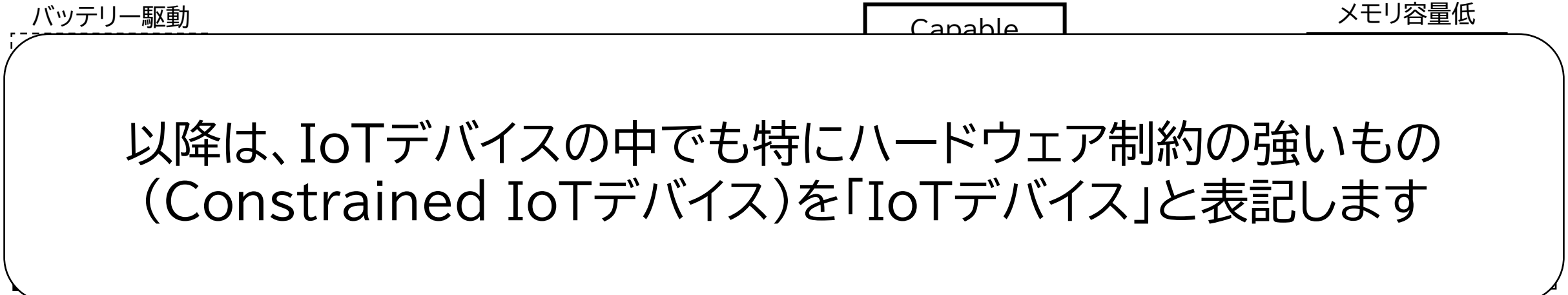
- Constrained Node（制約のあるノード）
  - データの処理・格納・伝送性能が低い、UIが限定的、…
- **Constrained Device**（制約のあるデバイス）
  - Nodeのうち特にデバイス単体の性能に着目する場合
- Constrained Network（制約のあるネットワーク）
  - 低帯域、高パケットロス、…

分類	アップデート可否
F0	不可、廃棄のみ
F1	可、上書き、要再起動
F2	可、実行中に適用可能、要再起動

（例）アップデートに着目した分類

分類	RAMサイズ	FLASHサイズ
Class 0	<< 10 KiB	<< 100 KiB
Class 1	10 KiB程度	100 KiB程度
Class 2	50 KiB程度	250 KiB程度

（例）メモリサイズに着目した分類



IoTが深く絡む標準は多数（抜粋）

信頼される安心を、社会へ。
SECOM

- IoTデバイスは大量にあり、人間が直接操作しないこともある
  - ➢自動化したい
- IoTデバイスやネットワークは性能が低い場合がある
  - ➢データサイズはコンパクトにしたい、実装すべき機能を省きたい（拡張は可能）

名称		概要	類似標準	自動化	データサイズ小	省機能
MUD	Manufacturer Usage Description	デバイスによるネットワークへの認可要求		✓	✓	✓
ACE	Authentication and Authorization for Constrained Environments	デバイスからの認証・認可プロトコル		✓	✓	
SUIT	Software Updates for Internet of Things	ファームウェアアップデートの指示		✓	✓	✓
COSE	CBOR Object Signing and Encryption	暗号メッセージのCBORへの格納方法	JOSE		✓	✓
CBOR	Concise Binary Object Representation	データ構造のエンコード方法	JSON		✓	
C509	CBOR Encoded X.509 Certificates	PKI証明書のCBORへの格納方法	X.509		✓	✓
CoAP	Constrained Application Protocol	メッセージを送受信するプロトコル	HTTP		✓	✓
CoRE	Constrained RESTful Environments	制約のあるネットワーク上での各アプリ層			✓	✓

解決対象であるConstrainedやIoTを含む略語名称を持つもの、解決した結果の簡潔を意味するConciseを含むものも多い

概要
要件
仕様
実装

IETF SUITのご紹介

SUITワーキンググループ（2017～）

信頼される安心を、社会へ。SECOM

2016年8月に
IoTデバイスを狙う
マルウェアMirai発見

- Software Updates for Internet of Things
  - 背景：IoTデバイスの脆弱性が明らかになり、
    セキュアなファームウェアアップデートの必要性が高まった
  - 課題：IoTデバイスを対象とする既存のアップデート手法は相互運用性が低い
  - 目的：IoT全体の安全性を高める
  - 手段：アップデート技術について相互運用可能な技術標準を作る

