Lignes directrices de sécurité IoT pour l'écosystème de dispositifs périphériques IoT

Lignes directrices de sécurité IoT pour l'écosystème de dispositifs périphériques IoT

Version 2.0

26 Octobre 2017

Ce document est une référence permanente non contraignante de la GSMA

Classification de sécurité : Non-confidentiel

Copyright

Avertissement

Avis antitrust

Table de Matières

1 Introduction

1.1 Introduction à l'ensemble de documents sur les lignes directrices de sécurité IoT de la GSMA

Ce document fait partie d'un ensemble de documents de lignes directrices de sécurité de la GSMA destinés à aider l'industrie de l'Internet des Objets (IoT) naissante à établir une compréhension commune des problèmes de sécurité de l'IoT. L'ensemble de documents non contraignants promeut la méthodologie pour développer des services IoT sécurisés afin de faciliter la mise en œuvre des meilleures pratiques de sécurité tout au long du cycle de vie du service. Les documents fournissent des recommandations sur la façon de diminuer les menaces et les faiblesses courantes en matière de sécurité au sein des services IoT.

La structure du jeu de documents de lignes directrices de sécurité de la GSMA est présentée ci-dessous. Il est recommandé de lire le document de synthèse «CLP.11 Aperçu des lignes directrices de sécurité IoT» [1] avant de lire les autres documents plus détaillés CLP.12 [2] et CLP.13 [3] (ce document).

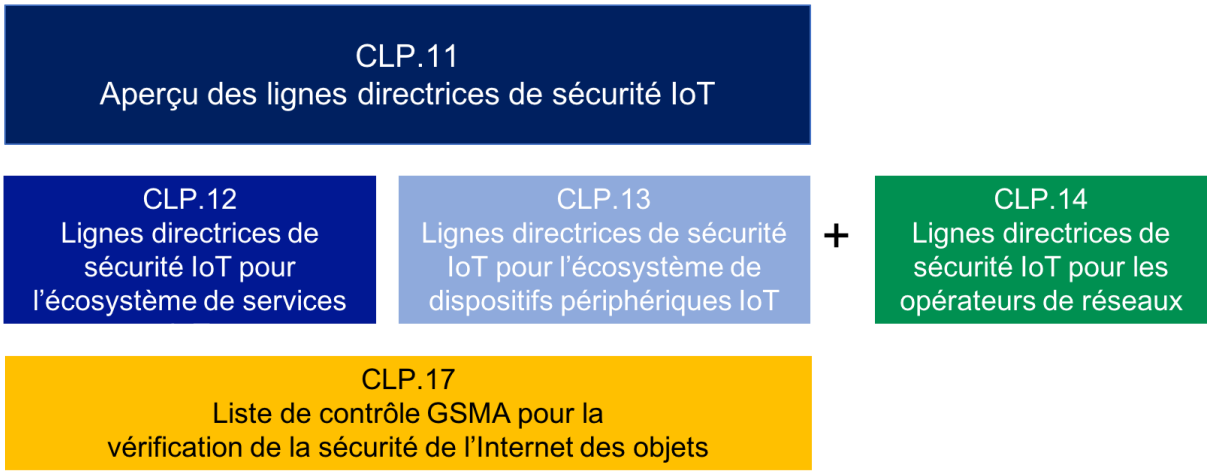

Figure 1 - Structure des Documents sur les Directives de Sécurité de la GSMA

Les opérateurs de réseau, les fournisseurs de services IoT et les autres partenaires de l'écosystème IoT sont invités à lire le document GSMA CLP.14 "Lignes directrices de sécurité IoT pour les opérateurs de réseau" [4] qui fournit des lignes directrices générales de sécurité aux opérateurs de réseau qui veulent être au même temps fournisseurs de services IoT pour assurer la sécurité du système et la confidentialité des données.

1.1.1 Liste de contrôle pour l'évaluation de la sécurité IoT de la GSMA

Une liste de contrôle d'évaluation est fournie dans le document CLP.17 [19]. Ce document permet aux fournisseurs de produits, services et composants IoT d'autoévaluer la conformité de leurs produits, services et composants par rapport aux lignes directrices de sécurité IoT de la GSMA.

L'achèvement d'une liste de contrôle d'évaluation de la sécurité de l'IoT de la GSMA [19] permettra à une entité de démontrer les mesures de sécurité qu'elle a prises pour protéger ses produits, services et composants contre les risques de cyber sécurité.

Les déclarations d'évaluation peuvent être faites en soumettant une déclaration remplie à la GSMA. Veuillez consulter le processus sur le site Web suivant de la GSMA :

https://www.gsma.com/iot/future-iot-networks/iot-security-guidelines/

1.2 Objectif du document

Ce document doit être utilisé pour évaluer les composants d'un service IoT à partir de la perspective des dispositifs périphériques IoT. Ce type de dispositif, du point de vue de l'Internet des Objets, est un périphérique informatique physique qui exécute une fonction ou une tâche dans le cadre d'un produit ou d'un service connecté à Internet. Un dispositif périphérique, par exemple, pourrait être un dispositif de fitness portable, un système de contrôle industriel, une unité de télématique automobile ou même un drone personnel. Toutes les technologies utilisées pour contrôler le dispositif périphérique doivent être évaluées en termes de risques de sécurité. Le résultat est un ensemble pratique de directrices pour la conception du dispositif qui permettent au lecteur d'identifier et de corriger presque tous les risques potentiels pour le service IoT.

La portée de ce document se limite aux recommandations relatives à la conception et à la mise en œuvre des dispositifs périphériques IoT.

Ce document ne vise pas à créer de nouvelles spécifications ou normes IoT, mais fera référence aux solutions, normes et bonnes pratiques actuellement disponibles.

Ce document n'a pas pour but d'accélérer l'obsolescence des services IoT existants. La rétrocompatibilité avec les services IoT existants de l'opérateur de réseau doit être maintenue lorsqu'ils sont considérés comme correctement sécurisés.

Il est noté que le respect des lois et règlements nationaux pour un territoire particulier peut, si nécessaire, annuler les lignes directrices énoncées dans ce document.

1.3 Public visé

Le principal public visé par ce document est :

- Les fournisseurs de services IoT - entreprises ou organisations qui cherchent à développer de nouveaux produits et services connectés innovateurs. Parmi les nombreux domaines dans lesquels les fournisseurs de services IoT opèrent, figurent les maisons intelligentes, les villes intelligentes, l'automobile, le transport, la santé, les services publics et l'électronique grand public.
- Les fabricants de dispositifs IoT - Fournisseurs de dispositifs IoT aux fournisseurs de services IoT pour activer les services IoT.
- Les développeurs IoT - créent des services IoT pour le compte des fournisseurs de services IoT.
- Les opérateurs de réseau qui sont eux-mêmes des fournisseurs de services IoT ou qui construisent des services IoT au nom des fournisseurs de services IoT.

1.4 Définitions

Terme	Description
Nom du point d'accès réseau	Identifiant d'un point de connexion au réseau, auquel un dispositif périphérique se rattache. Ils sont associés à différents types de services et, dans de nombreux cas, sont configurés par l'opérateur de réseau.
Attaquant	Un pirate informatique, un agent de menace, un acteur de la menace, un fraudeur ou toute autre menace malveillante envers un service IoT généralement dans le but de récupérer, détruire, restreindre ou falsifier des informations. Cette menace pourrait provenir d'un criminel, du crime organisé, du terrorisme, de gouvernements hostiles et de leurs agences, d'espionnage industriel, de groupes de piratage, de militants politiques, de pirates informatiques, de chercheurs, ainsi que d'atteintes involontaires à la sécurité et à la vie privée.
Cellulaire	Toute technologie de réseau mobile normalisée 3GPP (par exemple GSM, UMTS, LTE (LTE-M Inc.) Et NB-IoT).
Cloud	Un réseau de serveurs distants sur Internet qui hébergent, stockent, gèrent et traitent les applications et leurs données.
Dispositif Périphérique Complexe	Ce modèle de Dispositif Périphérique dispose d'une connexion permanente à un serveur principal via une liaison de communications par réseau étendu, telle qu'une connexion cellulaire, par satellite ou câblée telle qu'Ethernet. Voir la section 3.
Carte de SIM embarquée	Une carte SIM qui n'est pas destinée à être retirée ou remplacée dans un appareil, et qui permet la gestion sécurisée des profils selon la norme GSMA SGP.01 [2].
Dispositif Périphérique	Un dispositif périphérique IoT est un périphérique informatique physique qui exécute une fonction ou une tâche dans le cadre d'un produit ou d'un service connecté à Internet. Voir la section 3 pour une description des trois classes communes de périphériques IoT et des exemples de chaque classe de dispositif péreiphérique.
Internet des Objets	L'Internet des objets (IoT) décrit la coordination de plusieurs machines, appareils et appareils connectés à Internet via plusieurs réseaux. Ces dispositifs comprennent des objets du quotidien tels que les tablettes et l'électronique grand public, ainsi que d'autres machines telles que des véhicules, des moniteurs et des capteurs dotés de capacités de communication leur permettant d'envoyer et de recevoir des données.
Service IoT	Tout programme informatique qui tire parti des données des périphériques IoT pour rendre un service.
Écosystème de Services IoT	Ensemble de services, plates-formes, protocoles et autres technologies requis pour fournir des fonctionnalités et collecter des données à partir des dispositifs périphériques déployés sur le terrain. Voir CLP.11[1] pour plus d'information.
Fournisseurs de Service IoT	Entreprises ou organisations qui cherchent à développer de nouveaux produits et services connectés innovants.
Opérateur de réseau	L'opérateur et le propriétaire du réseau de communication qui connecte un dispositif périphérique IoT à l'écosystème de service IoT.

Terme	Description
Racine organisationnelle de la confiance	Un ensemble de politiques et de procédures cryptographiques qui régissent la façon dont les identités, les applications et les communications peuvent et doivent être sécurisées par cryptographie.
Point d'accès au service	Un point d'entrée dans l'infrastructure back-end d'un service IoT via un réseau de communication.
Module d'identité d'abonné (SIM)	La carte à puce utilisée par un réseau mobile pour authentifier les dispositifs de connexion au réseau mobile et d'accès aux services de réseau.
Ancre de Confiance	Dans les systèmes cryptographiques à structure hiérarchique, une ancre de confiance est une entité que représente l'autorité pour laquelle la confiance est supposée et non dérivée.
Base informatique sécurisée	Une Base informatique Sécurisée (TCB) est un regroupement d'algorithmes, de politiques et de secrets au sein d'un produit ou d'un service. La TCB agit comme un module qui permet au produit ou au service de mesurer sa propre fiabilité, d'évaluer l'authenticité des homologues du réseau, de vérifier l'intégrité des messages transmis ou reçue par le produit ou le service, et plus encore. La TCB fonctionne comme la plate-forme de sécurité de base sur laquelle des produits et services sécurisés peuvent être construits. Les composants d'une TCB changeront en fonction du contexte (une TCB matériel (HW) pour un dispositif périphérique ou une TCB logiciel (SW) pour les services cloud), mais les objectifs abstraits, les services, les procédures et les stratégies devraient être très similaires.
Environnement d'exécution approuvé (TEE)	Un environnement qui fonctionne auprès d'un système d'exploitation riche et fournit des services de sécurité à ce système d'exploitation. Il existe plusieurs technologies qui peuvent être utilisées pour mettre en œuvre une TEE, et le niveau de sécurité atteint varie en conséquence.
UICC	Plate-forme d'élément sécurisé spécifiée dans la norme ETSI TS 102 221 et pouvant prendre en charge plusieurs applications d'authentification de réseau ou de service normalisées dans des domaines de sécurité crypto graphiquement séparés. Il peut être incorporé dans des facteurs de forme incorporés spécifiés dans la norme ETSI TS 102 671.

1.1 Abréviations

Terme	Description
3GPP	Projet de Partenariat sur la Troisième Génération (« 3rd Generation Project Partnership »)
AC	Courant en Alterne (« Alternating Current »)
API	Interface de Programmation d'Applications (« Application Program Interface »)
APN	Nom du Point d'Accès (« Access Point Name »)
BLE	Bluetooth á basse consommation d'énergie (« Bluetooth Low Energy »)
BT	Bluetooth
CLP	Programme de la Vie Connectée (« GSMA's Connected Living Programme »)
CPE	Équipement des locaux du client (« Customer Premises Equipment »)
CPU	Unité centrale de traitement (« Central processing Unit »)

Terme	Description
EEPROM	Mémoire Morte Effaçable Électriquement et Programmable (« Electrically Erasable Programmable Read-Only Memory »)
eUICC	UICC Embarquée (« Embedded UICC »)
FIB	Faisceau d'ions focalisé (« Focused Ion Beam »)
GBA	Architecture d'Amorçage Générique (« Generic Bootstrapping Architecture »)
GPS	Système de positionnement global (« Global Positioning System »)
GSMA	Association GSM (« GSM Association »)
IoT	Internet des Objets (« Internet of Things »)
IP	Protocole internet (« Internet Protocol »)
ISM	Industriel, Scientifique et Médical (« Industrial, Scientific and Medical »)
LAN	Réseau local (« Local Area Network »)
LPWA	Réseau de Longue Portée et faible puissance (« Low Power Wide Area »)
LTE-M	Évolution à Long Terme pour les Machines (« Long Term Evolution for Machines »)
MCU	Unité Microcontroleur (« MicroController Unit »)
NB-IoT	Bande Étroite- Internet des Objets (« Narrowband-Internet of Things »)
NVRAM	Mémoire à accès aléatoire non volatile (« Non-Volatile Random Access Memory »)
OMA	Alliance Mobile Ouverte (« Open Mobile Alliance »)
OTA	Accès aux Paramètres SIM à Distance (« Over The Air »)
PAN	Réseau Personnel (« Personal Area Network »)
PFS	Confidentialité de Transmission Parfaite (« Perfect Forward Secrecy »)
PSK	Clef Pré-Partagée (« Pre-Shared Key")
RAM	Mémoire Vive (« Random Access Memory »)
ROM	Mémoire de seule lecture (« Read Only Memory »)
SCADA	Contrôle et d'Acquisition de Données (« Supervisory Control And Data Acquisition »)
SPI	Interface périphérique série (« Serial Peripheral Interface »)
SSH	Enveloppe de protection (« Secure Shell »)
SIM	Module d'Identité d'Abonné (« Subscriber Identity Module »)
SRAM	Mémoire d'accès aléatoire statique (« Static Random Access Memory »)
TCB	Base informatique Sécurisée (« Trusted Computing Base »)
TLS	Sécurité de la Couche Transport (« Transport Layer Security »)
TTL	Logique Transistor–transistor (« Transistor–Transistor Logic »)
UART	Récepteur / Émetteur Asynchrone Universel (« Universal Asynchronous Receiver/Transmitter »)

1.2 Références

Réf.	Numéro du Document	Titre
[1]	CLP.11	IoT Security Guidelines Overview Document
[2]	CLP.12	IoT Security Guidelines for IoT Service Ecosystem
[3]	CLP.13	IoT Security Guidelines for IoT Endpoint Ecosystem
[4]	CLP.14	IoT Security Guidelines for Network Operators
[5]	OMA FUMO	OMA Firmware Update Management Object www.openmobilealliance.org
[6]	na	ST-LINK/V2 in-circuit debugger/programmer http://www.st.com/
[7]	na	Mobile IoT Initiative https://www.gsma.com/iot/mobile-iot-initiative/
[8]	na	Nmap Security Scanner https://nmap.org/
[9]	CLP.03	IoT Device Connection Efficiency Guidelines https://www.gsma.com/iot/gsma-iot-device-connection-efficiency-guidelines/
[10]	na	Federal Information Processing Standards www.nist.gov/itl/fips.cfm
[11]	na	EMVCo www.emvco.com/
[12]	na	SIM Alliance - Open Mobile API simalliance.org/key-technical-releases/
[13]	GPD_SPE_013	GlobalPlatform Secure Element Access Control www.globalplatform.org/specificationsdevice.asp
[14]	GPD_SPE_024	GlobalPlatform Trusted Execution Environment API Specification www.globalplatform.org/specificationsdevice.asp
[15]	GPC_SPE_034	GlobalPlatform Card Specification www.globalplatform.org/specificationscard.asp
[16]	ISO/IEC 29192-1	Information technology -- Security techniques -- Lightweight cryptography www.iso.org/obp/ui/#iso:std:iso-iec:29192:-1:ed-1:v1:en
[17]	TS 33.220	Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA) www.3gpp.org
[18]	TS 33.222	Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS) www.3gpp.org
[19]	CLP.17	GSMA IoT Security Assessment Checklist https://www.gsma.com/iot/iot-security-assessment/

Réf.	Numéro du Document	Titre
[20]	TS-0003	oneM2M Security Solutions www.onem2m.org
[21]	3GPP TS33.163	Battery efficient Security for very low Throughput Machine Type Communication (MTC) devices (BEST) www.3GPP.org
[22]	na	http://www.blackhat.com/presentations/bh-usa-08/McGregor/BH_US_08_McGregor_Cold_Boot_Attacks.pdf

2 Le défi de sécurité du dispositif périphérique IoT

Un service IoT est exposé á un problème de sécurité qui est, dans de nombreux cas, directement lié aux caractéristiques spécifiques du dispositif périphérique IoT utilisé par le service. Par exemple, les dispositifs IoT ont des caractéristiques très différentes et peuvent subir des problèmes de sécurité particuliers :

2.1 Consommation d'énergie limitée

- Une faible consommation d'énergie peut être nécessaire pour prolonger la durée de vie de la pile (plusieurs années) pour un dispositif périphérique inaccessible à distance sans alimentation permanente ou parce que le dispositif dispose d'une alimentation électrique permanente, mais limitée comme par exemple, par énergie solaire.
- Les dispositifs à faible consommation d'énergie ne peuvent généralement effectuer que des opérations cryptographiques simples (par exemple le dispositif périphérique peut uniquement exécuter les opérations cryptographiques légères définies dans la norme ISO / IEC 29192 [16]) en raison des besoins élevés de consommation d'énergie associés aux opérations cryptographiques plus avancées et ne peuvent supporter que les communications à bande passante limitée contraignant à nouveau la capacité cryptographique.

2.2 Faible coût

- L'analyse de la rentabilisation de nombreux services IoT exige que le coût du dispositif IoT reste bas. Cela se traduit souvent par une capacité de calcul du périphérique limitée, de petites quantités de mémoire et un système d'exploitation limité. Le résultat net est que l'appareil peut être incapable d'effectuer une cryptographie comme celle utilisée dans les services Internet standard.

2.3 Longue vie (>10 années)

- De nombreux paramètres, en particulier pour les applications civiles et industrielles (par exemple un compteur de gaz intelligent), doivent être durables. Cela présente un défi car les choix de conception cryptographique effectués lors de la conception du dispositif périphérique devront être robustes pour sa durée de vie. Par exemple, la puissance de traitement par dollar disponible pour un attaquant au cours de cette période de 10 ans est susceptible d'avoir augmenté de 16 fois alors que les capacités du dispositif risquent de rester statiques.
- La gestion des dispositifs périphériques IoT à longue durée de vie constitue également un défi, en particulier si une vulnérabilité de sécurité ne peut pas être corrigée dans le dispositif.

2.4 Physiquement accessible

- De nombreux dispositif IoT sont physiquement accessibles à l'attaquant. Tous leurs composants matériels et leurs interfaces sont donc susceptibles d'être attaqués et doivent être sécurisés par le développeur.

Le résultat net de ce qui précède est dans de nombreux services IoT, les dispositif IoT ne sont pas directement connectés à des réseaux de communication étendus et n'ont pas de

capacités pour exécuter le protocole Internet (IP). Par exemple, un terminal IoT peut utiliser un émetteur-récepteur radio industriel, scientifique et médical (ISM) pour transférer des données à une passerelle de service IoT locale qui prend ensuite les données et les transmet au réseau de communication via IP, compliquant le processus de sécurisation des communications de bout en bout.

En fonction des capacités des dispositifs périphériques de l'IoT et des risques de sécurité qui y sont associés, il peut être nécessaire d'appliquer différentes méthodes de sécurité avec des degrés de complexité variables, comme expliqué dans la suite de ce document.

3 Le modèle des dispositifs périphériques IoT

Autrefois considéré comme un ensemble de technologies très hétérogènes, interagissant avec le monde physique et se connectant à un serveur quelque part sur Internet pour guider et soumettre des métriques, le modèle du dispositif périphérique IoT a radicalement changé. Dans l'ingénierie moderne, la technologie IoT s'est regroupée en un modèle prévisible composé de seulement plusieurs variantes. Le dispositif IoT devient de plus en plus prévisible et devrait prendre l'une de plusieurs configurations suivantes :

- Dispositif périphérique léger
- Dispositif périphérique complexe
- Passerelle de communication (ou « Hub »)

Dans le diagramme ci-dessous, quelques configurations courantes pour les dispositif périphériques IoT sont montrées :

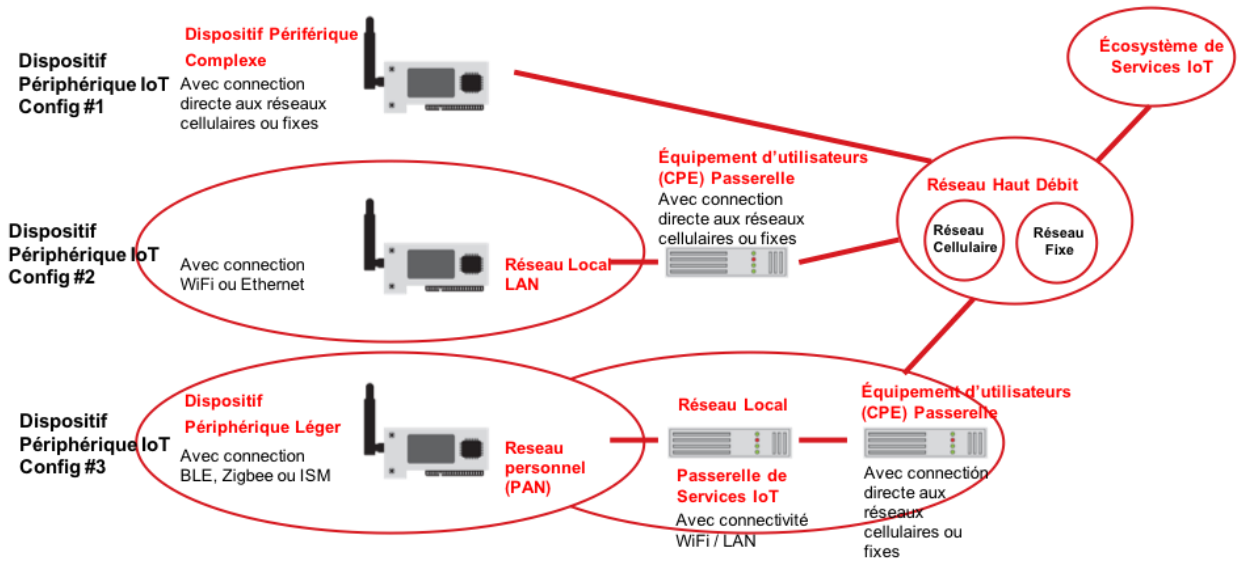

Figure 2 - Exemples de configurations de dispositifs périphériques IoT

3.1 Le dispositif périphérique léger

Ce type de dispositif est généralement un capteur ou un dispositif physique simple, tel qu'un commutateur d'éclairage ou une serrure de porte qui a peu de fonctions. Son but est de

servir un but physique singulier et de fournir des mesures à l'écosystème de services ou au consommateur. Il utilise généralement une CPU très bon marché, peut-être un microcontrôleur à huit bits, et un protocole de communication de réseau personnel, PAN (« Personal Area Network »), ou capillaire à courte distance comme « Bluetooth Low Energy » (BLE), Thread ou Zigbee. Il se caractérise généralement par une faible consommation d'énergie et peut fonctionner à partir d'une pile de type bouton, de l'énergie solaire ou d'une petite batterie au lithium-polymère. Ces périphériques sont généralement connectés à l'écosystème de services via une passerelle de service IoT et une passerelle d'équipement client ou résidentielle, comme indiqué dans la Figure 2 « Exemple de configuration de dispositif périphérique IoT Config #3 ».

Des exemples de dispositifs périphériques légers sont :

- Dispositifs portables
- Capteurs de sécurité résidentiels
- Balises de proximité
- Dispositifs opérant dans des réseaux capillaires non cellulaires

En raison du faible coût des dispositifs légers, les technologies de sécurité disponibles pour ces périphériques sont limitées. Les technologies de sécurité qui nécessitent d'une consommation de courant importante et un espace sur la carte de circuit imprimé aussi considérable et donc couteuses, ne sont généralement pas disponibles pour ces systèmes. Toutefois, les dispositifs périphériques légers peuvent toujours utiliser des ancres de confiance rentables et de petite taille pour implémenter un cadre de sécurité robuste.

3.2 Le dispositif périphérique complexe

Ce modèle de dispositif a généralement une connexion persistante à un serveur principal via une réseau étendu telle qu'un réseau cellulaire (y compris les réseaux LPWA) (voir « Exemple de dispositif périphérique Config # 1 » sur la figure 2) ou se connecte via Wi-Fi ou Ethernet via une passerelle (voir «Exemple de configuration du dispositif périphérique n ° 2» dans la figure 2). L'appareil peut inclure un processeur rudimentaire, même un microcontrôleur à huit bits, mais il est capable d'avoir une unité de traitement plus robuste car il est directement connecté à une source d'alimentation en courant alternatif (CA) ou il contient une batterie avec un accès à un système de recharge de batterie. Certains dispositifs complexes communiquent via des protocoles capillaires, mais nécessitent plus de puissance pour exécuter l'application locale de manière efficace, par exemple un périphérique multimédia.

Les exemples de dispositifs périphériques complexes sont :

- Les systèmes d'éclairage connectés IoT
- Les appareils tels que les réfrigérateurs ou les machines à laver « intelligentes »
- Les systèmes de contrôle industriel (par exemple SCADA)
- Le dispositif type, « Retro-Fit OBD2 », pour une "voiture connectée" permettant des services de control, entretien et de suivi

Les dispositifs périphériques complexes sont capables de consommer plus de courant, intègrent des processeurs plus robustes et disposent de plus d'espace sur la carte de circuit

imprimée disponible pour les technologies de sécurité. En conséquence, beaucoup plus peut être fait avec ces dispositifs. Ces appareils peuvent utiliser presque n'importe quel type d'ancre de confiance. Par conséquent, ils peuvent facilement mettre en œuvre un modèle PSK (« Personal Pre-Shared Key ») ou TCB (« Trusted Computing Base ») asymétrique, comme décrit à continuation dans ce document.

3.3 La passerelle (ou 'Hub')

Une passerelle est un dispositif périphérique, généralement connecté à une source d'alimentation dédiée, qui gère généralement la communication entre les points de terminaison légers et les systèmes dorsaux qui les pilotent. La passerelle gère les liaisons de communication longue distance, telles que cellulaire (LPWA), satellite, ligne fixe, fibre ou Ethernet. Il accepte les commandes des systèmes back-end de l'écosystème de service et les traduit en messages utilisables par les dispositifs périphériques.

Alors que la fonction principale d'une passerelle IoT est d'acheminer des messages vers et à partir de dispositif périphérique légers, elle est également capable d'effectuer des fonctions critiques, telles que :

- Découverte de périphériques
- Déploiement du pilote réseau
- Fonctionnalité de gestion
- Surveillance de l'exécution
- Authentification et sécurité telles que la configuration GBA ou TLS

Bien que les passerelles soient techniquement des dispositifs périphériques, elles ne sont pas nécessairement gérées par l'utilisateur final et peuvent être gérées par le fournisseur de services IoT ou l'opérateur de réseau (voir ci-dessous). Indépendamment de cela, les passerelles peuvent également être conçues comme des dispositifs complexes pour utiliser plus efficacement la distribution d'une liaison montante vers plusieurs dispositifs périphériques légers dans un réseau local.

Tout comme les dispositifs complexes, les passerelles sont capables de plus de puissance de traitement, de consommation de courant et ont généralement plus d'espace disponible sur la carte de circuit imprimée. Cela leur permet de mettre en œuvre des solutions TCB complexes et des technologies telles que les clients d'authentification GBA, avec une relative facilitée.

Ces attributs de la passerelle leur permettent également d'incorporer plusieurs technologies de communication pour acheminer les messages entre différents types de périphériques sur un réseau. Cela permet la communication entre les dispositifs qui seraient normalement incapables d'échanger des messages de manière efficace. De cette manière, les passerelles fonctionnent comme un point d'agrégation pour les dispositifs périphériques de l'écosystème local, ce qui leur permet de communiquer entre eux et, si nécessaire, sur les écosystèmes de réseau et de services.

Il existe généralement deux types de passerelle : une «passerelle de service IoT» et une « passerelle informatique (résidentielle ou pour entreprises) ». La différence est expliquée ci-dessous :

1. Une « passerelle de service IoT » est fournie par le fournisseur de services IoT. Elle peut appartenir à l'utilisateur final, mais elle est généralement géré par le fournisseur de services IoT. Une telle passerelle est généralement utilisée comme concentrateur pour connecter des dispositifs périphériques légers à l'écosystème de services (directement via une connexion fixe / cellulaire ou via une passerelle CPE), où l'utilisateur final achète un service géré auprès d'un fournisseur de services IoT.
2. Une « passerelle informatique » est fournie par un opérateur de réseau. Il s'agit généralement d'un routeur haut débit connecté à Internet par des réseaux cellulaires ou fixes. Cela peut être utilisé dans des environnements résidentiels ou d'entreprise. Dans cette configuration, la passerelle est généralement gérée et configurée par l'opérateur de réseau.

