Lignes directrices de sécurité IoT pour l'écosystème de services IoT

Lignes directrices de sécurité IoT pour l'écosystème de services 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 d'atténuer 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 diréctrices de sécurité IoT» [1] avant de lire les autres documents plus détaillés.

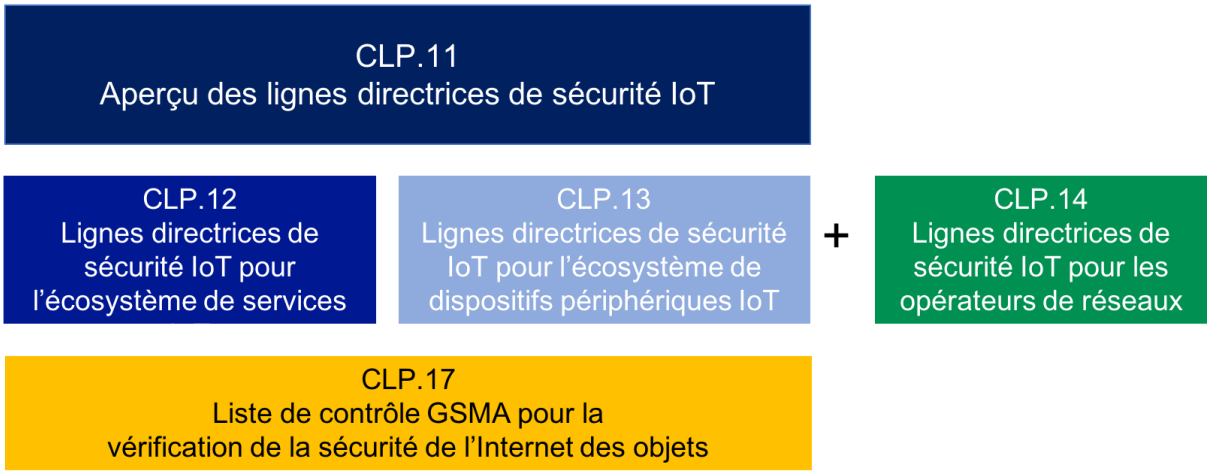

Figure 1 - Structure des Documents sur les Directives de Sécurité de la GSMA

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 de sécurité de haut niveau 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 [13]. Ce document permet aux fournisseurs de produits, services et composants IoT d'autoévaluer la conformité de leurs produits, services et composants 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 [13] 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 guide doit être utilisé pour évaluer tous les composants d'un produit ou d'un service IoT du point de vue de l'Écosystème de Services. L'Écosystème de Services comprend tous les composants qui constituent le cœur de l'infrastructure IoT. Les composants de cet écosystème sont, par exemple, des services, des serveurs, des clusters de bases de données, des éléments de réseau et d'autres technologies utilisées pour piloter les composants internes de tout produit ou service.

La portée de ce document est limitée aux recommandations relatives à la conception et à la mise en œuvre des services IoT et des éléments de réseau.

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
Liste de contrôle d'accès	Une liste d'autorisations attachées à un objet informatique
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.

Terme	Description
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.
Cloud	Un réseau de serveurs distants sur Internet qui hébergent, stockent, gèrent et traitent les applications et leurs données.
Conteneur	Une technologie qui permet d'exécuter plusieurs systèmes ou conteneurs isolés sur un même hôte.
UICC intégré (eUICC)	Un UICC qui prend en charge l'approvisionnement à distance du réseau ou des abonnements au service qu'il authentifie, tel que spécifié par la GSMA.
Client final	Désigne le consommateur du service IoT fourni par le fournisseur de services IoT. Il est possible que le fournisseur de services client final et IoT soit le même acteur, par exemple une entreprise de services publics.
Écosystème des dispositifs périphériques	Toute configuration de périphériques de faible complexité, de périphériques riches et de passerelles qui relient le monde physique au monde numérique de manière novatrice. Voir CLP.11 [1] pour plus d'informations.
Confidentialité Persistante	Une propriété des protocoles de communication sécurisés : Un protocole de communication sécurisé est dit avoir un secret de transfert si la compromission des clés à long terme ne valide pas les clés de session passées.
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.
Dispositif Périphérique IoT	Terme générique désignant un dispositif IoT léger, complexe, une passerelle ou un autre périphérique connecté.
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'informations.
Fournisseur de Service IoT	Entreprises ou organisations qui cherchent à développer de nouveaux produits et services liés à l'Internet des objets connectés.
Opérateur de Réseau	L'opérateur et le propriétaire du réseau de communication qui connecte le dispositif périphérique IoT à l'écosystème de service IoT.
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.
Groupe de Sécurité	Agit comme un pare-feu virtuel qui contrôle le trafic pour une ou plusieurs instances de serveur virtuel.