相互運用性（interoperability）とは…
異なるシステムや組織が**連携**し、
情報や**機能を共有**できる特性のこと
・ブラウザとWebサーバー
・IoTデバイスとファームウェアサーバー
（あるいはその開発組織）など

分類	RAMサイズ	FLASHサイズ
Class 0	<< 10 KiB	<< 100 KiB
Class 1	10 KiB程度	100 KiB程度
Class 2	50 KiB程度	250 KiB程度
（これよりも高性能なもの）		

SUIT WGが主な対象とする分類（RFC 7228より、再掲）

SUITグループが標準化してきたもの

信頼される安心を、社会へ。 SECOM

📣Informational（情報提供目的）
📖Standard（IETFで合意された標準）

- （用語定義）📣RFC 9019: SUIT Architecture
  - SUIT関連の用語にあいまいさを無くすための定義
    - 例）Deviceとは、Firmwareとは、Authorとは、Manifestとは、…
- （要件定義）📣RFC 9124: SUIT Information Model
  - ファームウェアアップデートをセキュアに行うために考慮すべき脅威
  - それらからIoTデバイスのアップデートを守るために要求されること
    - 例）アップデート指示の**真正性や完全性**、格納したデータが改ざんされないこと、…

  ➢SUIT以外でも参考になる情報
- （仕様書）📖SUIT Manifest
  - アップデート指示を格納する具体的なデータフォーマットのひとつ
  - エンコーディングに**小サイズのCBOR**、**真正性や完全性検証にCOSE**を採用

  ➢プログラムを書いて、標準化内容の妥当性を検証することができる段階

LwM2MやMatterなど、IoTを対象にアップデート機能を標準は他にもあります

（比較）JC-STAR

（RFC 9124相当） 要件

信頼される安心を、社会へ。 SECOM

IETFのIoTデバイスより
認証対象の範囲は狭い

- セキュリティ要件適合評価及びラベリング制度
  - IPAが運用する制度、経済産業省が取りまとめたものがベース
  - **インターネットプロトコル(IP)を使用したデータの送受信機能を持つもの**が対象
- セキュリティ要件カテゴリ（抜粋）

1. 脆弱な認証・認可メカニズム(例：汎用のデフォルトパスワード、脆弱なパスワード)を使用しない
2. 脆弱性の報告を管理するための手段を導入する
3. **ソフトウェアを最新の状態に保つ**
4. 機密セキュリティパラメータをセキュアに保存する
5. セキュアに通信する
6. …

厳しさに応じて★1～4の
レベルがある(図は★1)

https://www.ipa.go.jp/security/jc-star/tekigou-kizyun-guide/label1/begoj90000004zgc-att/star1_requirements.pdf

（比較）JC-STAR

（RFC 9124相当） 要件

信頼される安心を、社会へ。 SECOM

IETFのIoTデバイスより認証対象の範囲は狭い

- セキュリティ要件適合評価及びラベリング制度
  - IPAが運用する制度、経済産業省が取りまとめたものがベース
  - **インターネットプロトコル(IP)を使用したデータの送受信機能を持つもの**が対象
- セキュリティ要件カテゴリ（抜粋）

1. 脆弱な認証・認可メカニズム(例：汎用のデフォルトパスワード、脆弱なパスワード)を使用しない
2. 脆弱性の報告を管理するための手段を導入する
3. **ソフトウェアを最新の状態に保つ** ← IETF SUITの対象
4. 機密セキュリティパラメータをセキュアに保存する
5. セキュアに通信する
6. …

IETFの各種通信プロトコルの対象（例)HTTPS

厳しさに応じて★1～4のレベルがある(図は★1)

https://www.ipa.go.jp/security/jc-star/tekigou-kizyun-guide/label1/begoj90000004zgc-att/star1_requirements.pdf

SUITが解決できる範囲

要件

信頼される安心を、社会へ。
SECOM

3. ソフトウェアを最新の状態に保つ（セキュリティ要件を抜粋）
   1. ソフトウェアコンポーネントについて、アップデート可能にしなければならない
   2. 制約のある機器でない場合、**アップデートをセキュアにインストールするためのアップデートメカニズム**
   3. そのアップデートは、**ユーザが簡単に適用できる**
   7. ベストプラクティスの暗号技術を使用
   8. セキュリティアップデートは、適時でなければならない
   10. ソフトウェアアップデートがネットワークインタフェースを介して配信される場合、製品は、**信頼関係を介して各アップデートの真正性及び完全性を検証**