3.4 Le modèle global

Quel que soit le type de dispositif périphérique en cours d'évaluation ou de conception, ils ont tous des modèles de sous-composants similaires d'un point de vue matériel et logistique :

- Une unité centrale de traitement ou processeur (CPU) qui doit exécuter le code de l'application
- Le CPU doit charger / stocker les données et le code exécutable depuis / vers le stockage persistant
- La CPU doit calculer les données dans le stockage temporel
- Une base informatique de confiance doit être utilisée pour authentifier l'environnement
- Le dispositif doit communiquer avec son écosystème IoT

Il est à noter que les dispositifs périphériques légers ont moins de capacités de stockage et de calcul que les complexes ou les passerelles. Ils ont donc généralement moins de capacité de sécurité.

L'aspect le plus important du modèle global est que chaque type de dispositif périphérique a un rôle principal : définir une plate-forme fiable, de haute qualité et sécurisée pour l'exécution d'une application particulière. En d'autres termes, similaire à des plate-formes informatiques plus complexes telles que smartphones, serveurs Cloud et mainframes, avant qu'une application de haute qualité puisse fonctionner de manière fiable ou interagir de manière sécurisée avec ses homologues, l'équipe d'ingénierie doit s'assurer que le matériel présente une plate-forme fiable à l'application.

Les dispositifs périphériques IoT, par nature, font partie d'un réseau d'autres dispositifs. Ce ne sont pas des dispositifs autonomes qui effectuent une action sans l'influence ou la participation d'un service de supervision. Pour augmenter la fiabilité d'un périphérique donné et réduire le risque de responsabilité en raison des lacunes de sécurité ou de fiabilité, chaque dispositif périphérique doit être conçu avec l'idée que la fiabilité de l'ensemble de l'écosystème IoT commence par la construction matérielle du dispositif périphérique.

Dans cette perspective, il est clair que même le type dispositif le plus facile à développer doit se comporter de manière fiable, de haute qualité et sécurisée, car il devrait participer à un réseau possiblement très complexe pouvant atteindre plusieurs millions d'appareils. La

manière dont un seul dispositif se comporte aura certainement un effet sur l'ensemble de son écosystème IoT. Par conséquent, les ingénieurs doivent considérer les implications de la conception de l'architecture bien au-delà des attributs physiques associés à un périphérique embarqué donné. Les ingénieurs doivent réfléchir en termes de sécurité, de fiabilité et de qualité de l'ensemble de l'écosystème IoT.

4 Le modèle de sécurité

La sécurité dans les dispositifs périphériques peut être évaluée du point de vue des composants. En évaluant chaque composant requis pour construire un dispositif périphérique donné, l'ingénieur et l'adversaire peuvent créer un ensemble d'attaques susceptibles d'entraîner un compromis complet du système sans trop d'efforts.

En utilisant le modèle de dispositif périphérique global défini ci-dessus, les composants utilisés peuvent être évalués à partir d'un niveau global. La perspective de haut niveau de chaque composant orientera un analyste vers des technologies couramment utilisées et susceptibles d'être incorrectement sécurisées. En classant par ordre de priorité ces composants à partir du niveau d'expertise, de l'équipement et des coûts requis pour réussir, un analyste ou un adversaire peut créer un modèle d'attaque qui évaluera rapidement tout dispositif périphérique pour les failles de sécurité.

Dans l'Écosystème des Dispositifs Périphériques, il existe plusieurs surfaces d'attaques qui seront examinées par les adversaires en fonction de leurs ressources, de l'accès à l'infrastructure et de l'expertise. Ces surfaces d'attaques sont :

- Communications réseau
- Services réseau accessibles
- Accès à la console
- Communications par bus local
- Accès aux puces

4.1 Attaques contre les communications sur les réseaux

La première et la plus simple démarche pour tenter de compromettre un dispositif périphérique IoT implique généralement des faiblesses dans le modèle de communication. Les analystes observeront si le modèle de communication intègre les meilleures pratiques en matière de sécurité des communications. Si l'analyste peut facilement capturer des informations de connexion, des tokens de communication ou d'autres identifiants que l'écosystème de service utilisera pour identifier le dispositif périphérique, il a compromis le périphérique.

Cette stratégie peut passer d'extrêmement simple à extrêmement difficile. La raison de ceci est l'accès de l'analyste ou de l'adversaire aux données en clair passant par le canal de communication. Un analyste suffisamment équipé aura déjà la technologie pour intercepter les communications pour BLE, 802.15.4, et d'autres protocoles populaires. Étant donné que l'observation ou l'exécution d'une attaque de l´homme du milieu (« man-in-the-middle ») contre les communications d'un dispositif périphérique nécessite généralement peu ou pas de modification du dispositif lui-même, l'adversaire est dans une position très avantageuse. Cela nécessite très peu d'efforts et de travail pour implémenter ce type d'attaque.

Cependant, si le modèle de communication utilise les meilleures pratiques pour assurer la confidentialité et l'intégrité des données, l'adversaire aura une difficulté exponentielle à accéder à des secrets précieux. Cela amènera l'adversaire à passer au prochain modèle d'attaque le plus facile.

4.2 Attaques contre les services des réseaux accessibles

La prochaine démarche dans l'attaque d'un dispositif périphérique IoT est une évaluation des services réseau ouverts. Dans le premier cas, les messages sortants du dispositif périphérique sont capturés pour identifier si des secrets immédiatement utilisables sont accessibles dans les messages. Cela permet à un adversaire de réduire la quantité de travail nécessaire pour extraire des secrets du dispositif périphérique, lui-même. Si le modèle de sécurité des communications sortantes est valide, les services réseau sont analysés pour évaluer si le système d'exploitation du dispositif est accessible ou instrumenté depuis le réseau.

Une évaluation sera effectuée avec un outil tel que NMap [8] pour déterminer si les ports réseau sont ouverts. Si la topologie du réseau n'est pas compatible avec le protocole IP, ce qui est courant dans les réseaux BLE ou IEEE 802.15.4, l'adversaire peut toujours utiliser des outils facilement accessibles pour se connecter au dispositif périphérique via le protocole radio approprié.

L'adversaire tentera alors d'envoyer des messages au dispositif périphérique pour déterminer s'il peut être manipulé en exécutant des commandes ou en fournissant un accès distant à la console du système d'exploitation. Une méthode courante consiste à déterminer si une interface de connexion réseau, telle que Secure Shell (SSH) ou Telnet est disponible. Si les informations d'identification par défaut sont utilisées, l'adversaire peut être en mesure de se connecter au dispositif périphérique. Cela permettra à l'adversaire de manipuler le système d'exploitation local, et potentiellement abuser des vulnérabilités locales pour escalader les privilèges et extraire des secrets de l'appareil.

Un autre exemple courant inclut l'abus de services Web mal conçus, où les commandes peuvent être injectées sur des scripts CGI qui ne suppriment pas correctement les caractères de contrôle des champs d'entrée de l'interface utilisateur, ce qui entraîne l'exécution de code sur le système d'exploitation local.

4.3 Attaques par accès à la console d'administration

L'accès à la console n'est pas exactement une attaque, c'est une stratégie. Généralement, les consoles doivent être activées sur les dispositifs pour permettre aux développeurs et aux techniciens d'assurance qualité de diagnostiquer les anomalies matérielles ou logicielles. Cependant, les informations fournies par une console sont très utiles à un adversaire. En outre, une console peut fournir à un adversaire la possibilité de se connecter au système de dispositifs périphériques localement ou à distance.

Généralement, les consoles matérielles locales peuvent être trouvées sur les dispositifs périphériques en :

- Cherchant un connecteur avec 5 broches sur la carte de circuit imprimée indiquant un port série TTL

- Consultant les spécifications de la CPU ou de la MCU et identifiant les broches UART

Un multimètre peut être utilisé pour identifier un port TTL, car les broches adhèrent à la spécification de tension typique pour TTL. Alternativement, un analyseur logique peut être utilisé pour deviner la vitesse de transmission des broches où les données sont transmises en série. L'analyste sera rapidement capable de discerner si une console est disponible sur le dispositif analysé.

Dans de nombreux cas, le simple fait d'accéder à un port de console permet à un analyste d'accéder directement à une invite de commande sur le dispositif périphérique. Dans d'autres cas, les informations d'identification sont requises, mais elles peuvent généralement être devinées. Si une autre personne sur Internet a identifié les informations d'identification de connexion, et que toutes les informations de connexion des dispositifs périphériques sont les mêmes, tout ce qu'un analyste doit faire est d'effectuer une recherche Google en ligne pour voir si quelqu'un d'autre a posté les informations d'identification.

L'accès à la console distante peut être obtenu via des protocoles de réseau de diagnostic, des protocoles d'accès à la console (par exemple SSH ou Telnet) ou d'autres moyens. Ces méthodologies d'accès doivent être évaluées pour déterminer si un adversaire peut manipuler le canal d'accès, accordant ainsi à l'adversaire l'accès à une console à distance.

4.4 Attaques contre les communications par bus local

Si une invite de commande ne peut pas être obtenue via une console, l'adversaire ou l'analyste devra commencer à inspecter le matériel pour déterminer si le dispositif périphérique est facilement compromis. Cela prend de nombreuses formes différentes, mais il y a des mesures faciles à prendre :

- Les supports accessibles en écriture sont-ils présents et modifiables ?
- Les secrets cryptographiques sont-ils transmis en clair sur les bus matériels ?
- Des messages peuvent-ils être injectés dans le circuit matériel qui influencent le comportement de l'application ou du système d'exploitation en faveur de l'adversaire ?

L'attaque la plus simple identifie si des circuits programmables sont présents. Cela peut être un support simple à modifier, tel qu'une carte de mémoire externe (SD / MMC). Ou bien, une puce NVRAM ou une mémoire EEPROM peut être modifiée avec des modifications d'application ou de configuration pour autoriser l'accès aux invites de commande ou l'accès à des tokens stockés de manière sécurisée.

Si ce vecteur est correctement sécurisé, l'analyste déterminera si les secrets cryptographiques sont passés dans les bus matériels. Cela pourrait impliquer l'utilisation d'un analyseur logique pour intercepter des messages entre une EEPROM et une CPU, un microcontrôleur et un adaptateur de réseau connecté par SPI, ou d'autres attaques. Ces attaques peuvent être extrêmement simples et rapides, ou complexes et coûteuses, selon la complexité de l'attaque et la technologie utilisée.

Si l'adversaire ne peut pas intercepter de précieux secrets en utilisant la méthode ci-dessus, il peut tenter d'injecter des messages dans des bus matériels pour modifier le comportement d'une application s'exécutant sur le dispositif périphérique. C'est une attaque difficile qui

nécessite un haut degré d'expertise, d'équipement et la capacité d'évaluer les données spécifiques à l'application et son contexte.

4.5 Attaques par accès aux puces

Si les attaques ci-dessus sont trop complexes ou coûteuses, l'adversaire doit passer à des attaques encore plus complexes contre le matériel. Cela implique généralement d'abuser de la sécurité de la puce ou des autres composants sur la carte de circuit imprimé. Cela peut inclure :

- Décapage du microcontrôleur ou du processeur
- Extraire des secrets de l'EEPROM interne ou NVRAM
- Intercepter les messages SRAM internes
- Effectuer une analyse par rayons X ou une ingénierie inverse FIB

Toutes ces attaques exigent un haut degré de compétence, des connaissances en ingénierie électronique et un équipement coûteux. Alors que la plupart des organisations n'auront pas à craindre un attaquant utilisant ces méthodes pour faire de l'ingénierie inverse de leurs produits, il s'agit toujours d'une possibilité importante à considérer. La raison en est que ces attaques ne doivent être effectuées qu'une seule fois si les dispositifs périphériques ne sont pas provisionnés avec des secrets cryptographiques uniques.

Si elles ne sont pas provisionnées avec des secrets cryptographiques uniques, une attaque de cette classe va extraire des secrets qui peuvent affecter toute la gamme de produits. C'est un risque important, car si les données sont divulguées au public pour quelque raison que ce soit, la technologie fera l'objet d'attaques et d'abus jusqu'à ce qu'un correctif soit publié, si celui-ci peut être vraiment implémenté.

5 Foires aux questions de sécurité

La sécurité des dispositifs périphériques est décomposée en recommandations par priorité dans ce document. Mais, pour une utilisation pratique, il est plus avantageux d'évaluer les recommandations à partir d'un point de départ pratique. Les ingénieurs commencent généralement à établir une liste de recommandations en fonction d'un objectif technologique ou commercial. Cette section présente les objectifs communs du point de vue du dispositif périphérique et les recommandations pertinentes pour atteindre ces objectifs.

5.1 Comment combattons-nous le clonage ?

La protection de la propriété intellectuelle est un objectif important pour les entreprises modernes. Le matériel, les microprogrammes et les technologies de communication utilisés pour créer un produit du type d'un dispositif périphérique prennent du temps, de l'expertise et des ressources financières que les entreprises ne veulent pas facilement perdre en permettent à d'autres entreprises ou marques moins scrupuleuses de copier le dispositif facilement. Cependant, peu importe ce que fait une entreprise, quelqu'un peut utiliser exactement les mêmes composants matériels pour créer un clone d'un produit donné. La société ne peut rien faire pour empêcher cela en dehors de contrats légaux et de partenariats. Cependant, il existe des moyens rentables d'empêcher quelqu'un d'utiliser un tel clone.

L'implémentation d'une authentification dans les communications entre dispositifs garantit que chaque dispositif périphérique est prouvé cryptographiquement par le fournisseur de services IoT. Chaque fois que les services back-end, ou dispositif périphérique homologue, communiquent avec un autre dispositif, il peut différencier un dispositif périphérique valide d'un clone en forçant celui-ci à s'authentifier. Si le périphérique ne peut pas le faire, l'homologue ou le service peut rejeter la connexion. Cela nécessite les recommandations suivantes pour fonctionner :

- Authentification de l'identité du dispositif périphérique
- Authentification mutuelle mal conçue ou non implémentée

5.2 Comment sécuriser l'identité du dispositif périphérique ?

Pour authentifier correctement un dispositif périphérique, l'ingénieur doit pouvoir faire confiance à l'identité cryptographique de celui-ci. Ceci est plus complexe qu'il n'y paraît et nécessite une combinaison de processus, de politique et de technologie pour atteindre l'objectif. Cela est expliqué plus en détails dans la recommandation « Mettre en œuvre une base informatique de confiance », mais la façon dont les tokens d'authentification sont codés sur un dispositif périphérique déterminera la sécurité du système global.

Dans de nombreuses architectures de dispositifs périphériques, un adversaire peut simplement copier des tokens cryptographiques (s'il y en a) du dispositif cible afin de lui emprunter l'identité. Si chaque dispositif fabriqué par le fournisseur de services IoT utilise le même ensemble de tokens cryptographiques, l'adversaire peut être capable d'usurper l'identité d'un périphérique simplement en compromettant un seul ensemble de tokens.

Ainsi, la construction d'une TCB robuste nécessite les recommandations suivantes :

- Implémenter une base informatique de confiance
- Utiliser une ancre de confiance
- Utiliser une ancre de confiance infalsifiable
- Utiliser une API pour la TCB
- Utiliser un générateur de nombres aléatoires prouvé
- Utiliser un boîtier de produit résistant aux altérations
- Faire respecter la confidentialité et l'intégrité à l'égard de l'ancre de confiance

5.3 Comment réduire l'impact d'une attaque contre l'ancre de confiance ?

Il est également important de noter que la manière dont un périphérique est fabriqué et configuré a un effet considérable sur la sécurité d'un dispositif périphérique en production. Le processus de fabrication déterminera si les dispositifs périphériques sont codés de manière sécurisée avec des clefs. Le processus d'exécution et de provisionnement déterminera comment un dispositif périphérique est associé à un consommateur particulier et si l'appareil peut être compromis avant ou après la création d'une association.

- Considérer la sécurité de la chaîne d'approvisionnement
- Personnalisez chaque dispositif périphérique avant l'exécution
- Provision unique pour chaque dispositif périphérique
- Confidentialité et identificateurs de dispositifs périphériques uniques

5.4 Comment réduire la probabilité d'emprunt d'identité des dispositifs périphériques ?

Après le clonage de dispositifs pour des raisons commerciales, une attaque souhaitable du point de vue de l'adversaire est l'usurpation d'identité d'une personne ou d'un dispositif particulier. Cela peut ou non être directement associé à l'attaque d'un individu particulier. Il peut s'agir simplement de l'usurpation d'identité d'un périphérique dans le but de contourner un contrôle de sécurité, tel qu'un verrou numérique Bluetooth.

Indépendamment de la logique, la lutte contre cette attaque peut être réalisée en utilisant une TCB, la personnalisation, l'authentification, et aussi :

- PFS
- Verrouillage des sections critiques de la mémoire

5.5 Comment refuser la capacité d'usurper l'identité de services ou de dispositifs ?

Tous les réseaux IoT ne sont pas uniquement des dispositifs périphériques, mais aussi des services réseau et des éléments opérants dans les communications. Les dispositifs périphériques doivent être authentifiés par les services, mais les services doivent également être authentifiés par les dispositifs périphériques. Cela garantit que les services critiques, tels que les mises à jour d'applications, ne peuvent pas être détournés vers un compromis supplémentaire sur le réseau.

- Sécurité des communications des dispositifs périphériques
- PFS
- Utiliser un générateur de nombres aléatoires prouvés
- Mises à jour de l'application à distance (OTA)
- Authentification mutuelle mal conçue ou non implémentée
- Collecte de métadonnées non autorisée

5.6 Comment interdire la falsification de micrologiciels et de logiciels ?

Une fois qu'une racine de confiance a été établie, le dispositif périphérique peut s'authentifier à partir d'un composant fiable. Cela permet qu'il établisse une base de confiance et de s'assurer que l'application de l'étape suivante n'a pas été modifiée involontairement (par une NVRAM défectueuse, par exemple) ou intentionnellement par un adversaire. Accomplissez ceci par :

- Plate-forme d'exécution minimale viable (redéploiement de l'application)
- Signer des images d'application cryptographiquement
- Démarrage en dehors de l'EEPROM interne
- Verrouillage des sections critiques de la mémoire
- Bootloaders non sécurisés
- Utiliser un boîtier de produit résistant aux altérations

5.7 Comment réduire la possibilité d'exécution de code à distance ?

Si la falsification d'un micrologiciel ou d'un logiciel physique ne donne pas de résultats adéquats, l'adversaire peut passer à des attaques plus complexes, telles que l'exécution de code contre le bootloader ou les applications qui communiquent via des interfaces bus ou réseau. Si tous les homologues du réseau sont authentifiés, comme décrit plus haut dans ce chapitre, il sera beaucoup plus difficile pour un adversaire d'injecter du contenu malveillant. Pourtant, la plupart des appareils requièrent d'une manière ou d'une autre des communications publiques (par internet normalement) pour interagir avec des appareils d'autres organisations. Par conséquent, ils ne seront peut-être pas en mesure de faire respecter les restrictions sur l'origine des données.

Ainsi, l'entrée de données dans le système informatique à partir d'interfaces distantes et physiques doit être fortement contrôlée. Pour limiter le potentiel d'exploitation d'une application et limiter l'exposition une fois qu'une application est compromise, tenez compte des éléments suivants :

- Appliquer la protection de la mémoire
- Utiliser la mémoire interne pour les secrets
- Mises à jour de l'application par des protocoles OTA
- Exécuter des applications avec des niveaux de privilège appropriés
- Appliquer la séparation de fonctions dans l'architecture d'application
- Appliquer la sécurité au niveau des langages de programmation
- Appliquer les améliorations de sécurité au niveau du système d'exploitation
- Sécurité de l'interface utilisateur
- Audit de code tiers

5.8 Comment interdire le débogage non autorisé ou l'instrumentation de l'architecture ?

Un attaquant possédant des connaissances en architecture et un accès aux outils de débogage tentera généralement d'employer des utilitaires de débogage et de diagnostic standard pour accéder aux secrets du système ou pour modifier ou injecter du code qui peut être parfaitement exécuté. Restreindre la capacité d'un adversaire à le faire diminuera le potentiel d'attaques rapides et furtives qui peuvent ne pas être détectées par un consommateur.

- Utiliser une ancre de confiance inviolable
- Journalisation et diagnostic
- Verrouillage des sections critiques de la mémoire
- Détection d'anomalie
- Utiliser un boîtier de produit résistant aux altérations
- Désactiver les technologies de débogage et de test
- Sécurité de l'interface utilisateur

5.9 Comment gérer les attaques par canaux auxiliaires ?

Quand un adversaire est à court d'options typiques, il se tournera vers des attaques plus ésotériques afin d'extraire des secrets d'un appareil. Ces attaques évaluent le comportement

du hardware afin de déterminer si une tendance dans le comportement peut être comparer et assimiler à une valeur, telle qu'une valeur « 1 ou 0 », ou une instruction particulière. Ceci, avec le temps, donnera à l'analyste la capacité de désosser les données traitées par le système embarqué.

En outre, l'adversaire peut utiliser une technologie d'analyse coûteuse pour extraire des secrets de l'appareil, ou pour construire des circuits extrêmement petits qui pontent les connexions à travers les couches de sécurité dans le silicium. Bien que ces attaques soient extrêmement difficiles à combattre, il y a certaines choses que le concepteur peut faire pour dissuader les attaques :

- Personnalisez chaque périphérique avant l'exécution
- Utiliser la mémoire interne pour les secrets
- Utiliser un boîtier de produit résistant aux altérations
- Mémoire corrompue quand des attaques périphériques sont détectés
- Mettre en œuvre des seuils de verrouillage environnemental
- Appliquer les seuils d'avertissement pour l'alimentation des dispositifs
- Mise hors service et caducité des dipositifs périphériques IoT
- Vaincre les composants répliqués et les ponts non fiables
- Vaincre une attaque de démarrage à froid
- Combattre les faisceaux d'ions focalisés (FIB) et les rayons X

5.10 Comment mettre en œuvre une gestion à distance sécurisée ?

La gestion à distance est une partie essentielle du cycle de vie des dispositifs périphériques IoT qui doivent être protégés pour garantir que le canal utilisé pour la gestion et l'administration ne peut pas être utilisé abusivement. Ce n'est pas seulement un problème avec des adversaires tiers inconnus. Des abus internes peuvent également se produire, soit dans le cercle des consommateurs, soit au sein du fournisseur de services IoT.

- Gestion des mots de passe du dispositif périphérique
- Administration des dispositifs à distance
- Journalisation et diagnostic
- PFS
- Utiliser un APN privé

5.11 Comment détecter les dispositifs périphériques compromis?

En fonction de l'architecture du dispositif, il peut être pratiquement impossible de déterminer si le matériel ou le micrologiciel a été altéré si le périphérique se comporte normalement. Cependant, un périphérique compromis peut être détecté par un comportement anormal tant que l'infrastructure effectue le suivi, la journalisation et l'alerte lorsque des anomalies sont détectées. Considérez les recommandations suivantes :

- Détection d'anomalies
- Utiliser un boîtier de produit résistant aux altérations
- Appliquer les seuils d'avertissement pour l'alimentation des dispositifs

5.12 Comment déployer en toute sécurité un dispositif périphérique sans une connexion back-end ?

Il y a des moments où une connexion à un environnement back-end n'est ni disponible ni souhaitée. Dans ces conditions, la sécurité devient plus difficile en raison de l'incapacité évidente à gérer les clefs de sécurité, les identités et les mécanismes d'authentification dynamiques. Cependant, un niveau de sécurité acceptable peut être atteint. Considérez ce qui suit :

- Implémenter une Base Informatique de Confiance
- Définition d'une racine de confiance organisationnelle
- Personnalisez chaque dispositif périphérique avant l'exécution
- PFS
- Authentification de l'identité du dispositif périphérique
- Environnements sans connectivité back-end

5.13 Comment assurer la confidentialité des clients ?

La protection de la vie privée des consommateurs est une question complexe qui nécessite une analyse approfondie non seulement de la technologie des dispositifs, mais de l'ensemble du produit ou du service. Chaque composant du système global doit être analysé pour détecter d'éventuelles lacunes dans la confidentialité. Passez en revue les recommandations suivantes pour mieux comprendre l'application de la confidentialité :

- PFS
- Sécurité des communications des dispositifs périphériques
- Gestion de la confidentialité
- Confidentialité et identités des dispositifs périphériques uniques
- Utiliser un APN privé
- Collecte de métadonnées non autorisée
- Risques de sécurité non évidents (« voir à travers les murs »)
- Interception légale

5.14 Comment assurer la sécurité de l'utilisateur tout en appliquant la confidentialité et la sécurité ?

La sécurité est un sujet qui doit être considéré en fonction de l'application, de son objectif, de l'environnement dans lequel l'application va s'exécuter, du type de consommateur et de la technologie de communication utilisée. Il semble souvent que des compromis doivent être faits entre la sûreté et la sécurité. Cela peut ne pas être vrai, cependant. Au lieu de cela, le modèle architectural peut devoir être déplacé afin de maintenir à la fois la sûreté et la sécurité. Lorsque cela est possible, la sécurité ne doit pas être écartée en faveur de la sûreté. Les deux devraient être appliquées, dans la mesure du possible. Bien que ce soit une recommandation philosophique, il est important que la sécurité soit constamment examinée par l'équipe d'ingénierie. Tenez compte des recommandations suivantes pour lancer une discussion sur la sécurité dans l'IoT :

- Analyse critique de sécurité
- Déni de service intentionnel et non intentionnel

- Interception légale
- Considérer la sécurité de la chaîne d'approvisionnement

5.15 Quels problèmes ne peuvent pas être résolus ?

Dans chaque système, il existe des risques qui ne peuvent être résolus en raison des lois de la physique, du coût ou simplement du manque de solutions technologiques. Certains de ces problèmes sont notés ici :

- Déni de service intentionnel et non intentionnel
- Vaincre les composants répliqués et les ponts non fiables
- Risques de sécurité non évidents (« voir à travers les murs »)
- Combattre les faisceaux d'ions focalisés et les rayons X
- Considérer la sécurité dans la chaîne d'approvisionnement
- Interception légale

6 Recommandations critiques

Lors du développement d'un dispositif périphérique sécurisé, les recommandations suivantes doivent toujours être considérées. Les recommandations critiques aident à définir l'architecture d'un dispositif périphérique sécurisé. Sans ces recommandations, le dispositif périphérique aura un profil de sécurité incomplet qui sera abusé par un adversaire.

6.1 Implémenter une TCB pour les dispositifs périphériques

La première étape de la sécurisation de tout système embarqué est la définition de la TCB. Dans le contexte d'un dispositif périphérique (ou de périphériques embarqués similaires), une TCB est un ensemble de matériels, logiciels et protocoles qui garantit l'intégrité du dispositif, effectue une authentification mutuelle avec les homologues réseau et gère la sécurité des communications et des applications.

Le cœur de la TCB est l'ancre de confiance, pour sécuriser la technologie matérielle qui stocke et traite les secrets cryptographiques tels que les clefs pré-partagées (PSK) ou les clefs asymétriques. Les ancres de confiance, telles qu'une puce UICC, peuvent être utilisées pour authentifier non seulement les homologues pendant les communications réseau, mais peuvent être améliorer pour stocker des données utiles pour la sécurité de l'application du dispositif périphérique.

Une fois l'ancre de confiance choisie et intégrée dans la solution du dispositif périphérique, les bibliothèques du logiciel pour la TCB peuvent être choisies. La TCB permettra au système d'exploitation et aux applications principales du dispositif périphérique de gérer plus facilement la sécurité globale non seulement du périphérique, mais du réseau.

Il est important que l'équipe d'ingénierie choisisse la bonne ancre de confiance pour la solution, car chaque combinaison d'ancre de confiance et de TCB donnera un niveau de sécurité différent. Certaines combinaisons et implémentations d'ancre de confiance entraîneront un faux sentiment de sécurité.

Les variantes les plus courantes d'une base d'informatique de confiance ou TCB, par ordre de «moins sécurisé» à «plus sécurisé», sont les suivantes :

- Aucune implémentée (texte brut)
- Clef pré-partagée statique (PSK)
- Clef publique statique
- PSK personnalisé
- Clef publique personnalisée

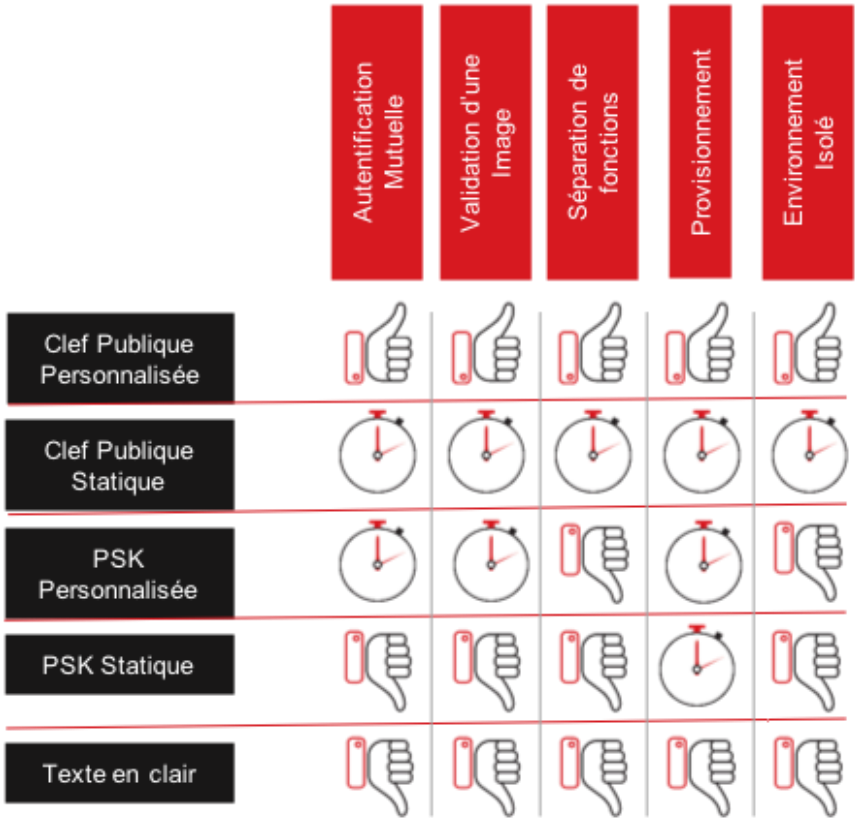

Figure 3 - Garanties de sécurité fournies pour chaque type de TCB.