Terme	Description
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 et de recevoir 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.
UICC	Plateforme 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.
Réseau Privé Virtuel	Partition sécurisée et séparée logiquement d'un réseau pour permettre une utilisation dédiée par un ensemble de services client particulier. Ainsi appelé parce que le VPN est privé du reste du réseau, puis agit comme un réseau virtualisé dans son propre droit

1.5 Abréviations

Terme	Description
3GPP	Projet de Partenariat sur la Troisième Génération (« 3rd Generation Project Partnership »)
ACL	Liste pour le Control d'Accès ("Access Control List")
API	Interface de Programmation d'Applications (« Application Program Interface »)
APN	Nom du Point d'Accès (« Access Point Name »)
CERTS	Équipes d'Intervention d'Urgence Informatique (« Computer Emergency Response Team »)
CLP	Programme de la Vie Connectée (« GSMA's Connected Living Programme »)
DDoS	Déni de service distribué ("Distributed Denial of Service")
GSMA	Association GSM ("GSM Association")
HSM	Module de Sécurité Matérielle ("Hardware Security Module")
IoT	Internet des objets ("Internet of Things")
IP	Protocole d'Internet ("Internet Protocol")
SQL	Langage de requête structuré ("Structured Query Language")
TCB	Base informatique Sécurisée ("Trusted Computing Base")
MV	Machine Virtuelle ("Virtual Machine")
VPN	Réseau Virtuel Privé ("Virtual Private Network")
WAF	Pare-feu d'application Web ("Web Application Firewall")

1.6 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]	n/a	OWASP Secure Application Design Project https://www.owasp.org
[6]	n/a	TCG Trusted Platform Module http://www.trustedcomputinggroup.org
[7]	n/a	TCG Guidance for Securing IoT http://www.trustedcomputinggroup.org
[8]	n/a	OAuth 2.0 http://oauth.net/2/
[9]		OpenID Foundation http://openid.net/foundation/
[10]	n/a	GSMA Mobile Connect https://mobileconnect.io/
[11]	GPC_SPE_034	GlobalPlatform Card Specification www.globalplatform.org/specificationscard.asp
[12]	GPD_SPE_010	GlobalPlatform TEE Internal Core API Specification www.globalplatform.org/specificationsdevice.asp
[13]	CLP.17	GSMA IoT Security Assessment Checklist https://www.gsma.com/iot/iot-security-assessment/
[14]	n/a	ETSI TC SmartM2M specifications www.etsi.org
[15]	n/a	oneM2M Specifications www.onem2m.org
[16]	3GPP TS 33.220	Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA) www.3gpp.org

2 Le modèle de l'écosystème de services

Les produits et services modernes de l'Internet des Objets nécessitent d'un Écosystème de Services pour donner du sens, de la fonctionnalité et de la valeur aux Dispositifs Périphériques, aux Partenaires et aux Utilisateurs. En fonction de la complexité des applications mises à disposition par l'offre IoT, l'infrastructure peut être vaste et inclure de nombreux types de services et de points d'accès aux services variés. D'un autre côté, l'infrastructure peut être rudimentaire pour des applications plus simples.

Quel que soit le format, l'Écosystème de Services agit comme le lien entre la fonctionnalité et la communication pour chaque facette principale de la technologie IoT globale. Tous les autres écosystèmes dépendent de l'écosystème de services pour l'authentification hiérarchique, la connectivité aux utilisateurs, la disponibilité, la gestion et d'autres tâches essentielles au fonctionnement quotidien de l'Internet des Objets. Pour accomplir ces tâches, l'écosystème de services est composé de n'importe quel nombre de couches requises pour atteindre les objectifs de l'infrastructure. Les clusters de bases de données, les serveurs d'applications, les serveurs proxy d'applications et d'autres types d'infrastructure sont des exemples que l'on trouverait dans de nombreux déploiements donnés. Comme l'indique le diagramme ci-dessous, les écosystèmes de réseaux et de Dispositifs Périphériques dépendent des fonctionnalités de base de l'Écosystème de Services.