この2項目は最低限の★1には入っていない

SUITは、制約のある機器でもセキュア（**真正性・完全性**）なアップデートを可能にする技術要件を定め、ベストプラクティスの暗号技術を選べるようプロファイルを提示している。

https://www.ipa.go.jp/security/jc-star/tekigou-kizyun-guide/label1/begoj90000004zgc-att/star1_requirements.pdf

アップデートメカニズム

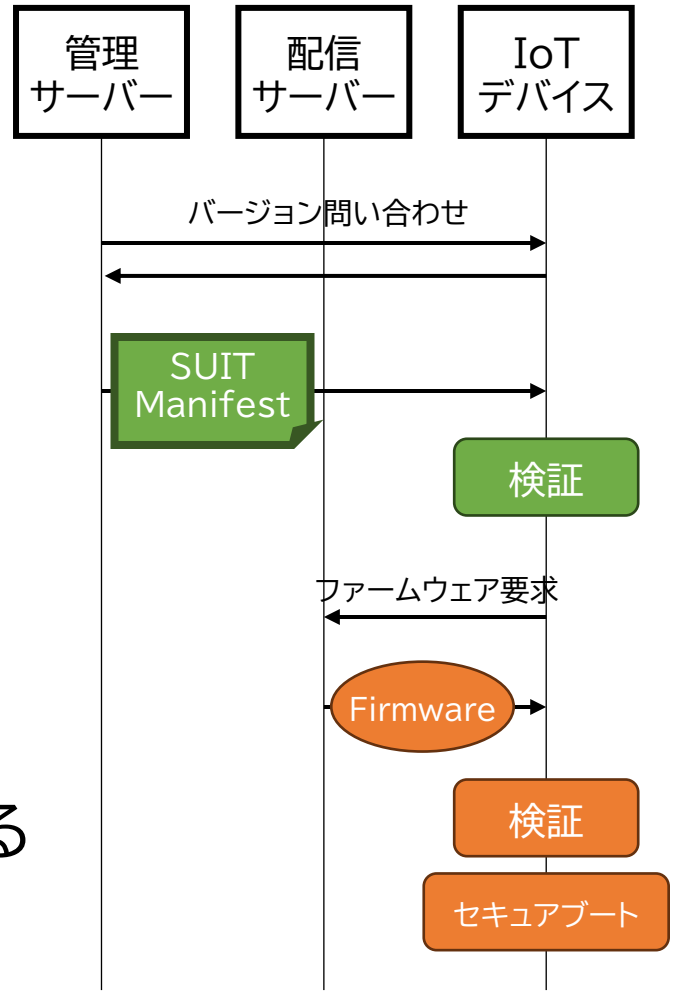
要件

信頼される安心を、社会へ。 SECOM

3.2. アップデートをセキュアにインストールするための アップデートメカニズム

3.3. ユーザが簡単に適用できる

- RFC 9019 SUIT Architecture
  - IoTデバイスはSUIT Manifestを**検証**してから実行し
  - ファームウェアを**検証**し
  - **セキュアブート**を行う
  - （自動化が可能で、人間の操作を必ずしも必要としない）
- RFC 9124 SUIT Information Model
  - ファームウェアは**真正**なサイズや**ハッシュ値**によって検証する