Considérez la figure ci-dessus. Dans cette table, les capacités de chaque variante TCB a un poids associé. L' icône avec le pouce vers le bas indique que le modèle TCB ne peut pas s'adapter à la stratégie de sécurité décrite le long de la rangée supérieure. L'icône avec un chronomètre indique que la stratégie de sécurité peut être utilisée, mais qu'elle fera l'objet d'une violation de sécurité dans un temps raisonnable. L' icône avec un pouce vers le haut montre que la stratégie de sécurité peut être correctement mise en œuvre et que la durée de vie de la stratégie de sécurité sera probablement longue.

Alors que la TCB peut être utilisé pour sécuriser de nombreux aspects du produit et du service IoT global, ce document se concentre sur cinq concepts de base :

- La validation d'une image d'application exécutable
- L'authentification mutuelle des éléments du réseau
- La séparation des fonctions dans l'architecture de sécurité IoT
- Le provisionnement et personnalisation

- La sécurité de l'environnement isolé (ou sécurité du site sans connexion)

Une TCB qui implémente la validation d'une image exécutable sécurise le dispositif périphérique en vérifiant de manière cryptographique chaque image exécutable à charger et à exécuter par le périphérique. Ce processus commence au bootloader, qui doit valider cryptographiquement l'étape suivante de l'exécution, généralement un noyau du système d'exploitation. Le bootloader peut également valider l'image du système d'exploitation ou une image d'application de microprogramme stockée dans la NVRAM.

Une TCB qui implémente une authentification mutuelle des éléments du réseau permet d'obtenir une racine de confiance pour l'authentification des composants du réseau et s'authentifie cryptographiquement auprès des dispositifs avec lesquels elle se communique. Cela augmente la probabilité que les éléments sur le réseau représentent les identités qu'ils prétendent représenter. Par exemple, si un dispositif du réseau prétend offrir un service de mise à jour du microprogramme, la TCB authentifie l'homologue dans la communication comme faisant partie du réseau du fournisseur de services IoT de base avant d'accepter les mises à jour du microprogramme de l'homologue.

Une TCB qui implémente une séparation des fonctions utilise une hiérarchie de clefs pour identifier les différents composants ou services dans les offres du fournisseur de services IoT. Par exemple, un ensemble de clefs cryptographiques pourrait représenter un service de mise à jour de micrologiciel, tandis qu'un second ensemble de clefs cryptographiques pourrait représenter un service "push". Puisque ces services ont des fonctionnalités complètement différentes, ils ne devraient pas utiliser les mêmes clefs et identités cryptographiques pour la communication. En tant que tel, la TCB devrait gérer et vérifier chaque identité pour séparer un service ou une fonction d'une autre. Ceci réduit la possibilité pour un adversaire de compromettre l'infrastructure de service IoT entière si l'une des clefs cryptographiques est compromise. En d'autres termes, si un attaquant compromet la clef du "service push", il n'aura pas la possibilité d'usurper l'identité du service de mise à jour du micrologiciel.

Une TCB qui implémente la personnalisation et l'approvisionnement assure que le dispositif périphérique possède une identité cryptographiquement unique par rapport aux autres dispositifs du même type. Il garantit également que toutes les identités des communications sont protégées afin de réduire le risque de fuites de confidentialité ou de suivi.

Une TCB qui implémente la sécurité de l'environnement isolé applique des politiques et des procédures qui valident l'authenticité des pairs et la confidentialité et l'intégrité des données même s'il n'y a pas de service back-end pour aider dans le processus. En d'autres termes, si la communication vers les services dorsaux est interrompue pendant une période prolongée, l'écosystème localisé de l'IoT sera toujours capable de fonctionner avec un haut degré de sécurité. Bien que l'intégrité des environnements isolés se dégrade avec le temps, une TCB bien conçu qui implémente une sécurité d'environnement isolée peut renforcer la résilience du réseau et allonger la durée de la sécurité de l'environnement.

Dans ce contexte, *personnalisé* est indicatif d'un ensemble unique de clefs associées à une ancre de confiance spécifique. Le processus de personnalisation comprend la génération et l'installation des clefs uniques, l'association des clefs avec la puce unique et la diffusion

sécurisée de ces informations et des métadonnées pertinentes aux autorités compétentes. Cela garantit que chaque puce possède une identité cryptographique unique. *Statique*, dans ce contexte, fait référence au même jeu de clefs utilisé pour chaque dispositif périphérique.

Alors que les TCB peuvent être utilisées pour résoudre presque tous les problèmes de sécurité qu'un système embarqué peut avoir, il y a plusieurs problèmes de base qu'une TCB doit être capable de résoudre.

- La validation de l'image de l'application pour un dispositif périphérique
- L'authentification réseau et / ou l'authentification des pairs dans les communications
- La séparation de fonctions
- Le provisionnement et la personnalisation
- Le provisionnement et la communication dans un environnement isolé (site sans connexion)
- La randomisation

Bien qu'il soit évident que choisir de ne pas implémenter une TCB entraîne un manque de sécurité, il y a des subtilités aux autres implémentations TCB communes qui devraient être adressées. Si ces subtilités ne sont pas traitées, elles peuvent entraîner des lacunes importantes en termes de sécurité.

6.1.1 Modèles de clefs pour une ancre de confiance

6.1.1.1 Clefs statiques

Une implémentation de clef statique, si c'est une clef PSK ou asymétrique, est définie comme une solution où chaque dispositif périphérique utilise le même secret cryptographique pour résoudre un problème donné. Bien que différentes clefs puissent être utilisées pour résoudre différents problèmes de base, la clef est toujours la même pour chaque dispositif périphérique.

Ce modèle semble sûr en ce que chaque problème résolu par la TCB peut être fait efficacement. Cependant, la durée de vie de la solution globale peut aller de très courte à extrêmement courte. Selon la sécurité de l'ancre et l'algorithme cryptographique et la taille de clef choisie, les adversaires peuvent être en mesure de rompre la solution presque immédiatement.

Le problème se pose vraiment en ce que l'engagement unique de la clef expose chaque système de dispositifs périphériques à un compromis. Cela dévalue l'implémentation TCB et annule le temps et l'argent utilisés pour implémenter la solution dans l'architecture d'un dispositif et dans l'écosystème IoT globale. Ainsi, ce modèle pour une TCB est dangereux à mettre en œuvre tel qu'il est, effectivement, une bombe à retardement.

6.1.1.2 Clefs personnalisées

Indépendamment de la mise en œuvre de PSK ou d'une solution asymétrique, la personnalisation est impérative pour qu'une TCB fonctionne efficacement. La personnalisation annule la capacité d'un adversaire à utiliser une ancre de confiance compromise pour subvertir la sécurité de l'ensemble de l'écosystème IoT. Si un adversaire est seulement en mesure de compromettre un seul dispositif périphérique à la fois, et

nécessitent un accès physique pour le faire, un compromis du système IoT et de la technologie utilisée sera extrêmement lent, coûteux et complexe à mettre en œuvre. Ceci est une victoire significative pour l'entreprise.

En raison des normes en matière de communications cellulaires qui ont évolué au cours des dernières décennies, les opérateurs de réseau ont perfectionné le modèle PSK pour la personnalisation des ancres de confiance, telles qu'à partir d'une carte UICC. Par conséquent, l' UICC peut parfois être configuré pour servir comme une ancre de confiance pour les applications dans un dispositif périphérique IoT, ce qui contribue à une solution de sécurité rentable pour les applications IoT. Dans un proche avenir, lorsque les dispositifs eUICC seront disponibles, cette fonctionnalité peut être activée même sur une puce eUICC déjà déployé sur le terrain.

Aujourd'hui, la technologie de clefs personnalisées est la solution de sécurité la plus efficace pour une ancre de confiance. Les TCB implémentés dans IoT aujourd'hui devraient être basés sur une solution TCB personnalisée. Les fournisseurs de services IoT doivent discuter avec leur opérateur de réseau afin de déterminer si l'UICC ou la carte SIM peut être implémentée en tant qu'ancre de confiance pour la couche application.

6.1.2 Protocoles et technologies de la TCB

Avec une ancre de confiance, la TCB doit incorporer des protocoles, des politiques et des bibliothèques de logiciels pour assurer la sécurité du produit ou du service IoT global. L'un des avantages de l'utilisation des ancres de confiance standard soutenues par la technologie cellulaire est la possibilité de supprimer les logiciels de provisionnement et de personnalisation qui existent déjà pour les opérateurs de réseau. Les technologies, les protocoles et les suites d'applications tels que les suivants aideront la capacité de la TCB à authentifier le dispositif périphérique sur le réseau :

- application oneM2M SM UICC comme spécifié dans oneM2M TS-0003
- Architecture d'amorçage générique (GBA) 3GPP TS 33.220 (voir l'annexe A)

L'utilisation de ces technologies permettra d'accélérer la mise en œuvre du provisionnement et de la personnalisation, car les bibliothèques et les protocoles ont été approuvés par des ingénieurs expérimentés et des analystes de sécurité pendant de nombreuses années. Cependant, ces protocoles ne permettent pas toujours à la TCB de valider l'application du dispositif périphérique ou de s'assurer que le dispositif est correctement authentifié ou autorisent des actions. La TCB doit inclure d'autres protocoles pour accomplir ces tâches, tels que la validation du micrologiciel, la validation des messages de mise à jour par liaison radio, etc.

Dans un futur proche, une technologie telle que l'eUICC augmentera les capacités du point de vue de l'application, et l'UICC proactive activera la technologie à double usage qui peut aider à amorcer le dispositif périphérique lui-même, tout en gérant la sécurité du réseau. Il s'agit d'une augmentation importante car les opérateurs de réseau peuvent gérer à distance et en toute sécurité le périphérique eUICC pour le compte du fournisseur de services IoT. En outre, la « Gestion du Contenu de la Carte Confidentielle » spécifiée dans la spécification de la carte GlobalPlatform [15] permet à plusieurs acteurs des écosystèmes de service IoT de

gérer leur propre application indépendamment l'un de l'autre, si cela est autorisé par l'opérateur réseau.

6.1.3 Risque

Choisir de ne pas implémenter une TCB est un point faible critique pour la sécurité de toute l'architecture IoT. Sans une TCB bien défini, l'interaction entre l'ancre de confiance et l'application de base sera vaguement définie, et peut avoir des lacunes qui peuvent être subvertis par les adversaires. La TCB garantit que les communications entre l'ancre de confiance, l'application principale et les homologues réseau sont sécurisées, fiables et à jour. Sans TCB, il n'y a pas de composant central pour gérer le cycle de vie de la sécurité des dispositifs périphériques.

6.2 Utiliser une ancre de confiance

Pour qu'un dispositif périphérique puisse participer à un écosystème, il doit pouvoir vérifier l'intégrité de sa propre plate-forme et être capable d'authentifier l'identité de ses homologues. Pour ce faire, les dispositifs périphériques nécessitent une ancre de confiance incorporée dans une TCB.

Une ancre de confiance est un élément matériel sécurisé, soit une puce physique distincte, soit un noyau sécurisé à l'intérieur d'une CPU, capable de stocker et de traiter en toute sécurité des secrets cryptographiques. Une carte UICC ou eUICC est un exemple de technologie sécurisée pouvant être utilisée comme élément de confiance pour stocker des secrets d'authentification.

L'utilisation d'un élément de confiance implique effectivement le stockage, la vérification, la mise à jour et le traitement des données. Les données peuvent être des informations secrètes ou publiques qui doivent être vérifiées cryptographiquement. Dans les deux cas, l'ancre de confiance doit être en mesure de déterminer de manière sécurisée si les messages et les identités peuvent être authentifiés, et doit être en mesure de communiquer de manière sécurisée à la TCB le résultat de toutes les opérations d'authentification ou de chiffrement. Cela permet à l'Application et à la TCB de prendre des décisions importantes qui affecteront la sécurité du dispositif périphérique global. Par exemple, l'ancre de confiance peut aider un dispositif périphérique à déterminer si un homologue réseau emprunte l'identité d'une ressource critique, telle qu'un serveur de déploiement de correctifs. Si l'ancre de confiance ne peut pas valider un élément dans le réseau, la TCB et l'application sur le dispositif périphérique doivent choisir de ne pas interagir avec celui-ci, et alerter l'utilisateur, si possible, de la ressource réseau frauduleuse.

Grâce à la baisse du coût des composants et à une forte augmentation de la demande, les ancres de confiance deviennent plus disponibles que jamais. Cela inclut non seulement la technologie d'ancres de confiance actuelle, mais également les bibliothèques et les interfaces approuvées pour une utilisation plus facile avec la technologie. Cela permet à l'équipe d'ingénierie de se tourner vers une solution d'ancre de confiance en très peu de temps, et de s'assurer que la longévité de la technologie n'est pas affaiblie par un logiciel personnalisé ou des normes mal implémentées. Dans la mesure du possible, les normes devraient être utilisées pour réduire le potentiel de lacunes en matière de sécurité.

Un autre défi dans l'implémentation d'une l'ancre de confiance dans un dispositif périphérique léger est la taille du composant. Si une ancre externe est utilisée, il sera nécessaire de maintenir un profil très réduit de composant. La réalisation de ce profil est difficile lorsque le facteur de forme recommandé nous mène vers une technologie telle qu'une UICC. Cependant, la norme ETSI TS 102 671 résout ce problème en introduisant un très petit facteur de forme d'environ 6 millimètres sur 5 millimètres. Ces extensions "MFF1" et "MFF2" à la carte à puce UICC permettent un accès total aux technologies supportées par l'UICC tout en assurant des exigences d'espace dans la PCI minimales. Une sécurité supplémentaire est ajoutée en utilisant un facteur de forme provisionné sur site qui est soudé sur le périphérique, ce qui rend plus difficile pour un adversaire le transfert l'identité du périphérique vers un autre dispositif.

Les dépenses engagées pour le développement et le déploiement d'une ancre de confiance peuvent inclure :

- Le coût de la technologie de base, intégrée dans le processeur ou une puce séparée
- Le coût de l'intégration de la technologie dans le circuit, si nécessaire
- Le coût de l'ingénierie ou de l'intégration du driver dans le système d'exécution et la TCB
- Le coût de l'ingénierie de l'application pour utiliser l'ancre de confiance
- Maintenir l'ancre de confiance, si nécessaire
    - Gestion des clefs de sécurité, révocation des clefs et des identités de déclassement
    - Maintenir l'infrastructure requise pour sécuriser et gérer les clefs et les métadonnées
- Surveillance de l'identité de l'ancre de confiance du côté Service
    - Mettre en place une liste noire de dispositifs, le cas échéant
- Intégrer les services de l'opérateur, là où ils sont disponibles, pour surveiller et gérer les ancres de confiance telles que l'UICC

6.2.1 Risque

Les risques de ne pas utiliser une ancre de confiance sont nombreux, mais tous proviennent du même problème de base : la capacité pour un adversaire de voler des clefs pertinentes pour l'ensemble de l'écosystème IoT. Le résultat de ceci est que l'adversaire peut :

- Cloner les Identités des dispositifs périphériques
- Usurper l'identité des services IoT
- Déployer des correctifs ou des mises à jour non autorisés
- Introduire des modifications non autorisées au logiciel du dispositif périphérique

Ces lacunes en matière de sécurité peuvent entraîner des problèmes coûteux pour l'entreprise au fil du temps, et les adversaires, mais aussi les concurrents, peuvent abuser de l'infrastructure en leur faveur.

6.3 Utiliser une ancre de confiance résistante aux altérations physiques

Certaines ancres de confiance ont une sécurité physique supplémentaire pour se prémunir contre certaines classes d'attaques, telles que les FIB, l'analyse des canaux latéraux et le « glitching ». Alors que certaines attaques, telles que l'utilisation d'un FIB, sont presque impossibles à éviter du point de vue des coûts, la fabrication d'une ancre de confiance peut utiliser des technologies modernes pour rendre les attaques plus coûteuses. Plus une attaque est coûteuse, moins elle est susceptible d'être utilisée contre des dispositifs périphériques de manière aléatoire. Au lieu de cela, les attaques seront concentrées sur des cibles où la dépense vaut la récompense.

Dans un proche avenir, certains fournisseurs prévoient de déployer des variantes de leur technologie, telles que FIPS (« Federal Information Processing Standards ») [10], EMVCo [11] et les Critères Communs approuvés. Les ingénieurs qui développent une nouvelle technologie devraient déterminer si leurs conceptions actuelles seront compatibles avec le remplacement de modules conformes aux nouvelles normes dans le futur.

Pour plus d'informations, consultez la dernière version de chaque norme, pour analyser les caractéristiques offertes par votre fabricant. Notez que certains niveaux de sécurité sont intentionnellement proches de l'impossible pour les dispositifs grand public en raison du coût et de la complexité des implémentations.

6.3.1 Risque

Le risque de ne pas utiliser une ancre de confiance inviolable est extrêmement élevé. Par exemple, si une ancre de confiance est simplement constituée par des clefs cryptographiques intégrées dans la NVRAM, tout attaquant disposant des outils et de la compétence pour extraire ces clefs peut potentiellement subvertir l'infrastructure entière. Cependant, si les secrets sont stockés dans une ancre de confiance infalsifiable, la dépense pour extraire les secrets est élevée, ce qui rendra moins probable que les secrets soient extraits, dévaluant l'ancre de confiance comme cible d'attaque potentielle.

Il est à noter que si l'implémentation de l'ancre de confiance est « faible », l'extraction de secrets aboutissant à un compromis peut être suffisamment élevée. Tout compromis invalidera les dépenses encourues pendant l'ingénierie, l'architecture, la production et l'exécution. Cela peut entraîner une perte financière importante. Par conséquent, s'assurer que l'entreprise a conçu la bonne implémentation est impératif.

6.4 Utiliser une API pour la TCB

Une fois qu'une racine de confiance a été établie au sein de la TCB, il faut utiliser un protocole qui incorpore efficacement les capacités du TCB et la racine de la confiance. L'API devrait s'assurer que :

- Toutes les vérifications de signature sont effectuées par la TCB
- Aucune clef privée n'est exposée à partir de la TCB

- L'échange de clefs peut être effectué par la TCB au nom de l'application
- Le décryptage peut être effectué par la TCB
- Le cryptage peut être effectué sur la TCB
- La signature des messages peut être effectuée sur la TCB
- Le « remplissage » sécurisé des messages peut être effectué sur la TCB
- Confidentialité et intégrité entre la TCB et l'application

Cet ensemble de fonctionnalités aidera à garantir que la TCB n'expose jamais les ressources de sécurité critiques à une application ou à un environnement matériel non sécurisé. Cela peut être accompli en utilisant une spécification existante qui applique ces exigences d'une manière uniforme. Envisager d'évaluer :

- SIM Alliance Open Mobile API [12]
- Contrôle d'accès aux éléments sécurisés GlobalPlatform [13]
- Spécification de l'API TEE (« Trusted Execution Environment ») GlobalPlatform [14]
- Groupe de calcul de confiance (TCG)
- oneM2M TS-0003 [20]

De nombreuses ancres de confiance viendront avec des bibliothèques de logiciels qui peuvent être implémentées en tant que TCB. Ces bibliothèques auront des API que les ingénieurs peuvent utiliser pour interagir avec la TCB. Les bibliothèques fournies par l'ancre de confiance sont préférables, lorsqu'elles sont disponibles, car elles ont probablement été vérifiées par des experts dans le domaine du développement de l'ancre de confiance. Cependant, l'équipe d'ingénierie devrait évaluer la liste des exigences énoncées dans cette recommandation, et devrait s'assurer que la bibliothèque tient suffisamment compte de ces préoccupations.

De plus, les TCB ne devraient être accessibles qu'à partir d'applications privilégiées s'exécutant sur le dispositif périphérique. Une interface TCB ne doit jamais être accessible à partir d'une application non privilégiée ou non approuvée (partie tierce) exécutée sur le dispositif. Tous les accès doivent passer par un proxy via un service de confiance qui évalue les demandes et, éventuellement, alerte l'utilisateur lorsque des demandes suspectes ou centrées sur la confidentialité sont effectuées par des applications non fiables.

Le défi dans la mise en œuvre de ce protocole est de garantir que tous les messages ne peuvent pas être altérés entre le point d'origine des données et la TCB, et vice-versa. Il est plus efficace si un segment d'EEPROM, appelable depuis l'application, peut effectuer ces fonctions pour le compte de l'application. En isolant le noyau du code API sur l'EEPROM interne et en utilisant la RAM interne pour traiter les messages, des données moins critiques seront exposées à des bus externes.

6.4.1 Risque

Si une interface de protocole d'application n'est pas bien définie, l'utilisation d'une TCB peut avoir des résultats inattendus ou des effets secondaires. En définissant le protocole à l'avance et en le vérifiant pour les problèmes de logique et de sécurité, l'équipe d'ingénierie peut identifier plus rapidement et plus efficacement les failles qui peuvent entraîner des problèmes de sécurité plus tard. Ainsi, la définition du protocole doit intégrer l'évaluation des API existantes qui intègrent les besoins du fournisseur de services IoT. Si une technologie

existante et bien établie peut être identifiée, elle sera toujours favorable par rapport à une solution personnalisée.

6.5 Définition d'une racine de confiance organisationnelle

Une racine organisationnelle de confiance est un ensemble de politiques et de procédures cryptographiques qui régissent la façon dont les identités, les applications et les communications peuvent et doivent être sécurisées par chiffrement. Une cryptographie forte devrait être utilisée, soit sous la forme de clefs symétriques uniques, de certificats ou de clefs publiques. Cela dépend du modèle disponible pour l'utilisation dans la TCB, les capacités de l'ancre de confiance, et ce qui est logique pour l'équipe d'ingénierie.

Une clef privée racine, soit symétrique ou asymétrique, doit être utilisée pour signer numériquement les autres clefs utilisées dans la hiérarchie. Par exemple, si notre exemple d'entreprise, « IoT Company LLC », souhaite créer une racine organisationnelle de confiance, elle génère une clef racine sur une machine approuvée. Cette clef représentera la racine organisationnelle. Elle générerait ensuite de nouvelles clefs représentant chaque organisation dans l'entreprise qui devrait avoir des hiérarchies de sécurité indépendantes. Les exemples peuvent être :

- Clef de signature de code
- Clef de communication du serveur
- Clef de communication de pair à pair
- Clef d'identité du dispositif périphérique
- Clef de révocation maître

Chacune de ces clefs doit être signée par la clef racine de l'organisation. Toutes ces clefs, leur signature correspondante et la clef racine doivent être stockées dans l'ancre de confiance utilisée par la TCB. Ensuite, chaque fois que l'application liée à une clef particulière est utilisée, l'application peut utiliser les clefs spécifiques pour valider les messages envoyés sur les canaux de communication.

Ce modèle permet de garantir que tous les messages sont sécurisés via la hiérarchie cryptographique. En séparant les fonctions entre des types de clefs spécifiques, les clefs compromises peuvent être révoquées via le même processus de communication.

Certains protocoles existants qui aident à déployer cette méthode sont :

- TLS (« Transport Layer Security »); la dernière spécification valide
- Shell sécurisé (SSH2)
- Protocole d'approbation de certificat en ligne (OCSP) IETF RFC 2560
- Architecture d'amorçage générique (GBA) (voir l'Annexe A) 3GPP TS 33.220

Des difficultés surviennent lorsque des services nécessitant les clefs cryptographiques doivent être déployés. Au lieu de placer un actif critique pour la sécurité, tel que la clef de communications du serveur principal, sur un serveur Web accessible sur Internet, un certificat ou une paire de clefs séparé doit être généré spécifiquement pour la couche du système qu'héberge ce serveur. Ensuite, ce certificat peut être signé par la clef de communication du serveur. De cette façon, n'importe quel dispositif périphérique peut

vérifier que le service a été authentifié par la racine de confiance, mais la clef d'organisation critique ne sera pas exposée aux adversaires.

Si une clef est compromise, elle peut être révoquée en utilisant la clef principale de révocation pour authentifier la révocation.

Il va sans dire que toutes les clefs de la racine organisationnelle de la confiance sont essentielles à la sécurité de l'infrastructure. Ces clefs doivent être fortement protégées et utilisées uniquement par les membres internes de confiance de l'équipe principale des produits et sécurité IoT de l'entreprise. L'utilisation d'un module de sécurité hardware (HSM) approuvé pour stocker, accéder et utiliser les clefs est fortement recommandée.

Alors qu'un HSM peut souvent être une dépense importante au début du déploiement d'une technologie, les effets financiers à long terme sont très positifs. Plutôt que d'engager une dépense importante plus tard dans l'analyse judiciaire et l'ingénierie pour diagnostiquer et combattre un risque particulier qui aurait pu être résolu par une TCB et un HSM, qui représentent une dépense initiale relativement faible.

6.5.1 Risque

Le risque de ne pas utiliser une racine organisationnelle de confiance est que tout compromis avec une seule clef peut entraîner une compromission de l'ensemble de l'écosystème. En séparant l'organisation en une hiérarchie et en déployant des clefs séparées pour la hiérarchie, les clefs peuvent être cyclées à intervalles réguliers et en fonction de la priorité de l'application ou de la sous-organisation à laquelle la clef se rapporte. Cela crée une séparation des fonctions entre les différentes organisations de l'entreprise et réduit la possibilité pour une clef compromise de subvertir la sécurité de toute l'infrastructure.

6.6 Personnaliser chaque dispositif périphérique avant l'exécution

Les dispositifs périphériques doivent être activés avec des identités cryptographiques uniques pour garantir que les adversaires, les concurrents et les programmateurs amateurs ne peuvent pas usurper l'identité d'autres utilisateurs ou périphériques dans les environnements de production. Pour ce faire, le processus de personnalisation doit être effectué lors de la fabrication. Cela peut être fait soit par le fabricant de la solution TCB particulière, soit pendant le processus d'assemblage de la carte de circuit imprimée.

Pour résoudre le processus de personnalisation, procédez comme suit :

- Générer une clef cryptographique unique
- Signez la clef à l'aide de la clef de signature du dispositif périphérique organisationnelle (ou une dérivée de)
- Stocker la clef dans l'ancre de confiance de la TCB
- Générer (ou utiliser) un identifiant interne unique pour ce dispositif périphérique spécifique
- Stocker l'identifiant unique dans l'ancre de confiance de la TCB
- Enregistrer l'identifiant unique, la clef et la signature dans le système d'authentification back-end du service IoT

Notez que la personnalisation de la plate-forme du dispositif périphérique est différente de la personnalisation de l'identité du réseau. L'utilisation d'une UICC pour l'authentification du réseau est bénéfique pour de nombreuses raisons, et si possible, l'UICC pourrait être utilisé comme une ancre de confiance. Toutefois, si l'ancre de confiance du réseau ne peut être utilisé que pour l'authentification du réseau, la personnalisation de l'ancre de confiance des applications doit être effectuée séparément. L'unicité cryptographique de l'ancre de confiance des applications est requise pour s'assurer que la plate-forme d'applications est vérifiée avant l'exécution d’une application sur le dispositif périphérique.

En faisant un accord approprié avec un opérateur de réseau ou une autre partie émettrice, les cartes UICC peuvent parfois être provisionnées avant la livraison pour servir d'ancre de confiance centrée sur l'application. Dans un futur proche, les développeurs de dispositifs périphériques devraient évaluer si la technologie eUICC est adaptée pour être utiliser dans les produits et services IoT. Ces technologies permettront le provisionnement sur le terrain de secrets cryptographiques d'une manière similaire à une ancre de confiance axée sur l'application. Étant donné que l'industrie de la téléphonie mobile est le leader dans le processus de personnalisation et de provisionnement, il peut être avantageux d'utiliser l'eUICC comme une ancre de confiance.

En outre, ces technologies intégreront des capacités de provisionnement à distance et de canal sécurisé pour une communication sécurisée entre l'application et l'ancre de confiance intégrée dans une carte UICC. Ces fonctionnalités fourniront une personnalisation sur le terrain, ce qui réduira le coût global de personnalisation et de provisionnement pour chaque dispositif périphérique individuel.

Un court tutoriel sur l'utilisation des cartes UICC dans un écosystème de service IoT figure à l'annexe B.

Le défi vient avec la gestion des identités des dispositifs périphériques et le processus de signature. Chaque identité doit être cataloguée, avec des identifiants uniques correspondant à l'identité, dans un système qui ne peut pas être falsifié. Alors que le processus est généralement effectué à l'assemblage de la carte de circuit imprimée, une connexion de cette installation à l'entreprise doit être établie pour transmettre de manière sécurisée les données d'identité.

Le déploiement de cette solution peut être simple avec certaines installations plus familières avec la personnalisation cryptographique. D'autres installations de fabrication peuvent ne pas avoir un processus en place pour accomplir ceci. L'industrie mobile a été capable de réussir de cette manière en raison de sa capacité à contrôler la fabrication et la réalisation de technologies embarquées telles que l'UICC. Alors que l'industrie des communications mobiles est leader dans ce domaine depuis un certain temps, le processus de personnalisation et de provisionnement de l'application pour un dispositif périphérique n'en est qu'à ses débuts.

Soyez prêt à déterminer si l'identité du dispositif périphérique devrait (ou pourrait) être gérée par une passerelle ou une liaison montante. L'évaluation de l'architecture du produit ou du service IoT permettra de déterminer si cet attribut de gestion de l'identité aura une incidence sur le processus de personnalisation. Alors que la confiance peut être distribuée aux passerelles, l'organisation doit déterminer si la confiance peut être déléguée de manière

adéquate sans diminuer la sécurité globale du système de communication et d'authentification.

Les dépenses liées à la personnalisation comprennent généralement, mais sans s'y limiter :

- Mise en place du processus de personnalisation chez le fabricant de puces
- Coordination ou livraison des valeurs personnalisées uniques chez le fabricant et le fournisseur de services IoT
- Mise en place et gestion des identités personnalisées

6.6.1 Risque

Si l'entreprise choisit de ne pas implémenter la personnalisation du dispositif périphérique, elle risque de ne pas pouvoir différencier un dispositif périphérique d'un autre. Si toutes les clefs sont conformes à tous les systèmes de dispositifs périphériques, cela n'a pas d'importance si les numéros de série sont uniques. La raison en est que si des clefs sont extraites d'un seul dispositif périphérique, l'adversaire serait capable d'usurper l'identité de n'importe quel dispositif.

La personnalisation combat cela en forçant l'adversaire à extraire les secrets cryptographiques de chaque dispositif qu'il veut cloner ou usurper. Parce que le coût de ce processus peut être très élevé, la personnalisation à l'aide d'une ancre de confiance est la méthode la plus efficace pour lutter contre le clonage et l'usurpation d'identité.

6.7 Plate-forme d'exécution minimale viable (redéploiement d'une application)

Une plate-forme d'exécution minimale viable (MVeP) est la quantité minimale de travail qui doit être effectuée afin de créer un environnement d'exécution fiable pour se communiquer avec l'ancre de confiance. Typiquement, cela signifie :

- Configuration de l'horloge interne ou de l'oscillateur
- Configuration des périphériques de base (mémoire, stockage)
- Activation de divers ponts de connexion sur le hardware ou périphériques
- Authentification du bloc de code à exécuter par le processeur
- Exécution du prochain bloc de code
- Gérer le redéploiement et l'installation de l'image de l'application

Une fois que ce MVeP a été défini, le bootloader minimal peut utiliser l'ancre de confiance pour vérifier un bootloader plus robuste, ou peut exécuter le reste du bootloader après avoir vérifié les applications externes. Cela permet de définir un environnement cohérent avec un minimum d'effort qui authentifie les chaînes de code suivantes qui définiront la plate-forme d'applications.

Un autre avantage est qu'avec l'utilisation du modèle MVeP, même les processeurs avec une quantité minimale de mémoire NVRAM ou EEPROM interne peuvent amorcer une architecture sécurisée à l'aide d'une ancre de confiance interne ou externe.

Enfin, un MVeP est important pour revenir aux versions stables d'une plate-forme particulière. Si un MVeP peut être défini avec les fonctionnalités minimales requises pour vérifier l'intégrité des images du microprogramme de l'application et configurer l'environnement d'exécution, sa fonctionnalité peut être séparée de la fonctionnalité de l'application principale. Ainsi, si une mise à jour du micrologiciel échoue pour une raison quelconque, le MVeP peut toujours être utilisé pour se reconnecter au réseau principal et télécharger une autre image du micrologiciel (soit la même image, soit une image plus ancienne). Cela permet également aux dispositifs dotés de puces NVRAM détériorée de communiquer avec les services dorsaux et de soumettre des informations de diagnostic.

6.7.1 Risque

Bien que cela puisse paraître anodin, la définition d'un MVeP garantit que l'architecture du dispositif périphérique global vérifie de manière cryptographique chaque étape du processus de démarrage. Cette étape est essentielle pour garantir qu'un dispositif peut s'authentifier auprès du réseau et de ses homologues. Si le MVeP est mal conçu, il peut entraîner des failles de sécurité dans le processus de démarrage qui peuvent être exploitées par des adversaires, invalidant l'architecture de sécurité.

6.8 Provisionnement unique de chaque dispositif périphérique

Alors que la personnalisation garantit que chaque périphérique est unique une fois qu'il a été fabriqué, le provisionnement garantit qu'un périphérique unique est activé, mis à jour et associé à une identité de client particulière. Le processus de provisionnement permet de séparer les périphériques fabriqués à partir de périphériques achetés et / ou déployés dans un environnement IoT. Cela aide le fournisseur de services IoT à :

- Distinguer les dispositifs actifs et désactivés
- Associer des dispositifs périphériques à des réseaux ou à d'autres ressources liées à un client particulier
- Personnaliser un dispositif périphérique en fonction des besoins du client
- Plus facilement déterminer si un client particulier ou un dispositif périphérique a été compromis

Le processus de provisionnement ne se produit pas pendant la fabrication, mais repose sur le processus de personnalisation déployé pendant la fabrication. Le provisionnement se produit généralement sur le terrain, en fonction du client qui initialise le processus d'activation. Cependant, pour que le processus soit sécurisé, le provisionnement s'appuie sur les tokens de sécurité uniques définis lors du processus de personnalisation pour garantir que le dispositif périphérique unique est lié à un client unique. De cette façon, un adversaire ne peut pas enregistrer (ou désenregistrer) arbitrairement des périphériques simplement en devinant les détails du dispositif périphérique. Ils auraient, au contraire, besoin de chaque token cryptographique unique généré et mis en place au cours du processus de personnalisation, qui est infaisable d'un point de vue informatique.

De cette manière, le fournisseur de services IoT peut mathématiquement garantir qu'il est improbable que des adversaires puissent arbitrairement usurper ou enregistrer des périphériques à volonté. Cela conduit à un environnement IoT plus sûr et plus stable, où la relation entre les clients et les dispositifs peut être plus fiable.

6.8.1 Risque

Ne pas mettre en œuvre le processus de provisionnement peut entraîner une désynchronisation entre l'organisation et ses dispositifs périphériques. Il sera plus difficile pour l'organisation de suivre ses dispositifs et d'établir quels dispositifs ont été activés pour être utilisés dans l'écosystème ou déclassés. En outre, il peut être difficile d'établir quels dispositifs sont associés à des clients particuliers, ce qui augmentera la difficulté de repérer un dispositif problématique ou potentiellement compromis sur le terrain.

6.9 Gestion des mots de passe d'un dispositif périphérique

Les périphériques qui intègrent des interfaces utilisateur doivent être capables de gérer efficacement les mots de passe. Cela nécessite plusieurs choses :

- L'atténuation des attaques de force brute
- La désactivation de l'utilisation de mots de passe par défaut ou codés en dur
- Appliquer les meilleures pratiques pour les mots de passe
- Interdire l'affichage des informations d'identification de l'utilisateur sur les interfaces de connexion
- L'application de seuils et de retards progressifs pour les tentatives de mot de passe non valides

Les utilisateurs devront être protégés de l'attaque la plus simple possible : un autre utilisateur essayant de deviner leur mot de passe. Cela peut être atténué en annulant simplement le potentiel d'une attaque de force brute. Cela peut être fait en augmentant la limite de temps entre les tentatives de mot de passe. À chaque tentative de connexion échouée, le délai avant l'entrée du mot de passe suivant devrait être augmenté. Un plafond devrait être mis en place de telle sorte que pas plus de N tentatives peuvent être essayées à la fois. Sinon, une période de verrouillage raisonnable devrait être appliquée. L'utilisateur doit être averti de la tentative de force brute une fois que les données d'identification réelles sont utilisées pour rentrer au système.

Les mots de passe codés en dur ou par défaut ne devraient jamais être utilisés dans les systèmes IoT. Il ne devrait jamais y avoir de mot de passe administratif pour entrer dans un système. Il ne devrait jamais y avoir de compte privilégié avec des informations d'identification par défaut. Ceci est essentiel pour protéger les dispositifs des utilisateurs contre l'intrusion non autorisée par les pirates informatiques qui par hasard en surfant sur le web atteignent un dispositif avec une sécurité faible.

Les mots de passe doivent répondre à des exigences de qualité minimales représentatives des meilleures pratiques actuelles en matière de sécurité de l'information. Cela garantit que la force brute utilisée pour deviner un mot de passe sera difficile et aidera l'utilisateur à se prémunir contre le vol. Envisagez de revoir les directives OWASP ou SANS pour la sécurité des mots de passe afin de vous assurer que l'application est conforme aux meilleures pratiques récentes.

Les mots de passe ne doivent jamais être affichés sur l'écran d'un utilisateur. Toujours cacher le mot de passe avec le caractère astérisque, ou un autre glyphe facile à représenter sur l'interface graphique.

En outre, toutes les interfaces acceptant les mots de passe doivent utiliser la technologie d'atténuation par force brute. Il est également important que la technologie qui valide le mot de passe doit faire tout le procès depuis le début. Par exemple, JavaScript incorporé dans une page Web rendue sur un navigateur Web ne doit pas implémenter d'atténuation par force brute. N'importe quel attaquant web averti peut contourner ces contrôles en interagissant avec le serveur d'authentification back-end sur Internet. La technologie d'atténuation doit être implémentée côté serveur dans ce modèle. Dans les applications mobiles, où un code d'accès (PIN) ou un mot de passe local est intégré dans la zone de stockage sécurisée de l'application, le périphérique mobile doit atténuer les attaques par force brute dans cette interface.

En outre, après chaque tentative de mot de passe invalide, le système d'atténuation devrait augmenter le délai requis entre les tentatives autorisées. Il doit également y avoir un seuil maximum pour les tentatives de mot de passe non valides. Une fois ce seuil atteint, l'utilisateur doit être verrouillé en attente d'une authentification à deux facteurs ou d'un autre modèle plus invasif et difficile.

Ce processus est extrêmement simple à mettre en œuvre et nécessite très peu d'efforts de la part de l'équipe d'ingénierie.

6.9.1 Risque

Le risque de ne pas mettre en œuvre cette recommandation est :

- La possibilité pour les appareils volés d'être subvertis en devinant le mot de passe par force brute
- Les attaques Internet «Drive by» peuvent compromettre la sécurité des systèmes IoT en utilisant simplement des mots de passe codés en dur
- Les utilisateurs peuvent être compromis via la fonction de piquage de mot de passe (« lisant au-dessus de l'épaule ») si l'interface utilisateur affiche le mot de passe réel entré dans le système

6.10 Utiliser un générateur de nombres aléatoires prouvé

Déterminez si votre TCB est capable de générer un nombre vraiment aléatoire. Ceci est important car sans cela, le processus de vérification cryptographique peut être altéré, rendant les données cryptées plus faciles à deviner et affaiblissant l'intégrité des données.

Ceci est également extrêmement important pour la génération de clef cryptographique unique. Compte tenu d'un ensemble de conditions environnementales, un adversaire ne doit pas être en mesure d'influencer l'environnement pour amener une TCB à générer des nombres prévisibles pendant la génération de clef, la signature ou le remplissage de messages cryptographiques.

Ce processus est aussi simple que de déterminer si la TCB est capable de générer des nombres aléatoires approuvés FIPS [10], EMVCo [11] ou « Common Criteria ».

6.10.1 Risque

L'utilisation de la cryptographie sans générateur de nombres aléatoires est dangereuse pour plusieurs raisons. Bien que les raisons soient trop nombreuses pour être énumérées ici, il y a quelques faiblesses importantes à noter :

- La génération de clefs cryptographiques peut être compromise, entraînant la génération de clefs faibles ou prévisibles
- Les mots de passe ou clefs uniques peuvent être faibles ou prévisibles
- Le remplissage de message utilisé pour annuler le risque de reproduction d'un message peut être compromis

Ces problèmes peuvent entraîner des défaillances importantes de l'intégrité globale de la sécurité cryptographique de l'ensemble de l'écosystème IoT. Ce risque n'affecte pas seulement les dispositifs périphériques, il affecte l'ensemble du réseau.

6.11 Signer les images d'application cryptographiquement

Toutes les applications stockées en dehors de l'EEPROM principale d'un CPU doivent être cryptographiquement authentifiées. Pour ce faire, suivez simplement la procédure :

- Identifier les métadonnées représentant la version de l'image de l'application
- Générer un hachage cryptographique de l'image de l'application, y compris les métadonnées
- Valider que les métadonnées de l'application correspondent aux métadonnées internes
- Validez que la valeur de hachage correspond à la valeur interne à l'ancre de confiance
- Valider cryptographiquement la signature avec la clef de signature d'application
- Valider cryptographiquement que la clef de signature d'application a été signée par la racine organisationnelle

Ce processus est ordonné de telle façon que les activités les plus volatiles s'exécutent en premier, et les opérations les moins susceptibles d'échouer en dernier. De cette façon, la moindre quantité de travail est effectuée afin d'observer les risques les plus probables.

Ce processus est exceptionnellement facile à mettre en œuvre, en particulier lorsque la TCB est capable d'effectuer le gros du traitement au nom de l'application. Le vrai défi est plus subtil : c'est quand l'application effectue l'opération.

Une application qui n'a pas été vérifiée cryptographiquement ne peut pas effectuer l'opération, car elle n'a aucun moyen de savoir si son propre code a été corrompu par un adversaire. La modification du code dans la NVRAM est un moyen courant pour les attaquants de manipuler les systèmes embarqués, si ce dernier ne vérifie pas l'application.

Une application EEPROM interne doit, à la place, exécuter cette procédure en premier, sur toute image d'application dans un stockage persistant externe. Ensuite, cette application peut exécuter elle-même l'opération ou demander une application codée dans l'EEPROM interne pour effectuer ces types de tests en son nom.

6.11.1 Risque

Si l'image de l'application stockée dans le micrologiciel du dispositif périphérique (NVRAM) n'est pas cryptographiquement signée, le système ne pourra pas différencier le code autorisé et le code injecté par un adversaire. Cela pourrait permettre non seulement à un adversaire d'abuser du code exécutable pour manipuler un dispositif physiquement compromis, mais pourrait permettre à une entreprise rivale d'installer son propre logiciel sur un dispositif périphérique.

6.12 Administration des dispositifs périphériques à distance

Bien que tous les dispositifs périphériques ne requièrent pas une administration à distance, ceux qui le sont doivent être conçus de manière à garantir que les tiers ne puissent pas abuser des données d'identification administratives pour compromettre certains (ou tous) les dispositifs périphériques sur le terrain. La solution appropriée dépendra des capacités du dispositif périphérique. Cependant, les directives suivantes devraient être utilisées :

- Ne placez pas de composants cryptographiques privés dans un stockage non sécurisé dans les dispositifs périphériques, tels que les clefs privées SSH, les clefs privées TLS ou les mots de passe.
- Dans la mesure du possible, générez des tokens administratifs (clefs cryptographiques ou mots de passe) pour chaque dispositif périphérique
- Lorsque des mots de passe sont utilisés, imposer l'utilisation de mots de passe conformes aux meilleures pratiques concernant la complexité et la longueur du mot de passe
- Dans la mesure du possible, appliquer une authentification à deux facteurs pour les administrateurs
- Assurez-vous que l'utilisateur final est averti lorsqu'un administrateur accède à distance au dispositif périphérique
- Envisagez de restreindre l'accès administratif à un réseau privé virtuel (VPN)
- Ne pas intégrer de capacités d'administration à distance dans une application ou une API accessible au public, utiliser un canal de communication séparé
- Faire respecter la confidentialité et l'intégrité du canal de communication administratif
- Réduire le potentiel de relecture des commandes administratives en s'assurant que le protocole de communication a une entropie adéquate en utilisant un protocole de communication standard de l'industrie

6.12.1 Risque

Si vous ne définissez pas, ne mettez pas en œuvre et n'appliquez pas une stratégie sur l'administration à distance, vous risquez de compromettre les dispositifs périphériques aux accès frauduleux à distance. S'il n'existe pas de modèle de sécurité rigide pour l'accès super utilisateur aux périphériques, les adversaires peuvent être en mesure d'inverser la technologie ou d'extraire les clefs de sécurité des dispositifs périphériques qui donneront accès à chaque dispositif de l'écosystème. L'accès administratif est souvent l'une des premières technologies abusées par les adversaires dans les systèmes embarqués, car elles sont souvent mal configurées ou technologiquement faibles.

6.13 Journalisation et diagnostic

Afin d'évaluer les problèmes avec les dispositifs périphériques, le fournisseur de services IoT doit constamment évaluer le comportement du dispositif périphérique et déterminer s'il fonctionne dans l'ensemble des comportements approuvés. Pour ce faire, trois stratégies doivent être utilisées :

- Détection d'une anomalie
- Journalisation des dispositifs périphériques
- Diagnostic des dispositifs périphériques

Un dispositif périphérique doit consigner son propre comportement et télécharger de temps en temps ce journal dans les services dorsaux pour qu'il soit traité. Ce journal doit être composé d'une activité normale telle que les journaux du noyau du système d'exécution, les journaux d'application et d'autres métadonnées.

Les données de diagnostic doivent également être observées à intervalles réguliers et livrées au service de back-end soit avec ou séparément des journaux normaux. Les messages de diagnostic doivent inclure autant de données environnementales que possible sur le dispositif périphérique, notamment la température, l'autonomie de la batterie, l'utilisation de la mémoire, le temps d'exécution, les listes de processus (le cas échéant), etc. Cette information aidera à identifier quand - et quel (s) service (s) - est lié à un événement problématique ou anormal.

La détection d'anomalies dans le réseau devrait aider à détecter un problème qui ne peut pas être révélé par une analyse des journaux ou diagnostique. Il aidera également à classer les problèmes qui peuvent être observés dans les journaux ou les diagnostics, ou attribuera les problèmes à un composant spécifique qui réagit mal dans le monde physique. Par exemple, un module cellulaire qui se reconnecte constamment au réseau ou un capteur qui génère de mauvaises données.

Globalement, cette information permettra non seulement d'identifier si une faille dans la technologie est observée sur le terrain. Elle aidera également à déterminer si un comportement anormal est révélateur d'un événement de sécurité.

6.13.1 Risque

Si vous ne parvenez pas à mettre en œuvre la consignation et les diagnostics, l'entreprise risque de manquer des informations critiques. Cette information ne peut pas simplement avoir un impact sur la sécurité de l'écosystème, mais peut aider à diagnostiquer les défauts d'ingénierie critiques.

6.14 Appliquer la protection de la mémoire

Les systèmes embarqués sont souvent conçus avec des microcontrôleurs qui ne sont pas capables de technologie robuste tels que les unités de gestion de la mémoire (MMU) et les unités de protection de la mémoire (MPU). Cependant, ces technologies doivent être utilisées sur toute plate-forme qui veut :

- Exécuter des applications non privilégiées
- Exécuter des applications ou des applications non approuvées (tierces)

- Exécuter un émulateur ou une machine virtuelle dans un processus non privilégié

Tout environnement nécessitant une application non privilégiée doit être capable de se protéger des applications malveillantes ou compromises. Cela garantit que ces applications malveillantes ou compromises ne peuvent pas accéder aux zones de mémoire qui contrôlent les ressources privilégiées telles que la TCB, le driver de l'ancre de confiance ou les registres hardware des dispositifs périphériques.

Le défi dans ce domaine est souvent la migration d'une plate-forme de microcontrôleur de huit bits à une plate-forme plus robuste, comme un microcontrôleur 32 bits ou une architecture de processeur plus complète. Cependant, de nombreux systèmes d'exploitation sont disponibles gratuitement ou avec un droit de licence minime pour les systèmes embarqués qui implémentent correctement la protection de la mémoire avec un MPU ou un MMU.

6.14.1 Risque

Si ces technologies ne sont pas utilisées, les applications malveillantes ou compromises ne seront pas empêchées de modifier les ressources de base telles que les drivers, les registres périphériques ou même les services privilégiés tels que le noyau du système d'exécution et d'autres applications. Un manque de protection de la mémoire permet à toute application d'avoir un accès complet à la totalité de la mémoire intégrée dans le microcontrôleur ou le processeur. Les applications non privilégiées doivent être empêchées d'abuser de ces ressources.

6.15 Démarrage en dehors de l'EEPROM interne

La plupart du code du bootloader est intégré dans la mémoire EEPROM, interne à la CPU. Ce n'est pas toujours le cas, cependant. Déterminez si votre CPU charge son Bootloader à partir d'une source externe. Si le CPU n'a pas d'EEPROM lui permettant de vérifier le code Bootloader, il peut être manipulé par un attaquant local pour configurer le CPU pour son propre bénéfice.

En fonction du niveau de protection fourni à la puce ou à la région de mémoire hébergeant le Bootloader, un adversaire peut être capable d'utiliser un bus local (tel que le « Serial Peripheral Interface » (SPI)) ou une API distante (telle que que le micrologiciel OTA) pour manipuler le code embarqué. Cela se traduira par un adversaire capable de subvertir la plate-forme informatique en plaçant le code personnalisé au point d'exécution le plus fiable : la première étape du code exécutable. Une autre attaque pourrait être un adversaire échangeant simplement une puce Bootloader pour sa propre puce contenant des instructions personnalisées en dessoudant puis en soudant la nouvelle puce. Sans un moyen de vérifier l'intégrité du code externe, l'utilisateur sera incapable de faire la distinction entre les logiciels approuvés et non approuvés.

Afin de personnaliser un bootloader, un attaquant doit soit développer, soit externaliser le développement du bootloader. En fonction des ressources disponibles et du processeur cible, la difficulté de cette action peut varier énormément.

Envisagez d'utiliser une CPU ou un MCU / MPU avec une EEPROM interne ou une mémoire NVRAM verrouillée pour stocker le bootloader. Cela aidera à s'assurer que la

plate-forme peut au moins vérifier le premier exécutable chargé et exécuté par l'architecture, résultant en un périphérique plus fiable.

6.15.1 Risque

Ne pas évaluer la chaîne de confiance et appliquer une vérification d'intégrité pour le code initial chargé par la CPU peut entraîner une compromission complète du système. Cette étape est essentielle pour sécuriser le dispositif périphérique IoT et, par conséquent, l'écosystème.

6.16 Verrouillage des sections critiques de la mémoire

Les applications critiques stockées dans les régions exécutables de la mémoire, telles que les bootloader de premier niveau ou les bases de données sécurisées, doivent être stockées en « seule lecture ». Cela garantit que le périphérique peut être démarré dans une configuration valide sans interjection d'un adversaire. Sans cette assurance, le code exécutable chargé après la première étape d'exécution ne pourra pas confier qu'il a été démarré dans une configuration ou un état valide.

Même s'il est vrai que les adversaires peuvent encore subvertir le système en remplaçant ces sections critiques de la mémoire par leur propre code, cela les oblige à créer leur propre version personnalisée du logiciel, ce qui peut être un processus complexe et difficile. Cela augmente considérablement le coût global de l'attaque et la compétence requise pour réussir. En outre, si la personnalisation et le provisionnement sont utilisés, ces étapes forceront l'attaquant à recréer le processus pour chaque dispositif périphérique, en personnalisant sa solution aux caractéristiques cryptographiques uniques du système local. Cela rend l'attaque globale exceptionnellement coûteuse et diminue la faisabilité.

Pour éviter ce risque, identifiez simplement si la technologie qui stocke les sections critiques de la mémoire peut être verrouillée. Vous pouvez également démarrer avec une technologie EEPROM verrouillable.

Assurez-vous que si un code pour le verrouillage est utilisé, qu'il ne soit pas défini dans le logiciel. Les codes ou clefs définis par logiciel ne sont activés qu'après que le logiciel a exécuté la fonctionnalité correspondante pour engager le « verrou » qui protège le code. Il y aura une fenêtre de quelques millisecondes dans laquelle un adversaire peut abuser de l'état débloqué. Ainsi, les protections matériels, tels que les fusibles ou les bits de verrouillage, doivent toujours être utilisés dans la mesure du possible.

6.16.1 Risque

Sans une protection ou un état de seule lecture, les sections critiques de la mémoire peuvent être facilement modifiées par un adversaire. Cela peut leur donner suffisamment de privilèges pour compromettre la plate-forme des dispositifs périphériques entière sans autre action, en subvertissant tous les contrôles de sécurité suivants utilisés dans le système.

6.17 Bootloaders non sécurisés

La fonction d'un Bootloader consiste non seulement à configurer le processeur pour l'exécution d'une application primaire, mais aussi à charger et à transférer le contrôle d'exécution à l'application. Pour ce faire, le bootloader trouve et charge généralement

l'application principale dans la mémoire principale du processeur. Le problème se pose lorsque les bootloaders par défaut sont utilisés sur certains types de systèmes.

De nombreux bootloaders utilisés par les fournisseurs de microcontrôleurs, par exemple, permettent de charger le microprogramme externe dans la mémoire de la CPU pour l'exécution ou d'autoriser les mises à jour du microprogramme sur les interfaces série. D'autres bootlaoders peuvent inviter un utilisateur à rechercher des emplacements contenant des images d'application, ce qui permet à un utilisateur d'exécuter l'application de son choix.

Bien que cette fonctionnalité soit attendue dans un environnement tel qu'un ordinateur de table conventionnel, un ordinateur portable ou même un serveur, cela est inacceptable dans les systèmes embarqués. En effet, si un bootloader charge et exécute une application non vérifiée et non approuvée, il n'y a aucune garantie quant à la fiabilité ou à la sécurité de l'application exécutée, laissant l'état du périphérique embarqué dans un état incertain.

Par conséquent, pour corriger ce problème :

- Le bootloader doit être capable de vérifier cryptographiquement l'image de l'application à exécuter
- Le bootloader par défaut ou standard ne doit pas être utilisé s'il autorise des images alternatives ou un microprogramme sur une mémoire flash
- Le bootloader ne doit pas autoriser les images d'application chargées à partir d'emplacements de stockage arbitraires
- L'image exécutable du bootloader de premier niveau doit être verrouillée dans l'EEPROM et ne doit être mise à jour que par un processus sécurisé

De plus, la conception d'un bootloader devrait faire l'objet d'un examen par un analyste de sécurité tiers. Compromettre un bootloader par la manipulation de bogues dans le logiciel peut conduire à l'exécution de code personnalisé, ou à un contournement des contrôles de vérification d'intégrité. Cela peut entraîner un « jailbreak », ce qui peut ne pas être bénéfique pour l'entreprise. Assurez-vous que tous les bootloaders utilisés dans le système font l'objet d'un audit approfondi afin de déceler les failles de programmation pouvant entraîner des risques pour la sécurité.

6.17.1 Risque

Un bootloader non sécurisé peut être aussi néfaste qu'un processus d'amorçage mal conçu. La sécurisation du bootloader est une étape critique pour assurer l'intégrité du dispositif périphérique IoT.

6.18 Confidentialité de transmission parfaite (PFS)

PFS prend en charge la divulgation des clefs cryptographiques échangées lors de la configuration des communications entre deux dispositifs périphériques. Généralement, ils auront des certificats asymétriques utilisés pour authentifier leurs identités. À la fin de la phase d'authentification, une clef symétrique est générée et mutuellement acceptée en utilisant un chiffrement asymétrique pour protéger la négociation de clef. Une fois cette clef générée et acceptée, elle sera utilisée pour sécuriser le reste de la session entre les deux entités. Ceci est utilisé pour réduire la dépense de calcul impliquée dans la cryptographie

asymétrique. La cryptographie symétrique est moins coûteuse en termes de calcul, ce qui signifie à la fois plus rapide et moins gourmande en énergie dans les technologies embarquées ou de faible puissance.

Cependant, il y a un inconvénient. Ce modèle d'accord de clefs communes présume que les clefs asymétriques sont toujours gardées secrètes. Cela peut ne pas être le cas. À l'avenir, une entité suffisamment financée peut être capable de calculer la clef privée pour une clef asymétrique publique donnée. Si l'attaquant enregistre chaque session de communication entre une entité cible et ses pairs, l'entité sera alors capable de déchiffrer chaque message de communication du passé en générant la clef privée dans le futur.

En outre, la clef cryptographique d'un serveur peut être compromise par des tiers anonymes ou même par des initiés. Si cela se produit, toute personne qui a stocké des messages de communication sécurisés par la clef asymétrique volée peut maintenant déchiffrer ces messages.

Une solution à ce problème consiste à générer une paire de clefs asymétriques éphémères pendant le processus de négociation de clef. Seule la clef publique pour cette paire de clefs éphémères est transmise de chaque côté du lien de communication, elle peut être utilisée pour transmettre une clef symétrique.

Cette clef éphémère devrait être générée avec une entropie suffisante et une taille de clef suffisamment grande pour annuler le potentiel d'une attaque par épuisement informatique dans un délai raisonnable. Cela garantira que le processus de négociation clef est durable et moins susceptible d'être attaqué à l'avenir.