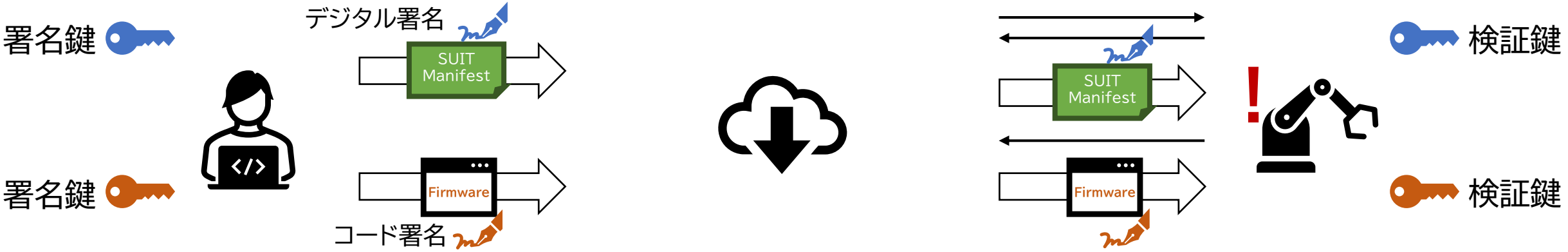
SUIT Manifest

要件

信頼される安心を、社会へ。 SECOM

3.10. 信頼関係を介して各アップデートの真正性及び完全性を検証しなければならない

- SUIT Manifestの検証に必要な、信頼関係の構築方法
  - デバイスは事前に、SUIT Manifestの真正性検証のための暗号鍵を持っておく
  - SUIT Manifestのハッシュ値とデジタル署名/MACを検証する
  - ➢そのSUIT Manifestは信頼する作者が作り、改ざんが無いことが分かる



COSE

3.7. ベストプラクティスの暗号技術を使用

- CBOR Object Signing and Encryption
  - 任意のデータに対して真正性(なりすましが無いこと)、完全性(改ざんが無いこと)、秘匿性(秘密を隠したままにできること)を保証するための機能を提供
  - 同じくIETFが標準化する軽量なバイナリエンコーディング方式CBORを使う
  - デジタル署名、メッセージ認証コード (MAC)、暗号化で実現

暗号技術	cryptography	真正性	完全性	秘匿性
ハッシュ	hash		✓	
デジタル署名	signature	✓	✓	
メッセージ認証コード	MAC	✓	✓	
認証付き暗号	AEAD	✓	✓	✓
暗号	encryption, non-AEAD			✓

SUIT Mandatory-to-Implement

仕様

信頼される安心を、社会へ。 SECOM

3.7. ベストプラクティスの暗号技術を使用

- 暗号アルゴリズムの組み合わせのベストプラクティス
  - ハードウェアの制約の中で十分な強度を持つものを選択

プロファイル名	要求性能	真正性	完全性	鍵共有	秘匿性
Symmetric	最低	HMAC-256	SHA-256	A128KW	A128CTR
Constrained Asymmetric 1	低	ES256	SHA-256	ECDH-ES+A128KW	A128CTR
Constrained Asymmetric 2	低	EdDSA	SHA-256	ECDH-ES+A128KW	A128CTR
AEAD Asymmetric 1	中	ES256	SHA-256	ECDH-ES+A128KW	A128GCM
AEAD Asymmetric 2	中	EdDSA	SHA-256	ECDH-ES+A128KW	A128GCM
Future Constrained 1	高	HSS-LMS*	SHA-256	A256KW	A256CTR

*将来的に実現される量子コンピュータによって真正性を破ることが難しいとされるアルゴリズム（PQC: Post-Quantum Cryptography）

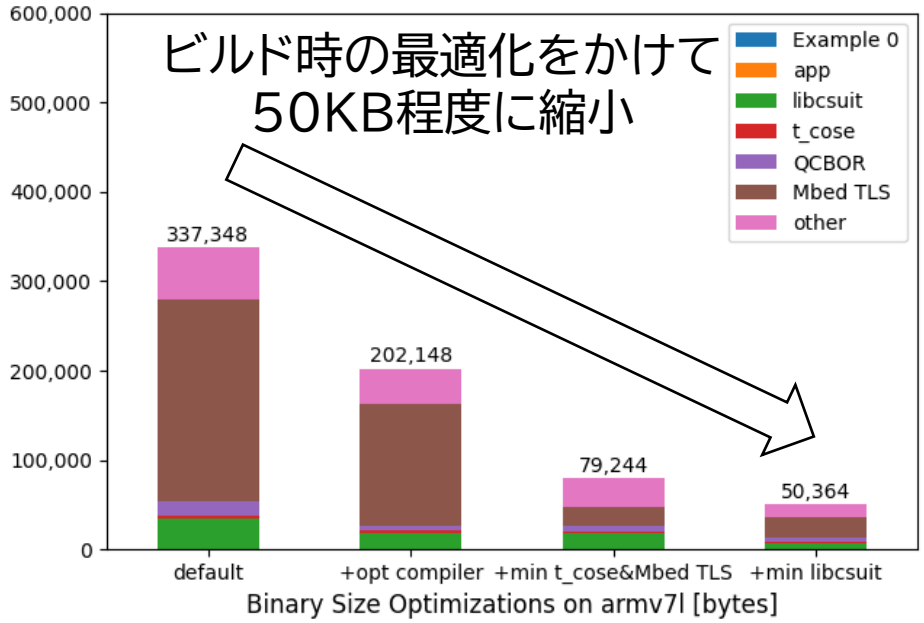
IETF文化:Running Code

実装

信頼される安心を、社会へ。 SECOM

- 標準化＝標準ドキュメントを書く（＋プログラムを書いて確かめる）
- （実装）libcsuit（https://github.com/kentakayama/libcsuit）
  - セコムがオープンソースソフトウェア（OSS）として公開
  - SUIT Manifestのパース処理を行うC言語のライブラリ
  - CBOR、COSE、暗号機能は既存のOSSライブラリを使う