De plus, cette méthodologie garantit que les pairs utilisent leur clef asymétrique persistante uniquement pour l'authentification, et non pour la confidentialité et l'intégrité. Si cette clef asymétrique est volée ou exposée au public, cela n'affectera que le processus d'authentification, et la confidentialité et l'intégrité du canal de communication resteront protégées.

Pour rendre ce processus encore plus résistant aux attaques, la clef asymétrique utilisée pour l'authentification doit faire l'objet d'un processus de révocation sécurisé garantissant qu'un dispositif périphérique sera en mesure de vérifier si une clef a été compromise. Celui-ci ne doit plus approuver cette clef pour l'authentification s'il a été averti qu'un tel compromis s'est produit.

6.18.1 Risque

La non implémentation de PFS peut exposer toutes les communications réseau à un adversaire si cet adversaire accède à une clef privée utilisée pour sécuriser le canal de communication. À tout moment dans le futur, si l'adversaire capture la clef privée, toutes les communications capturées par l'adversaire dans le passé seront alors déchiffrées. Cela conduira à de graves conséquences.

6.19 Sécurité des communications des dispositifs périphériques

Bien que couvertes dans plusieurs autres recommandations et risques tout au long de ce document, il est important de noter succinctement que la sécurité des communications des

dispositifs périphériques est la plus grande menace pour eux dans l'IoT. La capacité d'un adversaire à manipuler le canal de communication est le moyen le plus simple pour qu'un dispositif périphérique soit compromis.

Par conséquent, les concepteurs de dispositifs doivent mettre en œuvre la sécurité des communications selon les perspectives suivantes :

- Authentification des homologues réseau
- Confidentialité des données
- Intégrité des messages

Bien que les messages en texte clair puissent être envoyés et reçus pour interopérer avec des dispositifs périphériques conçus par d'autres organisations, les données communiquées sur tout canal intégrant des commandes, des données de confidentialité ou des messages système critiques doivent être sécurisées. La première étape consiste à authentifier le périphérique homologue pour s'assurer qu'il est ce qu'il prétend être. Ceci est particulièrement important si l'homologue représente un service de système.

Ensuite, la confidentialité des données est requise pour s'assurer que les tiers ne peuvent pas lire les données critiques transmises par un canal de communication.

Enfin, l'intégrité des messages est requise pour s'assurer que les messages secrets n'ont pas été altérés par un adversaire.

Ces trois attributs, combinés ensemble, se traduira par un modèle de communication qui peut survivre pendant des années avec peu de changements d'ingénierie.

Ce processus est rendu beaucoup plus simple grâce à l'utilisation de protocoles de sécurité existants et bien analysés, tels que, mais sans s'y limiter :

- La dernière norme TLS approuvée
- La dernière norme DTLS approuvée
- SSH2 pour l'authentification et l'échange de clefs
- GBA pour la génération et l'échange de clefs
- OAuth2 pour l'autorisation
- BEST (« Battery Efficient Security »), sécurité efficace pour un système alimenté par batterie dans les dispositifs de communication de type machine à très faible débit (MTC) [21]

Alors que l'équipe d'ingénierie peut utiliser n'importe quelle bibliothèque de protocoles de communication qui répond aux exigences mentionnées, l'utilisation d'une solution standard réduira le nombre d'erreurs qui seront observées sur le terrain. En effet, les experts en sécurité de l'information et en cryptographie sont impliqués dans le développement de protocoles standardisés.

Les propriétés de sécurité de la technologie de communication cellulaire 3GPP, y compris les technologies de réseau LPWA normalisées NB-IoT et LTE-M, peuvent être trouvées dans le doument, GSMA PRD CLP.14 [4].

6.19.1 Risque

Même s'il va sans dire que la sécurité des communications est une exigence, il est parfois déroutant de savoir pourquoi c'est une exigence. La sécurité des communications ne garantit pas seulement qu'un adversaire ne puisse pas lire les données. Cela garantit également :

- Un dispositif périphérique ne peut pas être emprunté
- Un service critique ne peut pas être usurpé
- Les messages compromis peuvent être détectés
- Les modifications apportées aux configurations logicielles ou de sécurité peuvent être effectuées en toute sécurité

Sans la sécurité des communications, il n'existe aucune garantie quant à la qualité, la fiabilité ou la confidentialité d'un produit ou service IoT.

6.20 Authentification de l'identité d'un dispositif périphérique

Si chaque dispositif périphérique porte une identité cryptographique unique, telle qu'un numéro de série unique, le périphérique doit être en mesure de prouver qu'il représente réellement ce numéro de série. Pour ce faire, la TCB doit cryptographiquement signer un message avec une clef connue uniquement par elle-même et par le service back-end IoT, complexité qui peut être gérée avec des technologies telles que GBA. Le message doit contenir l'identité unique (numéro de série ou autre token) et les métadonnées associées au dispositif périphérique.

Le message devant être signé par la TCB doit également contenir un défi émis par le système back-end. Ceci annule la possibilité pour un adversaire de rejouer un message d'authentification déjà envoyé par la TCB au back-end. Si une entropie suffisante est contenue dans le défi, le potentiel de relecture du message est annulé.

Pour défier l'identité d'un dispositif périphérique :

- Recevoir une demande du dispositif périphérique contenant le token d'identité unique
- Générer un défi unique et l'envoyer au dispositif périphérique
- Recevoir la réponse au challenge du dispositif contenant la signature et le message
- Vérifiez que la signature est correcte à l'aide de la clef partagée
- Assurez-vous que le message signé contient le token d'identité correct et toutes les autres métadonnées pertinentes
- Reconnaître la signature vérifiée

Pour traiter un défi :

- Se connecter au système back-end
- Recevoir l'identité cryptographique du système back-end
- Authentifier de manière cryptographique l'identité du système back-end à l'aide de la TCB
- Envoyer un message contenant l'identité du dispositif et d'autres métadonnées à l'arrière-plan
- Recevoir un défi du back-end

- Générer un message contenant le token d'identité unique, les métadonnées et le défi
- Signer le message
- Envoyer le message et sa signature au back-end
- Vérifiez que le back-end a approuvé le message signé

6.20.1 Risque

Le risque de ne pas appliquer cette recommandation est que les dispositifs périphériques seront clonables ou vulnérables aux attaques d'usurpation d'identité. Cela peut ouvrir l'infrastructure de l'entreprise aux attaques des concurrents et des adversaires. Les concurrents peuvent utiliser un manque d'authentification d'identité des dispositifs périphériques pour créer une plate-forme concurrente à partir de la même nomenclature, mais à moindre coût.

Alternativement, un concurrent peut utiliser le manque d'authentification pour vendre du matériel qui se superpose à l'infrastructure de l'organisation. Ces problèmes peuvent entraîner une perte de revenus pour l'entreprise et une augmentation des dépenses d'exploitation, car le concurrent peut bénéficier de l'utilisation de l'infrastructure réseau de l'entreprise, même s'il ne paie pas pour l'utiliser. Étant donné que la bande passante du réseau a un coût quantifiable et que les serveurs Cloud, l'utilisation de capacités de traitement de données, l'utilisation du disque et d'autres ressources ont un coût quantifiable, ce genre d'activité parasitaire peut avoir un impact sérieux sur une entreprise vulnérable.

7 Recommandations d'haute priorité

Les recommandations à haute priorité représentent l'ensemble des recommandations à mettre en œuvre, mais uniquement si l'architecture des dispositifs périphériques l'exige. Par exemple, toutes les architectures n'ont pas besoin d'un boîtier de protection contre une intrusion physique. Ces recommandations devraient être évaluées pour déterminer si l'analyse de rentabilité les considère comme une exigence.

7.1 Utiliser la mémoire interne pour les secrets

Lorsque cela est possible, les processeurs doivent utiliser leur mémoire interne pour le traitement des secrets de base et des clefs de chiffrement non contenues dans une ancre de confiance. Cela garantira que si un adversaire surveille ou est capable de manipuler le bus mémoire, il n'obtiendra pas de secrets de base, mais ne verra que les effets de l'utilisation de ces secrets sur une application en cours d'exécution.

Ce modèle va assurer la longévité en ce qui concerne les secrets cryptographiques, car l'attaquant ne pourra pas découvrir ces secrets facilement. Au lieu de cela, l'attaquant devra s'appuyer sur la manipulation de bits dans la RAM qui correspondent aux effets de l'utilisation de ces secrets. Cela nécessitera que l'attaquant change les bits de la mémoire chaque fois que les secrets sont utilisés en interne, augmentant ainsi considérablement la complexité de l'attaque.

Tous les systèmes d'exploitation ne définissent pas de modèles d'utilisation de RAM interne pour le traitement des secrets. Par conséquent, il pourrait être nécessaire pour l'équipe d'ingénierie de mettre cela en œuvre. Bien que ce processus ne soit pas difficile, il n'est pas

trivial non plus. Le code exécutable doit s'assurer que ses routines de mémoire utilisent toutes des régions spécifiques garanties pour représenter la mémoire interne du processeur. Cela peut nécessiter un travail supplémentaire, en fonction du système d'exploitation et de la chaîne d'outils du compilateur utilisée.

7.1.1 Risque

La plupart des microprocesseurs et certaines CPU ont une petite quantité de mémoire SRAM interne dédiée au code qui s'exécute depuis l'EEPROM interne ou la NVRAM interne. Cette SRAM est généralement inaccessible aux périphériques externes, sauf si elle est délibérément exposée en utilisant une technologie telle que DMA. Si elles restent privées, les secrets cryptographiques traités par le code risquent beaucoup moins d'être exposés à des adversaires capables d'intercepter les communications RAM.

Bien que ce ne soit pas un risque élevé, les secrets cryptographiques ne devraient pas être transmis sur des bus accessibles au public, afin de réduire le risque d'attaque. Des adversaires bien équipés capables d'intercepter des communications RAM à des vitesses potentiellement élevées peuvent capturer des données telles que des secrets cryptographiques. Cependant, il faudrait un ingénieur qualifié qui sache appliquée une analyse inverse sur la technologie employer pour capturer des messages à travers la RAM qui pourraient être attribués aux opérations cryptographiques.

Par conséquent, bien qu'il s'agisse d'une recommandation importante, il n'est peut-être pas essentiel d'assurer la sécurité physique. Si les clefs cryptographiques de base sont stockées dans l'ancre de confiance et que seules les clefs de session sont traitées par l'application, le traitement des clefs dans la mémoire RAM externe n'entraînera vraisemblablement pas de compromis immédiat. Toutefois, cela suppose que l'architecture cryptographique limite les clefs exposées à celles qui ne sont pas essentielles aux opérations IoT de base, telles que la rotation des clefs, la génération de clef de session et la révocation des certificats.

7.2 Détection d'anomalies

La modélisation du comportement d'un dispositif périphérique est une partie essentielle de la sécurité de l'IoT. En effet, un dispositif compromis peut être difficile à distinguer par rapport à un dispositif se comportant normalement si seules les interactions réussies avec le périphérique sont consignées et analysées. Pour une perspective plus complète d'un environnement d'IoT, l'empreinte comportementale complète d'un dispositif doit être cataloguée pour identifier les anomalies qui peuvent être indicatives d'un comportement contradictoire.

Un comportement anormal d'un dispositif périphérique peut inclure :

- Des redémarrages erratiques ou des réinitialisations de dispositifs
- Quitter ou rejoindre un réseau de communication à intervalles irréguliers
- Une connexion à des services de dispositifs anormaux ou une connexion à des moments inappropriés a ces mêmes types de services
- Une empreinte de trafic réseau significativement différente de la normale
- Plusieurs messages mal formés envoyés du dispositif à ses homologues ou aux services back-end

Si le comportement normal d'un type de dispositif périphérique est catalogué par le fournisseur de services IoT, l'organisation sera en mesure d'identifier les comportements qui devraient indiquer un comportement anormal. En définissant une base de comportement, puis en surveillant continuellement les valeurs aberrantes potentielles, l'organisation peut diagnostiquer plus rapidement les problèmes de sécurité et de performances dans les environnements de production.

Le catalogage de l'empreinte comportementale peut également aider l'organisation à lier plus rapidement un ensemble de fonctionnalités défectueux à une caractéristique ou à une condition de l'écosystème IoT particulière. Cela peut conduire à des solutions d'ingénierie à un rythme plus rapide que si les données comportementales ne sont pas collectées.

7.2.1 Risque

Sans détection d'anomalie, la détection d'un dispositif périphérique compromis dans l'écosystème IoT peut prendre un temps excessivement long. Si le comportement anormal du dispositif périphérique n'est visible qu'en dehors des opérations normales, l'équipe administrative peut n'avoir aucune raison de se méfier du dispositif périphérique. Cependant, si la détection d'anomalie est mise en œuvre dans l'ensemble de l'écosystème, un comportement malveillant peut être détecté - et donc contenu - bien plus tôt.

7.3 Utiliser un boitier de produit inviolable

Le dispositif physique ne doit pas seulement être inviolable au niveau de puce, il doit également être inviolable au niveau du produit. Le boîtier utilisé dans le produit devrait offrir une protection contre les utilisateurs adversaires ou curieux. Cela peut être accompli de plusieurs façons :

- Les circuits qui invalident la mémoire NVRAM lorsqu'un boîtier est ouvert
- Les capteurs qui déclenchent les fusibles de sécurité quand la lumière est détectée
- Les capteurs qui déclenchent une alerte lorsque l'emplacement d'un dispositif physiquement statique est déplacé
- L'époxy couvrant les composants du circuit de base
- Les alertes déclenchées quand des composants internes ou amovibles sont supprimées de l'appareil

L'utilisation de ces méthodes peut améliorer la résistance au sabotage physique d'un dispositif périphérique. Cependant, il peut être plus rentable d'améliorer la conception du circuit lui-même. Bien que ces méthodologies aillent loin pour diminuer le potentiel de compromis par les amateurs bricoleurs ou adversaires, elles n'atténueront pas les analystes de sécurité bien équipés et expérimentés.

Ainsi, ces méthodes améliorent la capacité de l'entreprise à s'assurer que le produit lui-même ne peut pas être altéré alors qu'il est hors de la possession du consommateur qui l'a acheté. En d'autres termes, si un consommateur laisse son appareil à la maison ou sur le terrain, un adversaire doit non seulement avoir un accès physique pour compromettre l'appareil, mais aussi vaincre les contrôles de sécurité inviolables afin de modifier puis de remplacer l'appareil. Cela annule la possibilité de compromettre et de remplacer rapidement

les périphériques, ce qui constitue une amélioration précieuse de la sécurité des périphériques physiques.

Cependant, si le modèle de menace ignore cet aspect et se concentre sur la prévention contre l'attaque physique en général, y compris les attaquants avancés et équipés, il ne remédie pas complètement ce type de menace. Dans ce cas, ces solutions matérielles inviolables ralentiront un adversaire, mais n'arrêteront pas un adversaire avec le temps et l'expertise nécessaires.

Ainsi, un équilibre doit être atteint entre ce qui est rentable et le modèle de menace de l'appareil donné. Un guichet automatique (ATM) est un exemple d'un tel dispositif. La sécurité anti-sabotage dans l'emboîtage est nécessaire pour la sécurité de l'ATM, afin de s'assurer qu'un adversaire ne peut pas ouvrir et modifier physiquement l'ATM pour, disons, capturer des données de bande magnétique et enregistrer des numéros d'accès. Cependant, des adversaires avertis ont conçu des composants physiques, des « skimmers », qui doivent être adaptés à un ATM existant pour capturer les données des utilisateurs. Ainsi, l'inviolabilité physique ne peut atteindre qu'une partie du résultat souhaité. La conception de l'application et du matériel doit aller au-delà pour diminuer les attaques physiques.

Les ingénieurs et les chefs d'entreprise doivent évaluer le modèle de menace d'un produit ou d'un service donné et équilibrer le risque d'attaque avec les mesures anti-sabotage mises en œuvre dans l'appareil. Chaque type de résistance au sabotage entraînera des coûts, en fonction du procédé, de l'ingénierie et des matériaux impliqués. Et pourtant, l'effort peut ne pas aboutir au niveau de sécurité souhaité.

Un exemple de ce problème est l' utilisation d'époxy comme revêtement de puces. Bien que ce processus soit utile, un attaquant peut facilement faire deux choses pour contourner l'utilisation de l'époxy :

- Ponter les broches émanant du composant recouvert d'époxy
- Retirer l'époxy

Alors que l'époxy cache le composant de la puce, elle ne bloque pas et ne peut pas empêcher les électrons de traverser les circuits qui émanent de la puce revêtue d'époxy. Ainsi, si des secrets critiques sont communiqués sur des bus, l'époxy n'empêchera pas l'adversaire d'intercepter ces données.

De plus, l'époxy lui-même peut simplement être retiré. Au cours des dernières années, des techniques de bricolage maison sont apparues qui décrivent clairement une méthode pratique pour éliminer l'époxy d'un circuit en utilisant des produits chimiques et des processus prêts à l'emploi. Alors que le processus peut être caustique et potentiellement dangereux, les procédures décrites par des ingénieurs qualifiés sont solides et peuvent être mises en œuvre par toute personne ayant un laboratoire ou un bureau correctement ventilé.

Ainsi, une évaluation des risques doit être effectuée qui pèse clairement les avantages de la technologie inviolable avec la facilité du compromis. Si chaque dispositif doit simplement être sécurisé contre un adversaire souhaitant facilement manipuler ou abuser d'un dispositif aléatoire, une résistance à l'altération doit être utilisée. Si l'exigence est que les attaquants

avancés doivent être à l'abri de l'interception des messages sur les bus matériels, une architecture de sécurité plus résiliente pour l'application et le système d'exploitation doit être considérée comme une résistance à l'altération.

7.3.1 Risque

Comme indiqué dans la section précédente, le risque de ne pas déployer un dispositif IoT résistant au sabotage varie énormément avec les exigences de l'appareil. Si l'exigence est que le dispositif doit alerter l'utilisateur s'il a été ouvert, brisé ou modifié, la résistance au sabotage est importante. Si l'exigence est que l'appareil doit être protégé contre l'analyse par un amateur ou un chercheur spécialisé en sécurité ou un adversaire, la sécurité d'une l'architecture bien conçue est probablement la bonne résolution du risque.

Dans l'un ou l'autre cas, le risque de ne pas déployer une solution résistante à l'altération dans le cas est tel que l'utilisateur ne sera pas en mesure de déterminer si un adversaire a falsifié le dispositif physique. Même si cela ne signifie pas grand-chose pour les applications avec une architecture hardware et de sécurité pour les applications robuste, cela signifiera beaucoup pour les produits qui offrent des services critiques à ses utilisateurs tels que les dispositifs médicaux, les systèmes télématiques et les systèmes d'automatisation.

7.4 Application de la confidentialité et de l'intégrité à l'égard de l'ancre de confiance

Toutes les communications à destination et en provenance de l'ancre de confiance doivent être authentifiées et doivent garantir la confidentialité et l'intégrité. La seule exception à ce modèle est si l'ancre de confiance est interne au noyau du processeur. Toute ancre de confiance externe, telle qu'une UICC, ne peut être approuvée que si les messages reçus et envoyés peuvent être approuvés.

Pour ce faire, choisissez des ancres de confiance capables d'authentification et de cryptage et validez que tous les messages contenant des réponses aux défis sont envoyés de manière confidentielle et, si possible, avec une intégrité vérifiable.

Les UICC qui peuvent être gérés avec un canal sécurisé sont capables de confidentialité et d'intégrité. Le fournisseur de services IoT devrait discuter avec l'opérateur de réseau si la technologie de canal sécurisé UICC peut être utilisée pour fournir la sécurité de l'application. À l'avenir, les eUICC seront capable de sécuriser les applications. Le canal sécurisé peut ensuite être utilisé pour faciliter la sécurité de l'application du dispositif périphérique depuis l'étape du bootloader jusqu'à l'étape d'authentification du réseau.

Bien que cela semble être un exercice simple, il y a des subtilités à ce processus. Tester chaque aspect de la couche de communication est nécessaire. Certains messages provenant de diverses ancres de confiance peuvent ne pas être confidentiels ou garantir l'intégrité. Par exemple, un message qui indique si une opération a réussi ou échoué peut sembler bénin, mais doit être protégé pour s'assurer qu'un adversaire n'envoie pas une réponse personnalisée, en trompant l'application.

Certaines ancres de confiance peuvent ne pas être capables d'assurer l'intégrité dans le canal de communication. L'intégrité est préférée, et devrait être employée pour garantir qu'un message n'a pas été falsifié. Mais, cela nécessite une base de confiance à la fois sur

le processeur hôte et sur l'ancre de confiance, ce qui peut ne pas être raisonnable pour l'application.

Puisque tous les systèmes embarqués sont capables de faire des compromis avec un adversaire physique suffisamment équipé, il peut être exagéré d'exiger une racine de confiance sur les deux processeurs simplement pour les communications par bus local. Cependant, dans les applications où la sécurité physique est essentielle, l'intégrité doit être mise en œuvre.

7.4.1 Risque

Le risque de ne pas imposer la confidentialité et l'intégrité est intéressant à analyser. Ce risque peut aller d'un compromis au système complet à une collecte d'informations normale. C'est parce que certains messages peuvent être interceptés et corrompus. Par exemple, si une TCB demande que l'ancre de confiance vérifie l'intégrité d'un message, il transmettra le message via un bus matériel à l'ancre de confiance.

Si l'ancre de confiance est interne à la CPU, il est peu probable qu'un attaquant puisse modifier ce message sans équipement sophistiqué et coûteux. Cependant, si l'ancre de confiance est une puce distincte sur la carte de circuit imprimé, il peut y avoir une opportunité pour l'adversaire de modifier le message en permutant le circuit et en insérant leur propre matériel. Si, par exemple, l'ancre de confiance reçoit le message et répond simplement à la requête en indiquant , "Oui, ce message est valide" sans aucun test d' intégrité, la TCB sera incapable de vérifier si le message a été manipulé par un attaquant avec un accès physique au bus de données.

En outre, même si la réponse est vérifiée, un adversaire ayant un accès physique au bus peut simplement compromettre le circuit, absorber la demande de message de la TCB, émettre son propre message de confiance à l'ancre de confiance et laisser passer la vraie réponse à travers la TCB. Si le bus de communication hardware n'est pas correctement sécurisé, cette attaque est également possible, ce qui annule la capacité de l'ancre de confiance à effectuer son travail.

Cependant, attendre que le processeur et l'ancre de confiance aient des ancres de confiance internes individuelles crée un paradoxe. Comment un CPU amorçable peut-il se faire confiance si le CPU peut être remplacé par un adversaire, mais le CPU doit utiliser sa propre EEPROM pour vérifier l'intégrité de l'ancre de confiance! Cela crée une énigme, mais qui peut être résolue.

Une solution consiste à insérer une clef publique dans la mémoire morte du processeur. Cette clef peut être utilisée pour vérifier l'intégrité des messages envoyés par l'ancre de confiance. Si un message arbitraire (à vérifier) est transmis sur le bus matériel à l'ancre de confiance, l'ancre de confiance peut répondre avec un message signé qui inclut le message d'origine dans le cadre de la réponse. Cela vérifie que le message provient en fait de l'ancre de confiance, et que le message en cours de traitement est en effet le message qui devait être traité. La seule préoccupation restante serait de s'assurer que les nonces utilisés dans le remplissage des messages garantissent que les messages cryptographiques ne sont pas reproduits.

Compte tenu de ce qui précède, il est facile d'identifier que la cryptographie peut échouer en raison de problèmes très subtils, non seulement dans la cryptographie, mais aussi dans les algorithmes qui supportent les communications cryptographiques. C'est pourquoi la mise en œuvre de la confidentialité et de l'intégrité (correctement) est si importante.

7.5 Mises à jour de l'application à distance (OTA)

La mise à jour à distance de l'image d'une application d'un dispositif périphérique peut être un processus simple et direct. La complexité provient de la sur-ingénierie de la solution de manière à ne pas corriger les failles de sécurité réalistes. Du point de vue du stockage persistant, le processus d'ingénierie est très simple :

- Définir un emplacement pour l'image de l'application active
- Définir un emplacement pour l'image de l'application de sauvegarde (le cas échéant)
- Définir un emplacement pour l'image de l'application d'urgence
- S'il existe un espace d'image d'application de sauvegarde, mettez à jour cet espace avec l'image active
- Vérifier cryptographiquement l'image active en utilisant la signature stockée dans la TCB
    - Cela garantit que le support de stockage n'est pas corrompu et qu'un adversaire n'a pas modifié les bits pendant le processus d'écriture
- Télécharger la nouvelle image en entier ou en morceaux, ses métadonnées et signatures
- Corriger et compléter l'image active avec les morceaux aussitôt que reçus
- Vérifier la signature cryptographique à l'aide de la TCB
- Redémarrez avec la nouvelle image

Si le processus échoue à un moment donné, le système doit soit revenir à une image de sauvegarde pour s'assurer que l'application fonctionne, soit utiliser le système d'urgence pour informer l'écosystème de service IoT qu'une erreur s'est produite.

La difficulté vient de la création d'un modèle de stockage qui résout deux problèmes :

- Un attaquant essayant de manipuler le processus de mise à jour
- Une anomalie matérielle

Sans un système de sauvegarde ou une partition d'urgence, l'appareil n'aura d'autre choix que d'échouer. Étant donné que les systèmes embarqués ne disposent généralement pas d'interfaces utilisateur robustes, cela peut représenter un important point de stress entre l'entreprise et ses clients. Un échec aussi éloquent que possible est impératif non seulement pour la confiance de l'utilisateur, mais aussi pour la fiabilité du système.

Il est à noter que certains attaquants peuvent vouloir corrompre le processus de mise à jour dans le but de forcer un système dans un état de vulnérabilité persistante. Par exemple, si une vulnérabilité exploitable est détectée dans la version active de l'application, mais un correctif est disponible dans la version la plus récente de l'application.

L'avantage de ce modèle est que même si l'attaquant corrompt le processus de négociation du réseau, le système back-end a la possibilité de prendre note de cet événement. Si le réseau principal identifie qu'un dispositif communique normalement à l'exception des mises à jour, une alerte doit être émise pour l'administration afin de déterminer si ce dispositif périphérique est utilisé de manière abusive.

7.5.1 Risque

Si le processus de mise à jour de l'application OTA n'est pas correctement conçu du point de vue de l'architecture, des adversaires peuvent injecter à distance un code exécutable dans les dispositifs périphériques. Si l'adversaire a une position privilégiée sur le réseau, il peut potentiellement affecter des milliers de dispositifs à la fois. Le résultat de l'attaque peut aller de l'exécution de code simple à un déni de service (annulant la fonctionnalité des dispositifs périphériques), ou modifier complètement la fonction et le rôle du dispositif périphérique.

7.6 Authentification mutuelle mal conçue ou inexistante

Dans les environnements de communication, les pairs se parlent à travers leur identité caractéristique inclue dans le protocole. Cela signifie différentes choses dans différents contextes, mais dans chaque environnement, une adresse quelconque identifie la destination d'un message. Tout module de communication qui implémente un protocole donné est capable de dire qu'il est le propriétaire d'une adresse particulière. Même si une implémentation particulière d'un protocole est conçue, ou forcée, pour utiliser l'adresse matérielle d'un module radio local, aucune règle ne stipule qu'un utilisateur peut modifier physiquement l'EEPROM de ce module et changer l'adresse matérielle. Même si l'implémentation refuse de permettre à un utilisateur de modifier dynamiquement l'adresse matérielle, elle peut toujours être manipulée pour changer l'adresse. Le résultat de cette fonctionnalité est, essentiellement, l'usurpation d'identité : ou l'acte consistant à prendre l'identité d'un autre ordinateur dans le but d'intercepter des messages destinés à cet ordinateur.

7.6.1 Authentification du client

Tous les environnements sont vulnérables à l'usurpation d'identité. Par exemple, n'importe quel terminal avec une transmission radio cellulaire peut signaler qu'elle est propriétaire d'une identité IMSI (« International Mobile Subscriber Identity ») donnée, qu'elle soit vraie ou non. N'importe quel ordinateur portable peut changer son adresse Ethernet, usurper l'identité d'autres ordinateurs sur le réseau local (LAN). Indépendamment du fait que la topologie traverse un espace physique ou un espace d'onde, l'identité d'un dispositif périphérique de communication peut être usurpée.

La protection contre cela est l'authentification. Par exemple, dans le réseau cellulaire, toute personne disposant du bon équipement peut prétendre détenir tout IMSI de son choix. Cependant, les opérateurs cellulaires imposent l'authentification en codant une clef cryptographique dans le module d'identité d'abonné (SIM) qui est unique pour cet abonné (IMSI). Lorsqu'un dispositif cellulaire communique avec une station de base en indiquant qu'il représente un IMSI particulier, la station de base émet un défi cryptographique qui ne peut être résolu que par une personne possédant la clef cryptographique unique stockée dans la carte SIM fournie pour cette identité particulière. Si l'attaquant ne parvient pas à

résoudre le défi cryptographique, la station de base peut vérifier que l'attaquant ne représente pas l'identité IMSI en question et peut interdire à cet utilisateur de s'associer au réseau.

Le modèle décrit ci-dessus décrit l'authentification basée sur le client. C'est le modèle dans lequel le sous-système du serveur (y compris les stations de base) permet aux clients (dispositif périphérique) de rejoindre et de quitter le réseau tant que les clients peuvent authentifier cryptographiquement leur identité. Cependant, il existe un problème inverse qui expose les clients à une manipulation : l'authentification du serveur.