FLASH100KiBのClass 1デバイスでは
プログラムサイズ50KBは大きく、要改善。
暗号ライブラリ**Mbed TLS**は多機能で20KBと大きい。
ソースコードからいじって不要な機能を削ぎ落せば
30KB程度までは小さくできる見込み。



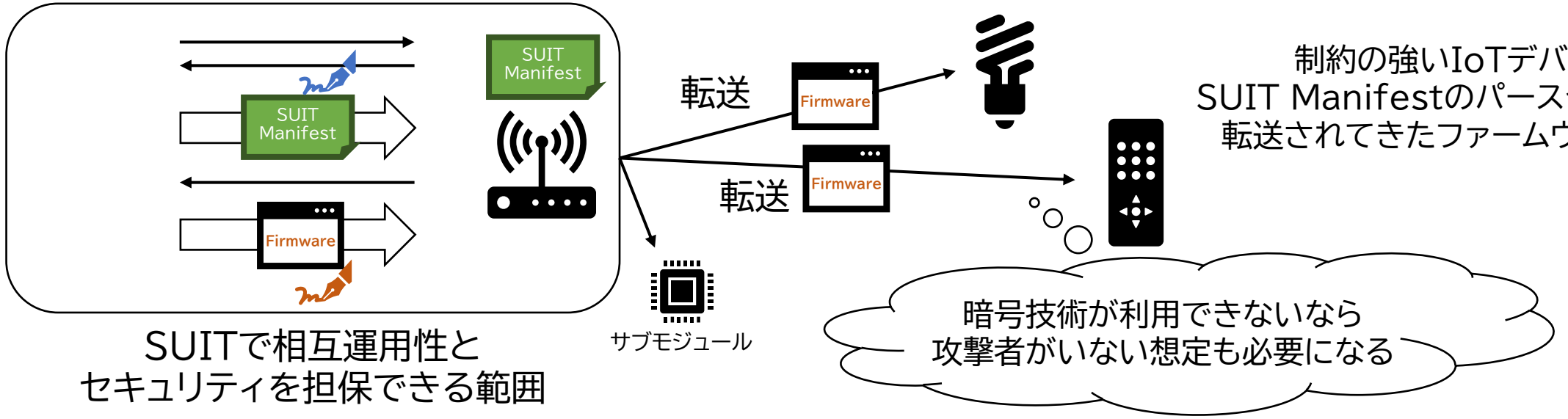
libcsuitはClass 0でも動くのか？

実装

信頼される安心を、社会へ。 SECOM

- 動くがかなり厳しい
  - たとえ30KBまで削れても、
    アップデート機能が主機能を圧迫しかねない
- 制約の弱いデバイスでlibcsuitを動かし
  真正性・完全性を検証したファームウェアを転送
  - そのような管理デバイスでは**相互運用性が求められる**

分類	RAMサイズ	FLASHサイズ
Class 0	<< 10 KiB	<< 100 KiB
Class 1	10 KiB程度	100 KiB程度
Class 2	50 KiB程度	250 KiB程度



SUITで相互運用性と
セキュリティを担保できる範囲

まとめ

- IETFとは
- Constrained IoTデバイスの課題
  - 自動化できる、データサイズはコンパクト、最低限必要な機能は少ない
- SUITが解決できる、JC-STARのセキュリティ要件
  - 3.1. 制約のある機器でない場合、アップデートをセキュアにインストールするためのアップデートメカニズム
  - 3.3. そのアップデートは、ユーザが簡単に適用できる
  - 3.7. ベストプラクティスの暗号技術を使用
  - 3.10. ソフトウェアアップデートがネットワークインタフェースを介して配信される場合、製品は、信頼関係を介して各アップデートの真正性及び完全性を検証
- SUITが利用可能な範囲
  - 理想はあらゆるIoTデバイス、他社連携・委任する場合は相互運用性が重要