7.6.2 Authentification du serveur

Dans le modèle 3GPP, seuls les dispositifs périphériques (appelés équipements utilisateur dans 3GPP) sont authentifiés. Les dispositifs n'authentifient pas les stations de base auxquelles ils se connectent. Ainsi, n'importe quelle station de base peut prétendre servir au nom de n'importe quel opérateur cellulaire. Les personnes capables de manipuler ou de construire une station de base cellulaire peuvent ensuite usurper l'identité de n'importe quel réseau commercial. Une station de base cellulaire personnalisée coûte actuellement moins de 1 000 USD à construire, mais la puissance résultante ne permet que l'interception des messages dans la zone locale. Une fois la fausse antenne est mise en place, la station de base peut emprunter l'identité d'un opérateur cellulaire local et intercepter des appels téléphoniques, des messages texte et même des données à partir des dispositifs périphériques dans la zone locale.

Les nouveaux protocoles de réseau 3GPP, tels que UMTS et LTE, imposent l'authentification mutuelle des deux entités. Cela permet aux dispositifs périphériques de vérifier cryptographiquement que la station de base sert pour le compte de la porteuse cellulaire qu'elle prétend desservir. Un adversaire doit maintenant casser la cryptographie de l'opérateur cellulaire pour usurper l'identité d'une station de base, ce qui augmente considérablement la complexité, la difficulté et le coût d'une attaque.

7.6.3 Systèmes d'interrogation cellulaire ou fausses stations de base

Cependant, il existe des exceptions à cette règle, telles que les systèmes d'interrogation cellulaires. Ces dispositifs, généralement utilisés par des acteurs gouvernementaux, les gouvernements et les services de renseignement, sont codés avec des clefs cryptographiques fournies à ces entités par certains opérateurs cellulaires, à des fins de sécurité nationale. Ces systèmes utilisent ces clefs pour intercepter de manière passive des communications bidirectionnelles ou pour effectuer activement des attaques de type « man-in-the-middle » contre des cibles spécifiques.

Cependant, dans le modèle moderne des menaces liées aux communications, l'accès à cette technologie ne se limite pas aux acteurs du gouvernement et des services de renseignement. Aujourd'hui, ces systèmes peuvent être construits à partir de composants qui ne coutent que plusieurs centaines de dollars américains, ce qui se traduit par une fausse station de base rentable capable d'intercepter ou d'usurper l'identité des communications cellulaires.

7.6.4 La sécurité des communications est la sécurité « porte à porte »

La description des systèmes d'interrogation cellulaires permet de résumer assez bien cette section en évoquant l'idée que la sécurité des communications n'est pas absolue. Elle protège uniquement le canal de communication entre deux entités. Ces entités, cependant, agissent comme des portes permettant aux données de passer et traverser les écosystèmes auxquels ces entités sont connectées.

Par exemple, une carte SIM particulière peut être fournie pour une utilisation dans un système de contrôle industriel tel qu'un dispositif de surveillance de puits de pétrole. Une carte SIM, étant donné sa conception, est un composant amovible. Toute personne ayant un accès physique au dispositif de surveillance de puits de pétrole peut extraire la carte SIM et la placer dans un ordinateur portable. Si l'ordinateur portable a un logiciel qui peut simuler la fonctionnalité du dispositif de surveillance, le serveur principal sera incapable de faire la différence entre le dispositif réel et l'ordinateur portable. Pourtant, l'ordinateur portable sera authentifié sur le réseau cellulaire à cause de la carte SIM! Ainsi, le réseau cellulaire a authentifié la carte SIM, mais pas l'ordinateur portable.

7.6.5 La solution de l'authentification mutuelle

Chaque pair d'un écosystème IoT doit authentifier tous les autres pairs qui participent à cet écosystème. Pour ce faire, une TCB doit être utilisée pour s'assurer que l'architecture cryptographique appropriée est le moteur de la technologie de communication. L'authentification mutuelle ne peut pas se produire si les clefs sont facilement exposées aux adversaires. Consultez la section qui décrit la TCB de ce document pour plus d'informations.

Une fois authentifié, chaque homologue doit crypter et signer les messages envoyés aux autres homologues du réseau. Chaque homologue qui reçoit un message doit valider de manière cryptographique les données avant d'agir sur elles. Comme tous les protocoles de communication ne sont pas capables de s'authentifier mutuellement, ou ont une cryptographie forte, il est impératif que l'ingénieur d'application conçoive un protocole suffisant qui impose la confidentialité et l'intégrité, plutôt que de s'appuyer sur le protocole de communication.

Des protocoles encore plus robustes qui intègrent l'authentification mutuelle, tels que LTE, n'abordent pas la sécurité de l'infrastructure au-delà du réseau de communication cellulaire. Seule une couche supérieure de protocoles de sécurité peut traiter le risque d'une sécurité insuffisante dans l'infrastructure au-delà du contrôle de l'opérateur cellulaire.

7.6.6 Risque

Le risque de ne pas adhérer à une sécurité forte au niveau des applications est que le dispositif périphérique doit faire confiance à la sécurité de la couche de communication. Comme indiqué dans cette recommandation, il peut ne pas être suffisant de faire uniquement confiance au réseau pour résoudre les problèmes de sécurité dans l'application. Même si l'on peut faire confiance à l'ORM (Opérateur Réseau Mobile), les messages peuvent passer par plusieurs éléments d'infrastructure réseau non détenus ou contrôlés par l'ORM avant que les données ne parviennent aux serveurs appartenant au fournisseur de services IoT. Par conséquent, le fournisseur de services IoT risque que quiconque contrôle

ces systèmes intercepter, modifier ou fabriquer des messages depuis ou vers des systèmes de dispositifs périphériques.

7.7 Gestion de la confidentialité

Un aspect impératif de la technologie IoT est leur capacité à connecter le monde physique au monde numérique. Le résultat de ceci est une lacune dans la vie privée, car l'environnement physique de l'utilisateur est directement associé avec les choses qu'ils aiment et qu'ils cherchent sur le web. Cela peut provoquer des effets indésirables au fil du temps.

Par conséquent, il est important que les fournisseurs de services IoT tiennent compte de la vie privée de leurs clients et développent des interfaces de gestion de la confidentialité intégrées à la fois dans les dispositifs périphériques ou terminaux, si possible, et à l'interface Web du produit ou du service.

Cette technologie devrait permettre à l'utilisateur de déterminer quels attributs de sa vie privée sont utilisés par le système, quelles sont les conditions d'utilisation et la possibilité de désactiver l'exposition de cette information à l'entreprise ou à ses partenaires. Ce système de granularité et d'exclusion contribuera à garantir que les utilisateurs ont le droit et la capacité de contrôler les informations qu'ils partagent sur eux-mêmes et sur leur monde physique.

7.7.1 Risque

Les risques potentiels de ne pas protéger la vie privée des consommateurs sont nombreux. Les problèmes liés au harcèlement criminel, à l'intimidation, à l'établissement de profils faux, aux menaces et autres sont des résultats réalistes et pratiques de ne pas protéger les données de l'utilisateur.

7.8 Confidentialité et identités de dispositifs périphériques uniques

Chaque dispositif périphérique est connu numériquement par une empreinte digitale. Cette empreinte est composée d'adresses, de numéros de série et d'identités cryptographiques propres au dispositif spécifique. Cependant, ces tokens peuvent également associer directement un dispositif à un client, un emplacement ou un service particulier. Dans de nombreuses situations, cela n'est pas souhaitable. Par exemple, les smartphones peuvent être suivis car l'adresse Wi-Fi intégrée dans le terminal a été utilisée lors de la recherche active des points d'accès 802.11. Ces adresses pourraient alors être suivies pendant leur parcours d'un réseau à un autre. Cela permettrait à toute personne en mesure d'associer une adresse Wi-Fi particulière avec un utilisateur particulier, de surveiller leurs déplacements dans le monde entier. Pour lutter contre cela, les fabricants de logiciels de smartphone ont généré des adresses de clients Wi-Fi aléatoires lors de la recherche de points d'accès, ce qui rend presque impossible le suivi des téléphones de cette manière.

Les dispositifs périphériques IoT peuvent être suivis de la même manière grâce aux adresses « Bluetooth Low Energy » (BLE), aux adresses 802.15.4, au Wi-Fi ou même à l'IMSI cellulaire. Dans la mesure du possible, le fournisseur de services IoT devrait développer sa technologie de dispositif IoT de manière à utiliser une adresse radio aléatoire

pour se connecter à de nouveaux environnements, permettant que les données de la vie privée de l'utilisateur restent confidentielles.

Cela est également vrai pour les clefs cryptographiques, telles que les clefs publiques SSH. Alors que les utilisateurs veulent généralement que leurs clefs publiques soient connues du public, les clefs publiques cryptographiques sur les dispositifs périphériques exposeront l'identité de l'utilisateur d'un dispositif particulier, ce qui n'est pas souhaitable. Au lieu de cela, l'utilisateur devrait être en mesure de choisir s'il veut que son identité soit connue lorsqu'il se connecte à un nouvel environnement.

7.8.1 Risque

Si ce risque n'est pas atténué de manière adéquate, les utilisateurs ayant des dispositifs mobiles pourront être suivis au fur et à mesure que leurs terminaux rejoignent de nouveaux réseaux. Cela ouvre d'importantes lacunes en matière de confidentialité que les équipes juridiques, les législateurs et même les compagnies d'assurance analysent actuellement. Une mise en œuvre inappropriée de la confidentialité pour réduire le potentiel de suivi pourrait ouvrir un nouveau fournisseur de services IoT à des conséquences juridiques dans un proche avenir.

7.9 Exécuter des applications avec des niveaux de privilège appropriés

Les applications exécutées sur un dispositif périphérique ne nécessitent généralement pas de privilèges de super-utilisateur. Le plus souvent, les applications nécessitent un accès aux pilotes de périphériques ou à un port réseau. Alors que certains de ces périphériques, ports ou autres objets peuvent nécessiter des privilèges de super-utilisateur pour y accéder initialement, les privilèges de super-utilisateur ne sont pas requis pour effectuer des opérations ultérieures. Ainsi, il est recommandé d'utiliser uniquement les privilèges de super-utilisateur au début de l'application pour accéder à ces ressources. Ensuite, les privilèges de super-utilisateur doivent être supprimés.

La suppression des privilèges de super-utilisateur est un processus commun qui est bien documenté et qui a été implémenté exceptionnellement bien dans des applications telles que « Secure Shell » (SSH), Apache2 et d'autres serveurs bien conçus. Le processus comprend généralement :

- Démarrage de l'application avec des privilèges élevés
- Accès à toutes les ressources nécessitant des privilèges élevés
- Identification d'une identité d'utilisateur (par exemple, ID utilisateur UNIX et ID de groupe) que l'application doit utiliser pour ce profil
- Changement complet de l'identité du processus à l'ID utilisateur / groupe cible, supprimant ainsi les privilèges de super-utilisateur de l'application en cours d'exécution

Un modèle plus complexe peut être vu dans l'implémentation SSH de *privsep*, qui exécute un service privilégié dont le seul but est d'amorcer l'application principale sous une identité d'utilisateur / groupe cible. De cette façon, si le service se termine, il peut être redémarré facilement sans compromettre les ressources privilégiées.

Pour plus d'informations, voir « SSH Privilege Separation » :

http://www.citi.umich.edu/u/provos/ssh/privsep.html

7.9.1 Risque

L'exécution d'applications avec des niveaux de privilèges élevés peut entraîner une compromission complète du système si une seule application est compromise. Puisque les privilèges de super-utilisateur accordent à une application un accès complet à l'ensemble du système en cours d'exécution, il n'y a aucun moyen de contenir un adversaire une fois qu'il a compromis une telle application. La suppression des privilèges aide à contenir l'adversaire et limite sa capacité à augmenter ses privilèges dans le système embarqué. Cela peut être la différence entre un compromis système complet et un ennui mineur.

7.10 Application de la séparation des tâches dans l'architecture d'application

Les applications exécutées sur un dispositif périphérique doivent avoir des identités d'utilisateur différentes associées à chaque processus unique. Cela garantit que si une application est compromise, une application distincte sur le même dispositif ne peut pas être compromise sans une deuxième attaque qui réussisse. Cette étape supplémentaire requise pour le compte d'un attaquant est souvent un obstacle majeur au processus global de développement d'exploit et augmente le coût et la complexité d'une attaque contre un dispositif périphérique.

Par exemple, un service réseau qui permet à un utilisateur de récupérer des informations sur l'état du dispositif ne doit pas également être en mesure de manipuler la TCB sur le même processus. Cette capacité serait hors de portée par rapport à l'objectif du service. Ces deux opérations doivent être traitées dans des applications différentes et exécutées sous des ID utilisateur différentes sur le système d'exploitation local, ce qui permet de séparer les tâches de l'application et de réduire le risque d'abus si un composant est compromis.

Pour l'implémenter correctement, la protection de la mémoire doit être activée dans l'architecture matérielle sous-jacente et le système d'exploitation doit avoir un concept de niveaux de privilège. Les logiciels non privilégiés doivent être interdits d'accès aux ressources privilégiées, telles que les pilotes, les fichiers de configuration ou d'autres objets.

Les services doivent effectuer des demandes d'accès aux ressources privilégiées, mais via une API avec des caractéristiques limitées, telle qu'un appel système, pour s'assurer que tous les messages sont bien formés et correspondent aux exigences de l'architecture de sécurité.

Le concept de multi-niveaux de privilège est un concept vieux d'un demi-siècle. Cependant, dans les systèmes embarqués, il est souvent négligé car les utilisateurs ne sont pas autorisés à se connecter à la console et à exécuter leurs propres applications. Par conséquent, tous les services sont souvent déployés en tant qu'utilisateur privilégié. Cependant, ceci n'est pas désirable.

Chaque application ou service doit être implémenté en utilisant un privilège personnalisé. Dans la plupart des environnements, il s'agit d'une identité d'utilisateur distincte. Cette séparation des tâches en imposant des identités d'utilisateur différentes garantit que si un service est compromis, il ne peut pas affecter directement les ressources utilisées par un

autre service sur le même système. Pour compromettre d'autres services et utilisateurs, des exploits secondaires doivent être trouvés dans le système d'exploitation local pour élever les privilèges.

Cela nécessite une planification et une architecture d'application solide qui utilisent correctement la séparation des privilèges.

7.10.1 Risque

Si une séparation des tâches n'est pas appliquée, toute compromission d'un seul service sur le dispositif périphérique entraînera une compromission de l'ensemble du périphérique, car chaque service ou application exécuté partagera la même identité d'utilisateur et / ou de groupe. Si la recommandation est mise en œuvre, un service à faible privilège compromis sur le réseau n'entraînera pas immédiatement une compromission de l'ensemble du système.

Cette recommandation étant simple à mettre en œuvre, elle est essentielle à la sécurité des dispositifs périphériques IoT. Il convient de noter qu'il faut souvent une grande quantité d'expertise pour compromettre à distance un service réseau. Si l'adversaire doit également élever des privilèges en implémentant un exploit au niveau du noyau, ou un autre exploit secondaire, pour prendre le contrôle du système complet, l'adversaire peut ne pas avoir le temps, les compétences ou l'équipement pour exécuter l'attaque.

Augmenter la difficulté d'une attaque avec un simple changement de configuration tel que celui-ci contribuera grandement à assurer la longévité de la solution IoT.

De plus, étant donné que des services compromis peuvent être détectés à travers la surveillance des processus et d'autres analyses, tout compromis de service peut alerter l'écosystème de services qu'une attaque de périphérique a été détectée. Cela permet aux administrateurs d'agir pour sécuriser le système avant qu'un compromis général sur le système soit réussi. Cela permet également aux administrateurs de diagnostiquer et de corriger le logiciel vulnérable avant un compromis plus sérieux. Cela donne à l'entreprise un avantage significatif contre les attaquants qualifiés.

7.11 Appliquer la sécurité au niveau du langage de programmation

Les langages de programmation ont différents degrés de sécurité, en fonction du but et du niveau cherchés pour une implémentation particulière. Certains langages fournissent des constructions permettant de limiter l'accès à la mémoire brute et imposent des contraintes sur l'utilisation de la mémoire. L'équipe d'ingénierie doit identifier un langage capable de fournir une sécurité à l'exécution de l'application ou au binaire qui en résulte.

Le compilateur ou le temps d'exécution doivent être sécurisés, dans la mesure du possible, afin de limiter le risque d'abus d'une vulnérabilité par un adversaire. Dans un environnement d'exécution bien défini, même un défaut de programmation facile à déclencher peut-être extrêmement difficile à exploiter pleinement. Cela suppose que les améliorations de sécurité sont utilisées pour protéger la manière dont l'application s'exécute, accède à la mémoire et est prise en charge par les améliorations de sécurité du système d'exploitation.

7.11.1 Risque

Le risque de ne pas choisir le langage de programmation adéquat pour un système IoT est que l'application résultante sera facile à exploiter. Certains systèmes de programmation tels que PHP sont pas très fiable et ne devraient jamais être utilisés par une équipe d'ingénieurs professionnels. D'autres langages, tels que Python, conviennent aux environnements de production, mais présentent des risques de sécurité subtils qui doivent être évalués. Ainsi, la volatilité du risque résultant peut aller d'un niveau critique à un niveau acceptable. L'équipe d'ingénierie doit utiliser le processus d'évaluation des risques et de modélisation des menaces pour évaluer suffisamment quelle langue est la meilleure pour son environnement de production.

7.12 Implémenter des audits et analyses persistantes de sécurité (« pentesting »)

L'exécution d'un audit de sécurité uniquement au moment du déploiement n'est pas suffisante pour la plupart des déploiements IoT où de nouveaux dispositifs périphériques peuvent être publiés sur le terrain et configurés à tout moment. Il est recommandé d'utiliser une approche « pentesting » persistante afin de détecter rapidement les logiciels dans les dispositifs vulnérables et les configurations non sécurisées.

La mise en œuvre d'une stratégie d'analyse persistante peut permettre une détection rapide et une gestion précoce des menaces identifiées, en augmentant la rapidité de la correction d'erreurs et en réduisant la période d'exposition à la menace.

Une stratégie d'analyse persistante complète devrait fournir une méthode automatique et programmée : découverte d'objets pour créer un inventaire des objets accessibles, identification et analyse des objets, vérification et exploitation des vulnérabilités connues, vérification des configurations non sécurisées et rapports appropriés et alertes qui devraient aider à la correction.

7.12.1 Risque

Le risque de ne pas mettre en œuvre une stratégie d'analyse persistante est que les audits de sécurité ne peuvent être exécutés qu'une seule fois au moment du déploiement, mais que les nouveaux points de connexion et configurations ne sont jamais évalués. Cette situation peut conduire à un ensemble de dispositifs périphériques IoT vulnérables qui ne sont jamais identifiés comme exposés tant qu'ils ne sont pas compromis par un attaquant.

8 Recommandations de priorité moyenne

L'ensemble de recommandations de priorité moyenne englobe l'ensemble de recommandations pertinentes en fonction des choix de conception de la technologie des dispositifs périphériques. Par exemple, l'application des améliorations de la sécurité au niveau du système d'exploitation n'est valide que si un système d'exploitation s'exécute sur le dispositif périphérique. Si le dispositif est composé d'une application de noyau monolithique ou d'un système d'exploitation temps réel (RTOS) intégré avec une seule application intégrée, la recommandation peut ne pas s'appliquer. Lorsque des recommandations s'appliquent à la conception du dispositif périphérique, elles doivent être implémentées.

8.1 Appliquer les améliorations de sécurité au niveau du système d'exploitation

Les applications exécutées sur un système d'exploitation doivent être conçues pour utiliser (de manière transparente ou intentionnelle) les améliorations de sécurité du système d'exploitation et du noyau sous-jacents. Cela inclut des technologies telles que :

- ASLR
- Mémoire non exécutable (pile, tas, BSS, ROData, etc.)
- Protection de Déréférencement de l'indicateur de l'utilisateur (UDEREF)
- Protection contre la divulgation d'informations sur la structure (« Structure Leakage »)

Chaque système d'exploitation utilisé dans un système embarqué fournira différentes variantes et combinaisons de ces technologies, parfois sous des noms différents. Déterminer ce que le système d'exploitation et le noyau sont capables de fournir, et activer ces technologies, si possible, pour améliorer la sécurité des applications.

Le défi vient d'identifier ce dont chaque système d'exploitation est capable. Par exemple, les applications s'exécutant sur des plates-formes qui n'ont pas d'unité de gestion de la mémoire (MMU) peuvent ne pas être capables d'ASLR. Cependant, l'équivalent de UDEREF peut être appliqué même dans des environnements avec seulement une unité de protection de mémoire (MPU). Évaluer quelle technologie est utilisée et quelles sont ses capacités, et déterminer quel niveau de sécurité peut être atteint grâce à la combinaison de l'architecture, du noyau, du système d'exploitation et des protections des applications.

8.1.1 Risque

Si cette recommandation n'est pas appliquée, l'environnement d'exécution de l'application sera nettement plus facile à exploiter. Ces améliorations limiteront considérablement le nombre d'adversaires capables (le cas échéant) de développer un exploit fiable pour un service vulnérable.

Ainsi, si une application développée par l'organisation présente une faille de sécurité susceptible d'être exploitée pour obtenir des capacités d'exécution de code à distance, le risque d'abus peut être annulé en appliquant les technologies ASLR, NX, UDEREF et autres. Cela limitera la possibilité pour un attaquant de développer un exploit dans un laps de temps raisonnable, car le développeur d'exploits devra utiliser des techniques avancées et complexes qui doivent être personnalisées par rapport à chaque cible individuelle. Cela augmente non seulement la difficulté, mais le temps et les dépenses nécessaires pour réaliser un exploit pleinement opérationnel.

Sans ces améliorations, un exploit pleinement opérationnel peut être développé en utilisant des logiciels prêts à être utiliser et disponibles gratuitement en quelques heures.

8.2 Désactiver les technologies de débogage et de test

Lorsqu'un produit est en cours de développement, il est souvent activé avec des technologies de débogage et de test pour faciliter le processus d'ingénierie. C'est entièrement normal. Cependant, lorsqu'un périphérique est prêt pour le déploiement de

production, ces technologies doivent être supprimées de l'environnement de production avant la définition de la configuration approuvée.

La configuration approuvée avec laquelle un produit est déployé ne doit jamais contenir d'interfaces de débogage, de diagnostic ou de test susceptibles d'être utilisées abusivement par un adversaire. De telles interfaces sont :

- Interfaces de console de ligne de commande
- Consoles avec des messages de débogage, de diagnostic ou d'erreur détaillés
- Ports de débogage matériel tels que JTAG ou SWD
- Services réseau utilisés pour le débogage, le diagnostic ou les tests
- Interfaces administratives, telles que SSH ou Telnet

Toutes ces technologies doivent être désactivées dans la configuration approuvée.

Les ports série qui peuvent être désactivés par le système doivent également être retirés physiquement de la carte de circuit imprimé. Cependant, plusieurs fois, les ports série tels que UART / USART sont activés sur les broches du microcontrôleur ou du processeur. Si ces broches sont toujours activées en tant que console, un adversaire peut simplement se connecter aux broches pour interagir avec la console. La suppression du port série physique lui-même, comme une interface DB9, ne désactive pas la console.

En outre, les ports de débogage tels que JTAG et SWD ne doivent pas simplement être désactivés par des commandes du logiciel. Ces dispositifs doivent être désactivés en modifiant les fusibles de sécurité ou des verrous au niveau du matériel. Désactiver ces technologies à partir d'un logiciel offre une fenêtre d'opportunité pour un adversaire de se connecter à JTAG, SWD ou à une interface de débogage de matériel similaire avant l'heure à laquelle le logiciel désactive l'interface. Cette fenêtre d'opportunité est souvent suffisante pour qu'un adversaire réussisse.

8.2.1 Risque

Sans mettre en œuvre cette recommandation, les organisations s'ouvrent à l'extraction de secrets critiques du processeur. Cela peut permettre aux adversaires de charger leur propre microprogramme dans la NVRAM ou l'EEPROM, et leur permettre d'extraire ou de modifier les secrets critiques qui compromettent davantage le réseau ou un dispositif périphérique IoT.

La désactivation des ports de débogage est une étape critique pour garantir l'intégrité du produit ou service IoT. Cependant, il est important que l'entreprise évalue le risque de désactiver ces technologies et les compare contre l'avantage de pouvoir diagnostiquer et déboguer les problèmes identifiés sur le terrain. Il peut être beaucoup plus difficile de corriger les défauts au niveau de la production dans le produit s'il n'y a aucun moyen de déboguer un système en cours d'exécution.

8.3 Mémoire corrompue avec des attaques sur les interfaces

Les systèmes de traitement reposent sur la cohérence pour garantir que les résultats des algorithmes est prévisible par rapport à un ensemble d'entrées données. Les systèmes de traitement s'attendent également à ce que les composants agissent de manière fiable, et

que pour chaque bit écrit, ce bit soit stable et inchangé jusqu'à ce qu'il soit modifié par le processeur. Dans les systèmes fermés, cette théorie est applicable. Lorsque des anomalies se produisent dans ce modèle, elles peuvent compromettre ou simplement endommager un environnement de traitement.

La sécurité de l'information présente la classe d'anomalies volontairement induite afin d'accéder à des objets qui autrement seraient inaccessibles. Une fenêtre d'opportunité pour l'induction d'un comportement anormal qui soit utiliser par un adversaire est l'accès direct à la mémoire (DMA). Dit simplement, DMA est une technologie que les processeurs peuvent utiliser pour permettre aux composants externes (périphériques) d'accéder à la mémoire du processeur principal sans interférence de la part du CPU. En d'autres termes, la CPU peut accorder á un dispositif périphérique un accès direct à une région de mémoire. Ce périphérique peut alors lire ou écrire dans cette région de mémoire.

Si le processeur ne limite pas correctement la région de mémoire utilisable par le périphérique, ce dernier peut avoir accès à plus de mémoire principale que nécessaire pour la fonctionnalité prévue. En d'autres termes, si le périphérique (par exemple, un contrôleur Ethernet) est alloué une région DMA destinée à être utilisée comme tampon circulaire pour les trames Ethernet reçues, et la région DMA attribuée comprend toute l'étendue de la mémoire principale, le firmware sur le contrôleur Ethernet peut maintenant lire et écrire arbitrairement dans toute la mémoire du système. La CPU n'aura aucun moyen de bloquer le firmware du contrôleur Ethernet d'écrire en mémoire.

Le résultat de cette attaque est double. Les données peuvent être divulguées à partir de la mémoire principale et codées dans des paquets réseau ou des informations d'application pour être transmis d'une manière secrète ou immédiate. Alternativement, un attaquant pourrait secrètement insérer un logiciel malveillant (« malware ») dans la mémoire principale en remplaçant le code exécutable d'une application.

Du point de vue du processeur, il n'y a pas grand-chose à faire pour déterminer si une partie de la mémoire trop permissive a été utilisée abusivement par un périphérique malveillant. Pour lutter contre cette attaque, identifiez si le processeur utilisé dans le système des dispositifs périphériques est capable de restreindre le DMA à des petites régions prévisibles de la mémoire. Si c'est le cas, assurez-vous que chaque région de mémoire est définie pour chaque périphérique qui en a besoin. N'activez pas des espaces de mémoire arbitraires, si possible, pour les dispositifs périphériques.

Certains processeurs peuvent ne pas autoriser la restriction granulaire sur la taille ou l'emplacement dans la mémoire linéaire ou virtuelle d'un espace pour le DMA. Comme les attaques DMA doivent être considérées comme une menace réaliste pour les dispositifs périphériques IoT et leurs applications critiques, évaluez s'il est logique de considérer un processeur alternatif avec des fonctionnalités plus détaillés sur l'utilisation de la mémoire.

Pour les plates-formes qui exposent des ports tels que IEEE1394, Thunderbolt, Express Card ou d'autres ports qui permettent un accès direct à DMA PCI (« Peripheral Component Interconnect »), des attaques automatiques et rentables sont déjà disponibles.

Pour les plates-formes où une attaque DMA nécessite la violation d'un composant hardware local, la difficulté augmentera certainement, mais cela n'échappe pas à un compromis de

sécurité basé sur la possibilité de réenregistrer le firmware de ce périphérique pour modifier le comportement du DMA afin de compromettre un périphérique local. Le coût, le temps et l'expertise seront cependant un facteur, faisant de l'acteur dans ce cas probablement un adversaire sponsorisé (payé).

8.3.1 Risque

Le choix de ne pas restreindre l'utilisation abusive de DMA par des composants externes peut exposer la plate-forme à un compromis complet ou, au moins, à l'extraction de clefs secrètes, de données centrées sur la confidentialité ou de propriété intellectuelle à partir du dispositif périphérique.

8.4 Sécurité de l'interface utilisateur

Les dispositifs périphériques IoT dotés d'interfaces utilisateur telles que des écrans tactiles, des interfaces riches en données ou des technologies alternatives doivent être capables de fournir des informations à l'utilisateur et de collecter des informations d'un utilisateur de manière sécurisée.

Bien que les attributs de l'interface utilisateur, tels que les mots de passe, aient déjà été abordés dans ce document, il y a quelques problèmes plus subtils qui doivent être discutés :

- Systèmes d'alerte
- Confirmation des actions

Lorsqu'une anomalie s'est produite, telle qu'une falsification d'un composant ou une application se comportant de manière rare, l'utilisateur devrait recevoir une alerte visible. Alternativement, l'utilisateur devrait être en mesure de revoir les alertes du système à partir de l'interface utilisateur.

De plus, toutes les actions exécutées par l'appareil et pilotées par des encodages ou des transitions continues d'une interface à une autre doivent être confirmées par l'utilisateur. Un exemple de ceci est si la caméra du dispositif IoT lit un code QR, ou se connecte á une carte NFC ou RFID par exemple, il demandera se connecter à une URL sur le web. Dans ces cas, l'utilisateur doit être invité à confirmer l'action et à valider que l'action effectuée est souhaitable. L'utilisateur devrait avoir la possibilité d'annuler l'action. L'utilisateur devrait être en mesure de voir tous les détails sur l'action donnée, y compris l'URL complète.

8.4.1 Risque

Si cette recommandation n'est pas implémentée, les utilisateurs seront vulnérables aux attaques qui ne peuvent pas être détectées. Alors que certains concepteurs de système apprécient la transition sans délai de la lecture d'une puce RFID vers, disons, le site Web du produit correspondant, il peut y avoir des effets indésirables de ce comportement. Les utilisateurs pourraient être contraints de consulter des documents indésirables sans leur consentement, ou les utilisateurs pourraient être amenés à visiter des sites Web ou à effectuer des actions qui affaiblissent leur sécurité et qui peuvent mettre en danger ses données privées.

En outre, les utilisateurs qui ont du mal à consulter leurs alertes peuvent ne pas comprendre les risques liés à l'utilisation d'un périphérique potentiellement compromis. Cela peut diminuer la sécurité physique de l'utilisateur et pourrait le mettre en danger.

8.5 Audit de code tiers

Chaque fois qu'une section de code, telle qu'un bootloader, est un composant essentiel dans la construction d'une plate-forme d'exécution sécurisée, elle doit être auditée pour détecter les risques. Si un bootloader peut être manipulé par un adversaire pour exécuter un code non fiable, ou pour contourner la séquence d'authentification, il est totalement inefficace. Cela aura un impact sur les couts du système, le temps et l'expérience utilisés par l'entreprise dans le déploiement de cette technologie, rendant inutiles les dépenses d'ingénierie.

Une lacune dans la sécurité dans ce domaine peut également entraîner un avantage concurrentiel par rapport à l'entreprise par l'usurpation d'identité, les abus d'API, l'interception de données, le clonage de périphériques et même le changement de marque d'un appareil. Ainsi, il est impératif que les blocs critiques du code soient audités par une tierce partie approuvée, afin de s'assurer que la technologie ne risque pas d'être abusée. Par conséquent, pour trouver une équipe de sécurité de l'information adéquate pour effectuer l'audit, évaluez quels types de code seront audités. Typiquement, dans ce modèle, cela signifie : C, Assembly, et éventuellement C ++ ou Java.

Identifier une équipe qui connaît bien ces langages, ainsi que l'architecture sous-jacente. Alors que de nombreuses équipes de sécurité informatique effectuent des audits de code source, peu d'entre elles peuvent effectuer des audits sur la plate-forme particulière utilisée par l'entreprise IoT. Chaque plate-forme a des différences subtiles, et il est préférable d'embaucher une équipe familière avec la plate-forme utilisée.

8.5.1 Risque

Bien que l'embauche de consultants externes pour évaluer la technologie développée en interne peut être un défi, c'est une exigence de sécurité. C'est parce que les ingénieurs développant la technologie doivent être en mesure de montrer que leur architecture est vérifiable. Cela est difficile à se faire d'une manière exhaustive si les ingénieurs développant l'architecture sont les seuls à l'examiner. Les ingénieurs ont tendance à visualiser leur code à partir de l'architecture qu'ils ont tenté de concevoir et de mettre en œuvre, et non de l'implémentation réelle et voulue. Ainsi, des consultants externes sont souvent nécessaires pour trouver des subtilités dans l'architecture et la mise en œuvre qui pourraient causer des failles dans la sécurité.

8.6 Utiliser un APN privé

Dans les réseaux cellulaires 3GPP, un nom de point d'accès (APN) agit comme un réseau privé configuré spécifiquement pour un ensemble de dispositifs authentifiés. Généralement, un APN privé (également appelé «APN sécurisé») est un réseau privé accessible uniquement aux dispositifs authentifiés associés à une entreprise spécifique. En utilisant un APN, les entreprises peuvent restreindre ce que les terminaux peuvent connecter à leur infrastructure de service via le réseau cellulaire. Cela permet de réduire le nombre d'utilisateurs ayant un accès direct aux services IoT dans l'infrastructure principale.

D'autres attributs d'un APN privé peuvent aider à réduire le risque que les dispositifs périphériques non autorisés abusent de l'écosystème IoT. Les pares-feux peuvent limiter les services ou les ordinateurs auxquels l'APN peut être connecté. Un APN bien configuré empêchera les dispositifs de se connecter directement les uns aux autres, ce qui empêche un dispositif compromis de migrer horizontalement à travers l'infrastructure réseau vers d'autres dispositifs périphériques.

Il est important d'analyser avec l'opérateur cellulaire ou l'opérateur de réseau mobile virtuel (MVNO) avec lequel l'organisation travaille, les technologies disponibles dans l'APN sécurisé. D'autres services tels que la surveillance, la mise en liste noire des dispositifs compromis et l'association des identités des utilisateurs à certaines actions enregistrées peuvent être disponibles.

8.6.1 Risque

L'utilisation d'un APN privé peut alléger de nombreux types d'attaques. Par exemple, les APN privés permettent à l'entreprise de réduire directement le nombre de connexions pouvant être effectuées à partir du dispositif périphérique vers Internet. Les dispositifs ne doivent jamais être autorisés à se connecter directement à des ressources Internet non fiables. Seules les organisations partenaires doivent être approuvées et ces services doivent être authentifiés.

Sans l'utilisation d'un APN privé, les dispositifs compromis peuvent communiquer avec n'importe quel service ou protocole Internet sans restriction. Cela peut permettre à un adversaire d'abuser du dispositif afin de lancer une attaque secondaire sur une infrastructure séparée. Cela pourrait impliquer une attaque par déni de service (DoS), ou pourrait faciliter une attaque plus dangereuse contre une autre entreprise, un gouvernement ou une personne quelconque.

Il est à noter, cependant, qu'un APN privé n'atténue pas le risque qu'un adversaire compromette le lien de communication entre le dispositif périphérique et l'APN privé. En outre, l'APN privé agit uniquement comme une passerelle vers les services dorsaux et n'applique aucune sécurité entre l'APN et les services dorsaux sur le réseau privé du fournisseur de services IoT. Ces failles potentielles dans la sécurité doivent être traitées séparément, indépendamment des améliorations qui sont accordées grâce à l'utilisation d'un APN privé.

8.7 Mettre en œuvre des seuils de verrouillage environnementaux

Les composants d'un système intégré sont conçus pour être utilisés dans certains seuils environnementaux. Cela inclut les niveaux de tension, la consommation de courant, la température ambiante ou de fonctionnement et l'humidité. Chaque composant est généralement évalué pour certaines gammes de niveaux approuvés. Si le dispositif est soumis à des états au-dessus ou au-dessous d'un niveau donné, le composant peut agir de manière erratique ou se comporter d'une manière qui est utile à un adversaire.

Par conséquent, il est important de détecter les modifications apportées à ces niveaux environnementaux pour déterminer si l'appareil doit continuer à fonctionner ou s'il doit être mis hors tension. Il convient de noter, cependant, que la mise hors tension peut être un effet désiré, et que l'adversaire peut abuser de cette décision d'ingénierie pour tirer parti d'un déni

de service. L'équipe d'ingénierie devrait évaluer ce modèle pour déterminer s'il est plus avantageux d'éteindre le dispositif ou plus utile de tenter rester en ligne.

Peu importe, le modèle d'action comporte généralement :

- Détection de la perte et d'une panne d'alimentation, lorsque la tension chute trop bas
- Protection du circuit par rapport au limite supérieur de tension pour s'assurer que les niveaux de tension ne dépassent pas une valeur déterminée
- Circuits limiteurs de courant pour assurer que le courant ne peut pas tomber ou dépasser certains niveaux
- Surveillance de la température interne pour les CPUs, les MCUs et autres composants qui surveillent les niveaux internes
- Optionnellement, les niveaux d'humidité peuvent être évalués pour déterminer si l'environnement devient trop humide ou trop aride

La température est extrêmement importante car des températures élevées peuvent indiquer un problème de circuit déclenché par l'utilisateur, l'environnement ou même un problème du matériel ou du logiciel. La surveillance de la température permettra au système d'exploitation ou à l'application d'arrêter les ressources (ou l'ensemble du périphérique) pour s'assurer qu'un incendie ou un autre problème n'est pas causé par le dispositif périphérique.

Les niveaux bas de température modifient également le comportement d'un appareil. Cela peut ralentir un circuit ou provoquer une réaction inattendue de ses composants. Ceci peut être utile à un attaquant si la température peut provoquer une anomalie prévisible qui affecte l'application ou les circuits d'une manière qui lui soit utile.

La difficulté des seuils de verrouillage se manifeste lors de l'analyse de la température et de l'humidité. Les niveaux de tension et de courant doivent être atténués par des circuits de protection sur la carte de circuit imprimé ou dans le processeur. Puisque les ingénieurs seront en mesure de rechercher les niveaux les plus adéquats aux seuils de tension et de courant d'une puce, ils pourront facilement mettre en place des protections pour ces problèmes.

Pour la température et l'humidité, la décision d'agir est un peu plus difficile car ces niveaux peuvent être causer par un adversaire sans toucher à l'appareil. En cas de température, les niveaux pouvant être indicatifs d'un événement de sécurité imminent et doivent amener l'appareil à prendre les mesures adéquates pour abaisser la température. Cependant, dans des environnements critiques tels que les systèmes de contrôle industriels ou les dispositifs médicaux, l'appareil doit tenter de continuer à effectuer les opérations critiques, lorsque cela est possible. Si les niveaux dépassent un certain point défini sur lequel les ingénieurs et les chefs d'entreprise se mettent d'accord, alors seulement l'appareil doit s'éteindre.

8.7.1 Risque

Pour ce qui concerne à la consommation électrique (tension et de courant) d'un dispositif, le risque d'abus est lié à une attaque par « Glitch » et autres attaques par canaux auxiliaires qui peuvent bénéficier de changements dans ces niveaux. Si la détection d'événements sur l'alimentation est implémentée avec le processeur, le risque d'abus est réduit. Sinon, le risque est lié à des pics de tension ou de courant qui pourraient causer des problèmes de

sécurité physique du dispositif, ou permettre à un attaquant d'effectuer une attaque par « Glitch » (et des attaques similaires) pour annuler la sécurité des composants.

Ces problèmes devraient être corrigés en utilisant des circuits sur le circuit imprimé qui protègent les composants contre les pics anormaux ou les chutes de tension ou de courant.

Pour les changements trop importants de valeurs environnementales, le risque est lié à la sécurité de l'utilisateur. Les températures élevées causées par une utilisation excessive du processeur ou d'autres anomalies peuvent provoquer des brûlures, des échappements de substances chimiques ou même des incendies.

8.8 Appliquer les seuils d'avertissement de consommation d' énergie

Les dispositifs périphériques qui fournissent des services critiques à l'utilisateur doivent être activés avec un seuil d'avertissement qui indique les événements liés à l'alimentation du dispositif. Ces événements peuvent inclure :

- Un état de batterie faible
- Un état de batterie critique
- Des événements de manque d'alimentation
- Des événements qui mènent à une réduction ou instabilité importante de la tension
- Passer aux événements de sauvegarde de la batterie

L'utilisateur doit être prévenu dans un délai suffisant pour lui permettre de compenser la perte de puissance. Ceci peut être accompli en activant une LED qui indique un état d'alimentation particulier tel que vert pour *OK*, orange pour *Alerte* et rouge pour *Critique*.

Les systèmes connectés à l'alimentation de courant alternatif doivent être configurés pour avertir l'utilisateur lorsque des événements se sont produits dans l'absence ou altérations importante dans l'alimentation. En outre, le dispositif périphérique doit consigner ces événements dans la mémoire persistante pour garantir que l'utilisateur et l'administration peuvent récupérer les informations ultérieurement. L'information devrait être horodatée.

Le défi dans ce processus est d'identifier à quel rythme l'énergie de la batterie est épuisée et l'énergie supplémentaire nécessaire pour informer l'utilisateur d'un changement de l'état de l'alimentation. Tout cela peut être réalisé grâce à l'expertise dans l'ingénierie électrique, et ne devrait pas être trop difficile de concevoir pour les sociétés d'ingénierie expérimentées.

8.8.1 Risque

Sans un système d'alerte d'alimentation bien défini, les utilisateurs seront incapables de se préparer adéquatement aux événements d'alimentation potentiellement critiques. Bien que cela ne soit pas trop important dans le cas de dispositifs simples tels que les bracelets fitness, les montres digitales et autres dispositifs portables, les dispositifs plus critiques tels que les suiveurs personnels, les systèmes télématiques et les systèmes de sécurité domestique peuvent être sérieusement affectés.

8.9 Environnements sans connectivité dorsale

8.9.1 Méthode

Les dispositifs périphériques complexes ou légers, passerelles, doivent être capables de renforcer la sécurité des communications même dans les environnements où la connectivité au réseau principal n'est pas disponible. Indépendamment du fait que ce manque de connectivité soit temporaire ou non, le dispositif périphérique doit être capable d'appliquer la sécurité comme si le système principal était disponible.

Pour ce faire, la TCB doit être utilisé pour authentifier tous les homologues auxquels le dispositif doit communiquer des données centrées sur la confidentialité, de configuration ou des séquences de commandes. La TCB peut être utilisé pour s'assurer que les messages envoyés et reçus proviennent d'une entité qui a été provisionnée par la même organisation. Cela réduit la probabilité de communiquer avec un dispositif compromis.

L'interopérabilité peut toujours être obtenue en communiquant avec d'autres périphériques qui ne peuvent pas être authentifiés. Cependant, le type d'information qui est communiqué à ces dispositifs devrait être restreint aux classes de données inter-opérationnelles et non-sensibles.

Le défi consiste à décider quels dispositifs périphériques doivent s'authentifier et avec quels dispositifs se communiquer en clair. L'entreprise doit décider quels types de données sont classées et doivent être conservés par des pairs non authentifiés. Une fois cette classification des données obtenue, l'organisation sera en mesure de déterminer quels pairs sont raisonnablement fiables même sans l'aide des services IoT de base.

8.9.2 Risque

Le risque de déployer des solutions dans des environnements sans communication est que cela ouvre la possibilité à la concurrence d'abuser de l'infrastructure. Les concurrents peuvent réduire l'activité en offrant l'interopérabilité et en utilisant des sites sans connexion pour faire ses essais.

Au lieu de cela, l'entreprise peut choisir d'autoriser l'interopérabilité, mais jusqu'à un certain point. Certains droits de propriété intellectuelle et services de base peuvent alors être réservés uniquement aux homologues authentifiés qui sont validés par l'utilisation d'une TCB. Cela aide à réduire l'exposition de l'entreprise aux problèmes de propriété intellectuelle et aux concurrents adversaires.

8.10 Mise hors service et caducité des dispositifs périphériques

Tous les dispositifs périphériques ont un cycle de vie, comme indiqué ailleurs dans ce document. Certains dispositifs doivent être mis hors service (déclassement) en raison de l'annulation de leur abonnement par l'utilisateur, alors que d'autres en raison d'un comportement anormal ou contradictoire. Quelle que soit la raison, l'entreprise doit être prête à déclasser le dispositif en toute sécurité en utilisant sa TCB et son modèle de communication.

La caducité progressive, comme discuté ailleurs dans ce document, est le processus de mise hors service d'un ensemble de dispositifs et de ses services. Une entreprise qui a

« abandonné » un produit ou un service ou qui décide de fermer doit déclasser ses appareils et réseau afin de réduire le risque d'abus par des adversaires qui tâchent de prendre le control sur tous ces éléments.

Pour ce faire, la TCB et les protocoles de soutien doivent être utilisés. Généralement, le processus consiste à :

- Créer un message de déclassement à partir de l'écosystème de service
- Adapter le message pour le dispositif périphérique unique qui reçoit le message
- Signez le message à l'aide de la clef PSK de déclassement ou de la clef asymétrique
- Envoyer le message jusqu'au dispositif périphérique
- Recevoir un message du dispositif reconnaissant cryptographiquement le déclassement
- Invalider le dispositif périphérique en concret dans la liste des périphériques authentifiés
- Interdire toute autre communication de ce dispositif

Côté du dispositif, l'application exécutée sur son logiciel doit :

- Se connecter à des services back-end critiques de l'écosystème de Services
- Interroger le service pour voir s'il a des messages critiques
- Recevoir le message de déclassement
- Vérifier la signature du message à l'aide de la TCB et de l'ancre de confiance
- Générer le message de confirmation et le signer cryptographiquement à l'aide de la clef PSK personnalisée ou de la clef asymétrique
- Effectuer l'opération de déclassement
- Renvoyer le message au service critique

Il est important que le message soit signé et préparé pour la transmission avant la désactivation liée au déclassement, car le processus de déclassement inclut l'invalidation et la suppression des clefs de sécurité de l'ancre de confiance. En raison de ce processus, les clefs utilisées pour signer le message de déclassement ne seront pas disponibles. Le service requiert la réception d'un message dont l'intégrité est vérifiable pour s'assurer que le dispositif périphérique a effectivement reçu et traité le message.

La difficulté de ce processus est principalement que la mise hors service d'un dispositif périphérique potentiellement compromis suppose que le périphérique n'a pas été compromis au point où il rejettera la commande de déclassement. S'il a été suffisamment compromis, il peut ne pas exécuter la commande de déclassement.

Par conséquent, il est impératif que le back-end fonctionnant dans l'écosystème de services invalide le dispositif périphérique pour qu'il ne puisse pas se communiquer avec des services critiques. Si le dispositif tente d'interagir avec d'autres éléments de l'écosystème, le système back-end doit déclencher une alerte et informer l'administrateur qu'un événement anormal s'est produit.

8.10.1 Risque

Les risques de ne pas mettre en œuvre le déclassement et la caducité sont nombreux, de la prise en charge complète d'un réseau entier par des adversaires jusqu'à permettre aux dispositifs compromis de continuer à utiliser des services sur le réseau. Le risque le plus courant est associé aux utilisateurs ayant mis fin à leur abonnement auprès d'un fournisseur de services IoT. Si ces utilisateurs ne sont pas déconnectés du réseau, ils peuvent continuer à communiquer avec d'autres dispositifs du réseau IoT ou accéder à des services qui ne devraient plus être accessibles. Cela entraîne un coût pour le fournisseur de services IoT, qui doit payer pour la bande passante, le temps CPU et le stockage dans l'écosystème de service.

8.11 Collecte de métadonnées non autorisées

L'IoT moderne est conçu pour relier le monde physique au monde numérique. Dans ce modèle moderne, les effets de la technologie sont potentiellement beaucoup plus invasifs qu'auparavant. En utilisant des métadonnées, des entreprises ou des particuliers peuvent intentionnellement suivre et surveiller le comportement de consommateurs aléatoires ou spécifiques.

L'analyse des métadonnées est utilisée lorsque la communication entre deux entités du réseau est cryptée, mais les structures de protocole qui identifient le type de message ou l'identité de l'expéditeur et / ou du récepteur sont exposées. Cette métadonnée peut être utilisée pour dériver l'intention.

Considérez le scénario où les automobiles émettent des messages contenant des métadonnées qui sont attribuables à un consommateur spécifique. Toute personne ayant la capacité de suivre (localement ou à distance) ce type de métadonnées peut être capable de surveiller les mouvements du consommateur, puis de dériver un comportement ou une intention de ces mouvements. S'il y a des failles de sécurité qui sont exploitables dans le système télématique du véhicule, il peut être possible de suivre et de cibler le système télématique d'un consommateur spécifique, en les exposant à des risques physiques.

Les organisations juridiques et les compagnies d'assurance s'inquiètent de l'impact de ces risques sur l'avenir des services automobile et commencent à s'impliquer dans la législation et les normes qui détermineront la manière dont les ingénieurs doivent concevoir l'équipement télématique. Ce changement finira par se répercuter sur les segments verticaux IoT moins actifs, à mesure que d'autres technologies seront développées.

Pour lutter contre la collecte de métadonnées, cryptez autant de données que possible et utilisez des identifiants binaires uniques pour les modules de communication. Appliquer une stratégie interdisant aux utilisateurs externes d'utiliser l'API du système IoT pour dériver des numéros de série matériels et d'autres identités pouvant être suivies à partir de profils utilisateur. Dans la mesure du possible, ne permettez pas à la structure d'un message d'être exposée à des tiers. Ne permettez pas que les actions, activités ou comportements soient exposés à des tiers. Faire respecter la confidentialité et l'intégrité de toutes les données relatives à la vie privée des utilisateurs.

8.11.1 Risque

L'implémentation d'une sécurité des communications médiocre peut permettre la collecte de données ou de métadonnées qui met en péril les données privées de l'utilisateur final. À mesure que les agences d'assurance plaident en faveur de la mise en application des exigences de confidentialité de l'utilisateur final dans la technologie, l'entreprise peut se mettre en danger si elle ne prend pas la responsabilité des données générées par ses appareils.

9 Recommandations de priorité basse

Les recommandations de basse priorité englobent l'ensemble des recommandations qui s'appliquent aux risques extrêmement coûteux à combattre ou qui ne sont pas susceptibles d'affecter la conception du dispositif périphérique. Bien que ces recommandations soient utiles et que les informations détaillées dans les recommandations soient importantes, les stratégies d'atténuation ou de remédiation discutées peuvent être hors de portée en ce qui concerne l'entreprise. Évaluer chaque recommandation et déterminer si les risques décrits sont pertinents ou importants pour l'entreprise et ses clients. Si les clients ont besoin de prendre en charge ces risques, appliquez les recommandations.

9.1 Déni de service intentionnel et non intentionnel

Pour les communications radio, il existe une menace constante de brouillage, ou la diffusion intentionnelle de bruit ou de modèles qui peuvent être utilisés pour brouiller les signaux légitimes. Comme les signaux radio sont simplement composés d'électrons volant dans l'espace selon un modèle spécifique, il est assez facile d'élaborer une série de signaux qui interrompent ou altèrent le modèle qui forme les données de communication.

Typiquement, le but d'une telle attaque est une simple perturbation, pour interdire ou refuser le service aux utilisateurs légitimes. Dans d'autres cas, l'abus peut être plus utile. Par exemple, les protocoles de communication qui n'ont pas de mécanisme d'authentification peuvent être usurpés. Pour ce faire, le signal doit être bloqué afin que le signal usurpé de l'adversaire atteigne plus facilement la cible.

Un exemple de ceci est l'usurpation par GPS (« Global Positioning Systems »). Les signaux GPS civils manquent de cryptage et d'authentification, car il s'agit essentiellement d'un signal de diffusion en clair que tout le monde peut recevoir. Il s'agit également d'un signal radio relativement faible et il est facilement atténué par des anomalies environnementales telles que les préamplificateurs UHF (Ultra High Frequency) pour récepteurs de télévision et micro-ondes.

Pour les périphériques qui requièrent des informations de localisation pour fonctionner correctement, un signal GPS coincé peut entraîner un risque de fiabilité pouvant entraîner un risque de sécurité de l'information, en particulier si une usurpation d'identité est utilisée par la suite.

Pour lutter contre le brouillage et d'autres formes d'attaques par déni de service (DoS), développez un protocole de communication robuste qui se concentre sur les méthodes de dévaluation des interruptions de service. Le réseau doit détecter si des périphériques ont soudainement ou anormalement quitté le réseau. Chaque dispositif périphérique devrait

lancer un message lorsqu'il a l'intention de quitter le réseau. Si ce n'est pas le cas, l'anomalie doit être notée pour l'analyse statistique.

En outre, les clefs de sécurité de communication doivent être renégociées chaque fois qu'un périphérique rejoint le réseau. La même clef de sécurité des communications ne doit pas être utilisée. Il devrait être amorcé par la même clef cryptographique asymétrique, mais toute clef symétrique dérivée de la négociation de clef devrait être renouvelée pour chaque session de communication.

Un brouillage involontaire peut se produire sur un signal radio pour plusieurs raisons : des conditions environnementales qui interdisent la propagation du signal, un équipement défectueux ou même un équipement adjacent fonctionnant à la même fréquence. Indépendamment de la raison sous-jacente, les ingénieurs qui utilisent des communications radio s'attendent à ce qu'il y ait des conditions temporelles qui provoquent une dégradation ou une perte de signal. Ces pertes doivent être compensées par la conception de l'application et du protocole de communication réseau.

Il est recommandé aux développeurs de lire les consignes d'efficacité de la connexion de la GSMA [9], qui contiennent des conseils sur la protection contre les attaques par déni de service involontaires et des conseils sur le DHIR (« Device Host Identity Reporting »).

9.1.1 Risque

Si vous ne parvenez pas à combattre le risque de DoS intentionnel, vous obtiendrez un comportement des dispositifs périphériques anormal ou non sécurisé. Si le dispositif utilise toujours la même clef de session, les adversaires pourraient abuser de l'architecture du réseau pour recueillir des informations sur la clef symétrique utilisée pour sécuriser les communications. Construire correctement une session sécurisée après chaque session déconnectée est impératif pour la sécurité des communications entre dispositifs.

9.2 Analyse critique de sécurité

La plupart des produits IoT intégreront certains aspects du monde physique avec la technologie numérique. En conséquence, il est probable qu'un humain prendra une décision dans le monde physique sur la base des informations fournies par un dispositif périphérique IoT. Alternativement, un dispositif IoT peut prendre une décision qui affecte le monde physique avec des informations obtenues à travers le monde numérique.

Par conséquent, il est impératif que les fournisseurs de services IoT évaluent leur produit du point de vue de la sécurité pour déterminer si, comment et quand la vie humaine peut être affectée par la technologie. Si des mesures de protection adéquates ne sont pas mises en place pour s'assurer que la technologie ne peut pas être utilisée abusivement afin de causer des dommages physiques, leurs clients peuvent être mis en danger.

Pour résoudre le problème de sécurité, discutez avec les équipes exécutives, juridiques et d'assurance du fournisseur de services IoT. Assurez-vous que ces équipes comprennent les capacités et les limites de la technologie utilisée dans le produit ou le service. Déterminer si ces technologies peuvent répondre aux besoins de l'entreprise et offrir aux clients le niveau de sécurité nécessaire pour l'application prévue.

9.2.1 Risque

Manifestement, le fait de ne pas prendre le temps d'évaluer l'impact du produit ou du service sur la sécurité des clients pourrait entraîner des pertes de revenus, des accidents imprévus ou même des accidents plus sérieux que peuvent mettre en danger la vie de l'utilisateur.

9.3 Vaincre les composants répliqués et les ponts non fiables

Les composants sur le circuit physique n'utilisent typiquement aucune stratégie fiable de confidentialité et d'intégrité lorsqu'ils communiquent entre eux ou avec l'unité centrale. En conséquence, tout adversaire peut lire ou écrire des données transmises sur ces bus. L'effet de cet écart dans la sécurité des communications est la capacité pour un adversaire d'usurper l'identité des dispositifs légitimes sur le circuit physique. Les adversaires peuvent emprunter l'identité d'un composant critique tel que NVRAM, RAM, ou même une ancre de confiance, s'ils le veulent.

Le but de cette attaque serait de contourner la sécurité employée entre deux composants sur le bus. Un exemple typique de ce scénario consiste à utiliser cette faiblesse pour contourner le processus de validation d'intégrité de l'analyse d'une image d'application stockée dans la mémoire NVRAM. Lorsque la CPU récupère la mémoire stockée dans la mémoire NVRAM, l'attaquant peut utiliser un système d'intercommunication pour fournir le contenu de la mémoire réelle à la CPU. Lorsque l'application exécutée sur la CPU a vérifié l'intégrité de l'image de l'application, l'attaquant peut alors instrumenter les communications sur le bus physique pour échanger sélectivement les contenus NVRAM qui sont bénéfiques à l'attaquant. En d'autres termes, la CPU vérifie une image d'application (l'image originale), mais charge l'image de l'attaquant dans la RAM et l'exécute.

Une façon de se prémunir contre cette attaque est :

- Charger le contenu NVRAM dans la RAM
- Valider l'image de l'application chargée dans la RAM
- Exécuter le code directement dans la RAM ou mettre en cache le contenu dans la RAM

Il convient également de noter à ce stade qu'un attaquant pourrait également subvertir la RAM, affaiblissant ainsi ce processus. Cependant, effectuer une attaque « man-in-the-middle » contre une mémoire RAM est beaucoup plus complexe et coûteuse qu'une attaque contre une NVRAM car la vitesse du bus et les accès sont beaucoup plus rapides et plus erratiques qu'avec la NVRAM, qui est principalement accessible en blocs.

Alternativement, l'attaquant peut créer des sommes de contrôle pour de plus petites régions du contenu NVRAM validé et vérifier périodiquement les signatures de NVRAM. Si les totaux de contrôle diffèrent, le contenu est manipulé. Cela peut réussir, mais a un potentiel de succès inférieur car l'adversaire peut seulement manipuler une petite quantité de données qui n'est pas vérifiée aléatoirement par l'application en cours d'exécution.

Il convient de noter que, bien que la meilleure façon de se prémunir contre cette attaque est de valider le contenu de la NVRAM puis de la charger dans la RAM exécutable, il n'y a pas de solution complète pour ce problème. Le coût de la sécurisation des composants

physiques est si élevé qu'il n'est pas pratique de résoudre cette attaque de manière plus complète, sauf si le client a besoin de telles garanties de sécurité.

Cette attaque est encore plus simple lorsqu'un protocole de communication physique plus basique, tel que I2C, est utilisé. Les bus tels que I2C sont essentiellement des systèmes de diffusion physique. Ainsi, tout composant connecté sur le bus I2C peut prétendre être n'importe quel autre composant. Cela permettra à un adversaire d'emprunter l'identité d'autres appareils sur le bus qui n'appliquent pas la confidentialité et l'intégrité sur le canal de communication. En cas de problème, appliquer la confidentialité et l'intégrité dans le protocole d'application utilisé en plus du protocole de bus physique.

9.3.1 Risque

Le risque de ne pas implémenter une solution entraînera la possibilité pour un adversaire de contourner les contrôles d'intégrité dans l'application. Cela permettra à l'attaquant de compromettre l'application en cours d'exécution par un code plus privilégié, tel que les bootloaders ou les TCB.

Il convient de noter, cependant, que cette attaque est beaucoup moins probable que des attaques plus simples contre le bootloader. L'exécution d'une attaque matérielle « man-in-the-middle » contre des composants tels que NVRAM ou des composants haute vitesse tels que la RAM est complexe et coûteuse. Bien qu'il soit toujours possible pour un adversaire de subvertir un système embarqué de cette manière, il peut être trop coûteux de le faire.

Ainsi, charger le code dans la RAM et vérifier l'intégrité peut être une solution raisonnable qui contournera la majorité des attaques, le cas échéant.

Aussi, pour les raisons décrites ci-dessus et plus, les clefs cryptographiques ne devraient pas être conservées dans des éléments non sécurisés comme ceux-ci. Ils doivent être stockés dans une ancre de confiance et utilisés par la TCB, et non stockés dans des supports tels que NVRAM qui pourraient être usurpés ou compromis.

9.4 Vaincre une attaque de démarrage à froid

Une attaque de démarrage à froid est une stratégie d'attaque physique contre les systèmes informatiques qui extrait des secrets d'un ordinateur en cours en retirant la mémoire physique de l'ordinateur et en plaçant la mémoire dans un système secondaire contrôlé par l'adversaire. L'avantage de cette attaque est que l'attaquant peut exécuter un système d'exploitation personnalisé qui sauvegarde le contenu de la RAM dans un stockage permanent. Cela permettra à l'attaquant de passer au peigne fin les données récupérées et de déterminer si des tokens liés à la sécurité peuvent être utilisés. Cela peut inclure :

- Secrets cryptographiques ou clefs privées
- Informations d'identification de connexion (noms d'utilisateur et mots de passe)
- Informations personnelles identifiables (PII)
- Tokens d'accès pour les services Web

Le but de l'attaque est de compromettre les secrets qui permettent à l'attaquant d'accéder à long terme à une ressource qui serait autrement hors de leur portée. Par exemple, briser les algorithmes cryptographiques utilisés dans la norme la plus récente de TLS serait

impossible pour l'attaquant moyen. Cependant, compromettre le certificat client privé utilisé dans un service TLS à authentification mutuelle permettrait à l'attaquant de simuler le client à partir d'un système plus pratique.

Pour réussir cette attaque, l'attaquant doit être capable de retirer la RAM du système informatique cible sans que les bits stockés dans la puce ne changent. Comme détaillé dans le document de recherche [22], cela peut être accompli en refroidissant les puces de mémoire. Cependant, la RAM doit être facilement amovible. Si la RAM est soudée à la carte de circuit imprimée, cela compliquerait énormément l'attaque et exigerait que l'attaquant utilise un pistolet à souder pour extraire de la mémoire, ce qui pourrait endommager le contenu.

Il est important de noter que le nettoyage de la mémoire à l'arrêt est toujours utile et conseillé pour améliorer la confidentialité d'un dispositif périphérique. Pourtant, une attaque de démarrage à froid peut se produire à tout moment, même lorsque le système est en cours d'exécution. Par conséquent, nettoyer la mémoire peut être utile, mais peut ne pas réussir à vaincre une attaque réelle.

Une atténuation plus efficace de cette attaque consiste à exécuter les actions centrées sur la sécurité à l'aide de RAM interne au processeur. De nombreux processeurs, microcontrôleurs et MPU disposent d'une petite quantité de mémoire SRAM interne pouvant être utilisée par une application en cours d'exécution. Si l'application limite l'utilisation de tokens de sécurité critiques (tels que des clefs privées) à cette RAM interne, le contenu de la RAM amovible (ou externe) aura moins de valeur pour un attaquant.

9.4.1 Risque

Ne pas considérer le risque d'une attaque de démarrage à froid peut entraîner l'extraction de clefs de sécurité critiques à l'aide d'un simple modèle d'attaque. Si les clefs de sécurité sont universelles pour tous les dispositifs périphériques de l'écosystème du fournisseur de services IoT, un compromis important peut être possible.

Pour plus d'informations, voir - https://citp.princeton.edu/research/memory/

9.5 Risques de sécurité non évidents (« voir à travers les murs »)

Malgré l'activation et l'application de l'authentification mutuelle, de la confidentialité et de l'intégrité dans le réseau de communication, les modèles de trafic peuvent être directement corrélés aux événements. Lorsque les données sont traitées en réponse à certains événements physiques, une corrélation peut éventuellement être établie entre les événements physiques et les données. Cela peut permettre à un adversaire de surveiller les modèles de signal, puis d'en déduire le sens, que l'adversaire ait ou non un accès direct aux données en clair.

Un exemple de cela est la technologie domotique qui réagit en fonction de la présence physique d'un utilisateur dans une pièce particulière. Un adversaire capable de surveiller à distance le système de communication peut être capable d'observer combien d'utilisateurs sont dans une maison particulière, où les utilisateurs sont dans la maison, et qui est l'utilisateur, uniquement en observant les modèles de communication entre dispositifs périphériques IoT, passerelles et systèmes dorsaux.

L'adversaire peut être capable de différencier facilement une maison très peuplée et des maisons où il y a une seule personne et la localisation de cette personne. Les compagnies d'assurance et les personnes morales devront comprendre comment cela augmente potentiellement le risque pour les propriétaires et les autres locataires dans leurs maisons.

Combattre ce risque peut être difficile. Le modèle le plus commun et le plus simple consiste à envoyer des échantillons à un débit prédéfini, que l'utilisateur soit présent ou non pour prélever des échantillons. Si la confidentialité et l'intégrité sont appliquées, en empêchant les adversaires distants d'évaluer le texte en clair des données, un observateur sera incapable de faire la différence entre un échantillon contenant une activité de l'utilisateur et un échantillon vide.

Toutefois, ce modèle présente des problèmes, tels qu'une augmentation de la saturation du spectre, une augmentation de la consommation d'énergie pour les technologies à faible consommation ou à batterie, et le niveau de traitement accru nécessaire pour déchiffrer, vérifier et interpréter les échantillons vides.

Une alternative consiste à envoyer des échantillons à des intervalles aléatoires, avec des rafales variables. Ce type de modèle est moins coûteux, demande moins de puissance et nécessite moins de puissance de traitement. Pourtant, il peut toujours être possible d'observer des changements subtils qui indiquent la présence de l'utilisateur. Par exemple, tout système véritablement entropique est entièrement aléatoire et imprévisible. Le comportement de l'utilisateur est cependant entièrement prévisible. Si un utilisateur entre dans une pièce et que les capteurs de cette pièce réagissent et commencent à envoyer des données à des dispositifs IoT présents dans le réseau, l'introduction d'un comportement cohérent peut indiquer la présence d'un utilisateur.

Toute équipe développant une technologie sujette à ce type de risque devrait étudier les effets potentiels de l'exposition à la vie privée et consulter l'équipe juridique pour déterminer si la technologie aurait un effet sur la position juridique ou le modèle d'assurance de l'entreprise.

9.5.1 Risque

Si le fournisseur de services IoT n'évalue pas sa technologie du point de vue des expositions potentielles à la vie privée et des risques de sécurité, l'architecture peut devoir être considérablement remaniée afin de compenser les risques qui doivent être traités. Au lieu d'essayer de faire des ajustements coûteux à l'architecture à un moment ultérieur, introduisez ces solutions dans le produit au début de la phase d'ingénierie, ou, dès que possible.

9.6 Combattre les faisceaux d'ions focalisés et les rayons X

Un faisceau d'ions focalisé (FIB) est un instrument de fabrication couramment utilisé dans l'évaluation des semi-conducteurs. La technologie est capable d'inspecter et de modifier les circuits au niveau nanométrique, ce qui permet aux analystes d'identifier les défauts de fabrication et de tester les correctifs de circuit avant de modifier le processus de fabrication.

En matière de sécurité de l'information, un FIB peut être utilisé pour exploiter des bus internes dans le but d'intercepter des données traitées sur des composants internes. En

outre, un FIB peut être utilisé pour modifier les circuits internes, ce qui modifie le fonctionnement du composant interne, permettant à un adversaire de contourner une restriction de sécurité.

Presque tous les appareils sont sujets à l'attaque d'un FIB. Cependant, seuls certains périphériques seront analysés via un processus FIB. C'est parce qu'un FIB lui-même est une technologie extrêmement coûteuse, à environ 1.000.000 USD par unité. En raison du coût élevé de la technologie, peu d'organisations ont un tel dispositif dans leur écosystème d’outils de test. De plus, l'appareil n'est pas automatisé. Il nécessite un haut degré de compétence pour le manipuler, ainsi qu'un très haut niveau d'expertise dans l'analyse des semi-conducteurs, pour être utilisable. Ainsi, le coût réaliste d'un FIB est bien au-delà d'un simple chiffre d'un million de dollars, et se prolonge dans les millions de dollars pour l'utilisation elle-même de l’appareil, et la formation, le salaire et l'expertise de l'utilisateur.

Des organisations sont toutefois disponibles pour l'externalisation de ce type d’analyses. Comme l'ingénierie inverse est en grande partie légale, les organisations fourniront des services d'attaque de semi-conducteurs aux clients qui sont intéressés par l'ingénierie inverse d'un dispositif. Ces missions coûtent entre 10 000 USD et 1 000 000 USD selon le niveau de personnalisation et d'expertise requis pour attaquer un composant particulier. Par exemple, une entreprise experte en FIB aurait un playbook pour contourner les protections sur une puce commune. Mais, une solution FPGA personnalisée avec une technologie de verrouillage de sécurité innovatrice coûterait beaucoup plus cher car aucun playbook n’auraient pas encore les règles nécessaires. Un nouveau processus serait nécessaire pour utiliser le FIB avec succès, ce qui coûterait beaucoup de temps et d'argent.

Certaines nouvelles technologies, telles que les variantes d'ancre de confiance modernes, revendiquent la résistance des sondes FIB. Bien qu'il y ait une certaine validité à ces revendications, toute protection matérielle qui n'est pas dynamique (et la plupart ne le sont pas) donnera lieu à un playbook après avoir passé suffisamment de temps à analyser les techniques de contournement. Par conséquent, ces nouvelles revendications peuvent être valides, mais ne peuvent être valables que pour une certaine période de temps.

Par conséquent, pour compenser les technologies d'attaque invasives mais presque toujours efficaces comme celles-ci, il est impératif que l'entreprise d'ingénierie conçoive une stratégie de sécurité qui n'atteigne pas son succès sur l'ancre de confiance seule. Au lieu de cela, un protocole suffisant doit être conçu qui utilise la technologie comme une ancre de confiance de base, mais personnalise les clefs cryptographiques de chaque dispositif périphérique de sorte qu'aucune compromission d'un seul périphérique ne compromette l'ensemble du réseau des dispositifs.

Considérez le scénario dans lequel un adversaire doit utiliser un FIB pour extraire des données cryptographiques de chaque dispositif qu'il souhaite cibler. Cela deviendrait rapidement une proposition extrêmement coûteuse et serait hors de portée pour presque tous les budgets des adversaires. Étant donné que ces méthodes d'attaque ne peuvent pas être suffisamment atténuées, elles doivent être dévaluées afin de réduire les risques par une bonne conception d’architecture, et non par « l'obscurité ».

9.6.1 Risque

Le risque d'un FIB est que les secrets cryptographiques et d'autres propriétés intellectuelles peuvent être extraits presque de n'importe quel composant, même ceux qui sont sécurisés. Puisque vaincre un FIB d'une manière rentable pour l'IoT du consommateur est impraticable, l'entreprise doit modifier sa stratégie de protection des systèmes de dispositif périphériques ou risquer un compromis complet de l'écosystème.

9.7 Considérer la sécurité de la chaîne d'approvisionnement

La sécurité de tout système informatique commence avec les composants bruts qui s'intègrent dans une carte de circuit imprimée. Le silicium, les tokens cryptographiques, la mémoire morte (ROM), le micrologiciel et d'autres attributs de base d'un système embarqué contribuent tous à la sécurité d'un tel système. Si l'un de ces composants est altéré, l'ensemble du système pourrait être soumis à un compromis de sécurité.

Par conséquent, les fournisseurs de services IoT soucieux de la sécurité doivent tenir compte de la source de leurs composants, de leur assemblage et du processus d'exécution utilisé pour expédier la technologie assemblée. Si le processus utilisé pour générer la technologie n'est pas soigneusement planifié, un seul point de défaillance dans le processus pourrait entraîner une défaillance de sécurité critique.

Considérez les problèmes suivants :

- Où et par qui le silicium est-il fabriqué?
- La conception du silicium a-t-elle été analysée par une équipe de sécurité de l'information tierce expérimentée?
- Le silicium sera-t-il fabriqué dans une installation sécurisée?
- Comment la mémoire EEPROM ou NVRAM sera-t-elle remplie avec une image exécutable, telle qu'un bootloader?
- Le processus de la mise à jour de l'image exécutable est-il sécurisé?
- Comment l'image exécutable sera-t-elle livrée au fabricant?
- L'image exécutable est-elle vérifiée une fois qu'elle a été flashée sur l'EEPROM ou la NVRAM?
- Comment les secrets cryptographiques sont-ils provisionnés sur les puces?
- Si des secrets sont générés chez le fabricant, utilisent-ils un Générateur de Nombre Aleatoire éprouvé pour générer les clefs?
- Toutes les clefs de sécurité sont-elles uniques selon les recommandations de la TCB?
- Comment les secrets cryptographiques sont-ils partagés avec le fournisseur de services IoT? Sécurisé?
- Comment les identifiants de puces uniques (numéro de série, etc.) sont-ils corrélés avec les secrets cryptographiques et partagés avec le fournisseur de services IoT?

Bien que le choix d'une installation plus sûre pour construire et assembler un produit puisse entraîner un coût plus élevé, il peut être une étape majeure pour l'entreprise. Cela dépend du cas d'utilisation du produit, de l'environnement de déploiement prévu, du client prévu et d'autres facteurs tels que la sécurité humaine, les applications militaires et les déploiements d'infrastructure critiques. Lorsque la vie humaine peut être affectée par la technologie qui en

résulte, la chaîne d'approvisionnement doit être évaluée en fonction des lacunes en matière de sécurité.

9.7.1 Risque

Sans la sécurité de la chaîne d'approvisionnement, l'entreprise est exposée à de nombreux risques, dont certains peuvent être totalement inattendus et pourtant critiques pour l'entreprise :

- Clonage de dispositifs périphériques (fabrication illégale)
- Vol de technologie (les concurrents qui la volent en contactant et offrant une sous-cotation aux fournisseurs de services)
- Vol d'identité (interception de données ou attaques d'usurpation d'identité)
- Injection d'implants software malveillants («portes dérobées» malveillantes pouvant être activées ultérieurement)

9.8 Interception légale

L'interception légale est l'acte d'intercepter ou de manipuler légalement les communications entre un client et un fournisseur de services. Cela peut fonctionner de deux façons. Premièrement, le scénario le plus courant est qu'un organisme d'application de la loi soumettra une demande légale à un opérateur et demandera l'accès à des métadonnées ou à des données réelles provenant de communications faites par un abonné particulier. Deuxièmement, l'organisme d'application de la loi demandera au fournisseur de services IoT d'accéder aux données et / ou aux métadonnées d'un abonné particulier. Dans le scénario où l'agence demande l'accès via l'opérateur, le fournisseur de services IoT ne peut jamais être averti qu'il y a un problème, en fonction de la portée de la demande légale. Ainsi, le fournisseur de services doit être prêt à mettre en œuvre ou à se conformer à une demande légale faite par une telle agence.

Par conséquent, le fournisseur doit identifier les problèmes de confidentialité qui peuvent découler d'une demande d'application de la loi et être prêt à fournir des informations pertinentes au modèle juridique et à la politique de confidentialité de l'entreprise, dans leurs capacités juridiques respectives.

Dans un passé récent, des entreprises telles que Google, Apple et d'autres grandes entités ont adopté des Garanties du Canari pour informer légalement les utilisateurs lorsqu'une demande secrète a été faite à l'entreprise pour le compte d'une agence. L'entreprise peut supprimer une phrase, une image ou tout autre objet représentatif du fait qu'elle n'est pas en contact avec des agents d'interception légitimes. La suppression de cet objet est, bien sûr, indicative d'une demande qu'a était faite.

9.8.1 Risque

Ne pas préparer l'entreprise à une demande d'interception légale désavantage l'entreprise si une telle exigence est imposée à l'entreprise. Il se peut que l'entreprise doive se conformer à la demande, mais ne dispose peut-être pas de l'infrastructure juridique ou des politiques de confidentialité, ce qui pourrait la mettre en danger.

Ne pas préparer le protocole du dispositif périphérique et la plate-forme IoT pour assurer une confidentialité et une intégrité adéquates permettra aux communications d'être interceptées du côté du réseau sans que l'entreprise en ait connaissance. Cela peut mettre l'entreprise en péril en raison de la fuite des données de l'utilisateur ou d'un événement tel que les fuites de la NSA (« National security Agency ») du cas Snowden, réduisant considérablement la confiance du public dans la capacité de l'entreprise à sécuriser les données de l'utilisateur.

10 Résumé

En résumé, presque tous les risques de sécurité dans un produit ou service IoT peuvent être combattus par une architecture bien définie, en utilisant des solutions intelligentes pour identifier les risques avant et pendant les événements liés à la sécurité, et des politiques et procédures pour gérer de tels événements. En analysant quels concepts de sécurité de haut niveau sont importants pour le fournisseur de services IoT, les foires aux questions de sécurité peuvent être examinées. Cela devrait guider l'équipe d'ingénierie vers les recommandations qui sont les plus pertinentes pour résoudre les lacunes dans leur architecture de sécurité.

Au fur et à mesure que l'équipe progresse dans la définition de l’architecture, elle peut passer en revue des recommandations autonomes lorsque leurs questions et préoccupations de sécurité deviennent plus spécifiques à leur propre implémentation.

Dans l'ensemble, chaque équipe d'ingénierie devra faire face à des risques très similaires. Il est impératif que l'entreprise choisisse de partager ses préoccupations entre tous les membres des équipes IoT afin de créer une base de connaissances communes pour les risques et les stratégies de remédiation. Ensemble, nos entreprises peuvent construire à la fois la technologie et les connaissances pour s'entraider dans la construction de la sécurité dans l'avenir de l'IoT.

Annexe A Exemple d'utilisation de la technologie GBA

Le niveau de sécurité global d'un réseau multi-sauts est défini par le maillon le plus faible de la chaîne. Ainsi, le lien local entre le dispositif périphérique IoT et une passerelle doit être sécurisé avec un niveau de sécurité comparable à celui du réseau étendu pour conserver le même niveau global de sécurité.

Une technologie candidate pour y parvenir est l'architecture GBA (« Generic Bootstrapping Architecture ») [17] qui peut être utilisée à la fois pour l'authentification et pour l'intégrité des données. Ceci est basé sur des clefs pré-partagées qui sont ensuite utilisées pour générer des clefs à durée limitée (tokens) en tant que base de l'authentification et du chiffrement.

L'authentification est le processus qui consiste à déterminer si quelqu'un ou quelque chose est, en fait, ce qui ou quoi il déclare qu'il est. Dans l'espace IoT, où des milliards de dispositifs seront actifs, il est primordial de déterminer quel comportement de communication est authentique et de confiance. Le mécanisme établi pour créer cette relation de confiance doit satisfaire à l'exigence d'être évolutif et maintenable. De plus, la variété des services IoT impose l'exigence que le mécanisme d'authentification puisse être adapté pour accommoder ces différents services tout en maintenant une infrastructure commune. Un mécanisme qui a fait ses preuves au fil du temps est l'authentification réseau basée sur la carte SIM. Cette infrastructure d'authentification a le mérite de fournir non seulement une authentification, mais aussi des capacités de chiffrement basées sur des secrets pré-partagés. L'explosion du nombre de dispositifs et la portée mondiale de l'IoT rendent l'utilisation de la carte SIM limitée en raison de l'itinérance du réseau et de la faiblesse de la sécurité de pouvoir retirer physiquement une carte SIM d'un dispositif. L'arrivée de technologies telles que la carte SIM embarquée fournit une infrastructure pratique pour l'authentification basée sur des secrets pré-partagés, améliorant l'authentification réseau basée sur la carte SIM actuelle. De même, la croissance de l'IoT est plus susceptible de se produire sous la forme de réseaux capillaires (le PAN tel qu'illustré dans les exemples de configurations 2, 3 et 4 dans la section précédente de ce document). Ces réseaux capillaires ont une énorme quantité de périphériques connectés à une passerelle. La plupart de ces périphériques sont des dispositifs légers (c'est-à-dire qu'ils ne contiennent pas de SIM ni connectivité Cellulaire). Ces périphériques légers nécessiteront néanmoins des capacités d'authentification et de chiffrement. Dans les réseaux capillaires, la principale responsabilité de l'authentification retombe sur la passerelle, ce qui réduit le nombre de dispositifs complexes basés sur la carte SIM sur l'ensemble du réseau. Cette authentification et cette sécurité doivent être étendues de la passerelle au dispositifs périphériques, créant ainsi un canal sécurisé depuis le dispositif donné vers la plate-forme de services IoT.

L'authentification SIM est destinée à servir une seule application, c'est-à-dire l'authentification d'un périphérique unique pour la connexion au réseau. Les périphériques auront une multitude de services, chacun ayant un besoin différent et exclusif d'authentification. Une infrastructure qui étend l'authentification réseau à plusieurs services est nécessaire. Un cadre conçu à cet effet est l'architecture GBA, celle-ci exploite l'infrastructure basée sur la carte SIM pour générer des clefs partagées sur une base de temps entre les périphériques et les fonctions d'application réseau NAF (« Network Application Functions »). GBA est une méthode d'authentification normalisée par le 3GPP

dans la spécification 3GPP TS 33.220 [17]. La méthode permet l'authentification d'un dispositif avec un abonnement 3GPP à un service. Les tokens d'identification de l'abonnement se trouvent dans l'appareil, généralement stockées sur une carte SIM, telle qu'une carte UICC (« Universal Integrated Circuit Card ») ou comme des tokens d'identification gérées à distance, stockées et gérées sur une carte SIM embarquée (eUICC), comme spécifié par la GSMA pour les cartes SIM embarquées (eUICC) [5].

Les avantages de cette solution sont :

- L'authentification mutuelle basée soit sur PSK uniquement entre un périphérique et une fonction d'application réseau, soit sur une authentification d'équipement client à clef partagée avec authentification NAF basée sur certificat (TS 33.222) [18].
- Les tokens d'identification peuvent être sécurisées dans un environnement sécurisé
- Si une eUICC est utilisé, les tokens d'identification peuvent être modifiées à travers des outils OTA.
- L'évolutivité. La complexité et le coût économique de la maintenance augmentent linéairement avec le nombre de dispositifs, puisque l'authentification est "intégrée" dans la solution.
- L'intégrité des données. Les clefs générées avec une base de temps lors de l'authentification peuvent être utilisées pour établir des tunnels TLS-PSK, ce qui rendra cette connexion très solide et confidentielle.

Annexe B Didacticiel sur l'utilisation des cartes UICC dans un service IoT

L'UICC comme standardisé dans ETSI TS 102 221 est une plate-forme de carte à puce (un élément sécurisé programmable inviolable) fournissant une interface de système de fichiers sécurisé interopérable et un cadre d'application sécurisé aux dispositifs d'hébergement UICC. ETSI TS 102 221 fournit un cadre permettant à un dispositif d'hébergement UICC de découvrir des applications pertinentes sur une UICC, et chaque application UICC correspond à un ensemble connu d'informations de provisionnement et de configuration ainsi que des procédures opérationnelles (telles que l'authentification ou la dérivation de clef) qui peut être supporté par le dispositif d'hébergement en fonction de ses besoins.

Dans le contexte IoT, une UICC peut être disponible dans plusieurs formats de puce et gammes de valeurs environnementales pour l'alimentation comme spécifié dans ETSI TS 102 671. Dans son mode de réalisation le plus simple, l'UICC est généralement détenu par un opérateur réseau et héberge seulement une application d'accès au réseau. 3GPP TS 51.011, USIM selon 3GPP TS 31.102, CDMA CSIM comme spécifié par 3GPP2, WiMAX SIM, etc.). Dans ce cas, l'UICC fournit un support normalisé pour héberger des informations de configuration et de provisionnement de sécurité ainsi que des procédures cryptographiques sur un dispositif mobile pour permettre l'accès au réseau, avec des mécanismes supplémentaires pour gérer le contenu de l'UICC à distance. ETSI TS 102 225 / TS 102 226. L'écosystème du réseau mobile a mis en place des procédures permettant de sécuriser la personnalisation et le déploiement des UICC sous le contrôle de l'opérateur du réseau, aboutissant à l'établissement de clefs symétriques partagées individuelles entre les dispositifs d'hébergement UICC et l'infrastructure.

Une caractéristique importante de la plate-forme UICC est le support de domaines de sécurité isolés qui permettent à plusieurs acteurs dans un écosystème complexe de se voir assigner leur propre zone sur un UICC et de gérer son contenu de manière confidentielle. Cette fonctionnalité est héritée par l'ETSI TS 102 226 de l'spécification de la carte GlobalPlatform[15] « Amendement A ». Par conséquent, dans un contexte IoT, une seule UICC permet à plusieurs acteurs de stocker et d'administrer leurs propres tokens d'identification indépendamment les unes des autres.

En général, une UICC peut contenir plusieurs applications d'accès au réseau (une seule étant active à un moment donné) et potentiellement d'autres applications sécurisant l'accès à des services plus élaborés, comme les applications ISIM pour l'accès IMS (comme spécifié dans 3GPP TS 31.103) ou , dans le cas des services IoT, les applications SM 1M2M spécifiées dans l'Annexe D de oneM2M TS-0003. Une application 1M2MSM peut prendre en charge le provisionnement direct d'informations d'identification de service ou applications IoT dédiées, ainsi que le calcul à partir des informations d'identification d'accès réseau préexistantes sur l'UICC en utilisant le mécanisme GBA spécifié par 3GPP. Il permet en outre à un fournisseur de services IoT de personnaliser les procédures cryptographiques en fonction de ses besoins spécifiques, par exemple pour prendre en charge des mécanismes d'authentification de service spécifiques.

Un seul UICC peut également contenir plusieurs applications 1M2MSM, permettant le déploiement confidentiel de clefs symétriques dédiées à chaque fournisseur de services IoT.

Le propriétaire d'une UICC (généralement un opérateur réseau ou un fabricant OEM dans un contexte IoT) peut partager l'espace UICC avec les fournisseurs de services IoT qui le demandent, afin que la chaîne de personnalisation UICC et l'infrastructure qui permettent le déploiement sécurisé des informations d'accès réseau puissent être exploitées. par les fournisseurs de services IoT pour déployer leurs propres informations d'identification.

Lorsque la sécurité des applications IoT repose sur la cryptographie asymétrique, des applications UICC personnalisées peuvent également être utilisées pour faciliter le déploiement de paires de clefs publiques ou privées, comme cela est nécessaire pour un service IoT spécifique. De telles applications UICC doivent être spécifiées et prises en charge sur les dispositifs d'hébergement par rapport aux besoins spécifiques d'une application IoT.

Annexe C Gestion du document

C.1 Historique du document

Version	Date	Brève description du changement	Autorité d'approbation	Éditeur / Société
1.0	08-Feb-2016	Nouvelle version PRD CLP.13	PSMC	Ian Smith GSMA & Don A. Bailey Lab Mouse Security
1.1	07-Nov-2016	Références au schéma d'évaluation de la sécurité de l'IoT de la GSMA ajoutées. Corrections éditoriales mineures.	PSMC	Ian Smith GSMA
2.0	29-Sep-2017	Références aux ressources de la GSMA pour les réseaux LPWA ajoutées. Corrections éditoriales mineures.	IoT Security Group	Rob Childs GSMA

C.2 Autres informations

Type	Description
Propriétaire du document	GSMA IoT Programme
Contact	Rob Childs – GSMA

Nous avons l'intention de fournir un produit de qualité pour votre usage. Si vous trouvez des erreurs ou des omissions, veuillez nous contacter avec vos commentaires. Vous pouvez nous en informer à prd@gsma.com

Vos commentaires ou suggestions et questions sont toujours les bienvenus.