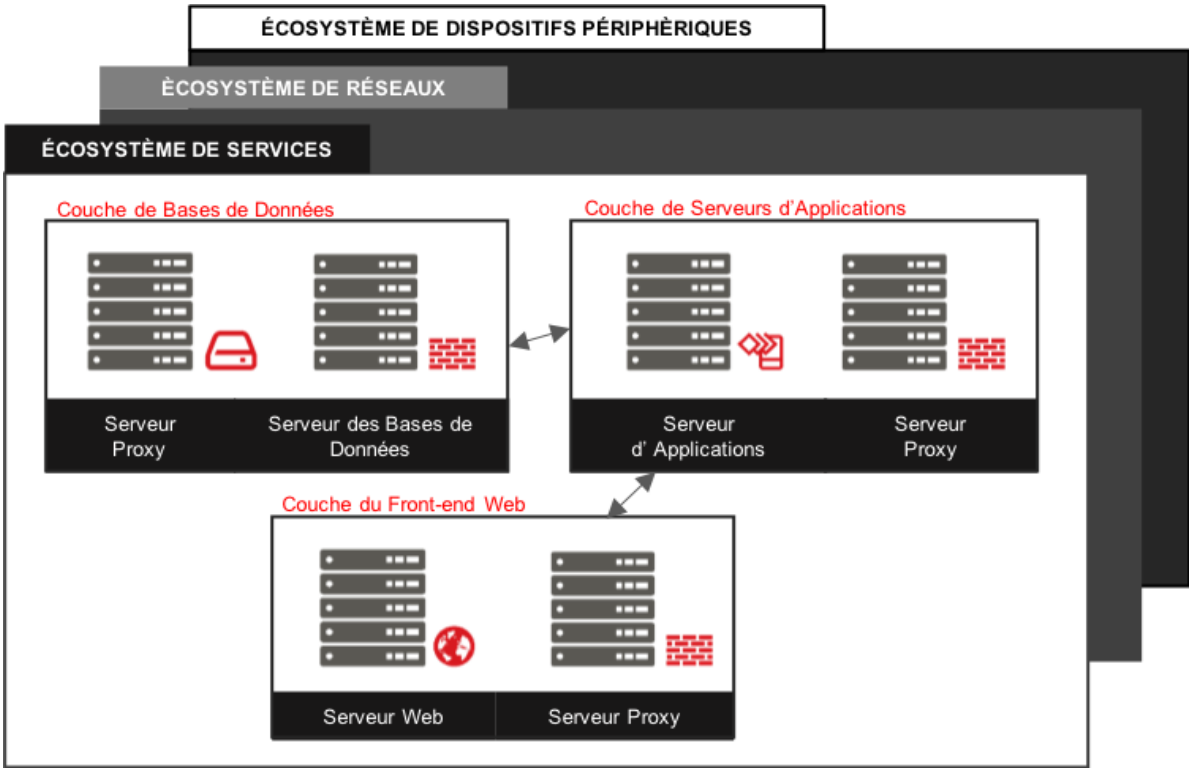

Figure 2 - Dépendances liées à l'écosystème de services

Quelques exemples d'écosystèmes de services modernes incluent, mais ne sont pas limités à :

- Solutions basées sur l'infrastructure cloud
- Déploiements d'applications basés sur des conteneurs
- Environnements de serveurs de centre de données traditionnels
- Clusters de bases de données
- Clusters de services de cadre d'application Web

Bien que chacun de ces exemples d'environnements puisse sembler très différent dans leur conception, leur topologie et leur implémentation, ils sont basés sur les mêmes théories concernant la manière dont les informations circulent vers et hors d'une application.

Tous les systèmes informatiques modernes nécessitent un point d'entrée, appelé point d'accès au service, dans l'infrastructure d'une application. Les sous-systèmes internes qui créent le contenu et le contexte pour cette application doivent être capables de traiter les données à partir d'environnements et réseaux internes, sécurisés et fiables. Les données doivent être stockées quelque part, puis renvoyées à la couche de services qui répond ou envoie des commandes autorisées à divers composants dans le même écosystème, ou d'autres écosystèmes et leurs réseaux associés.

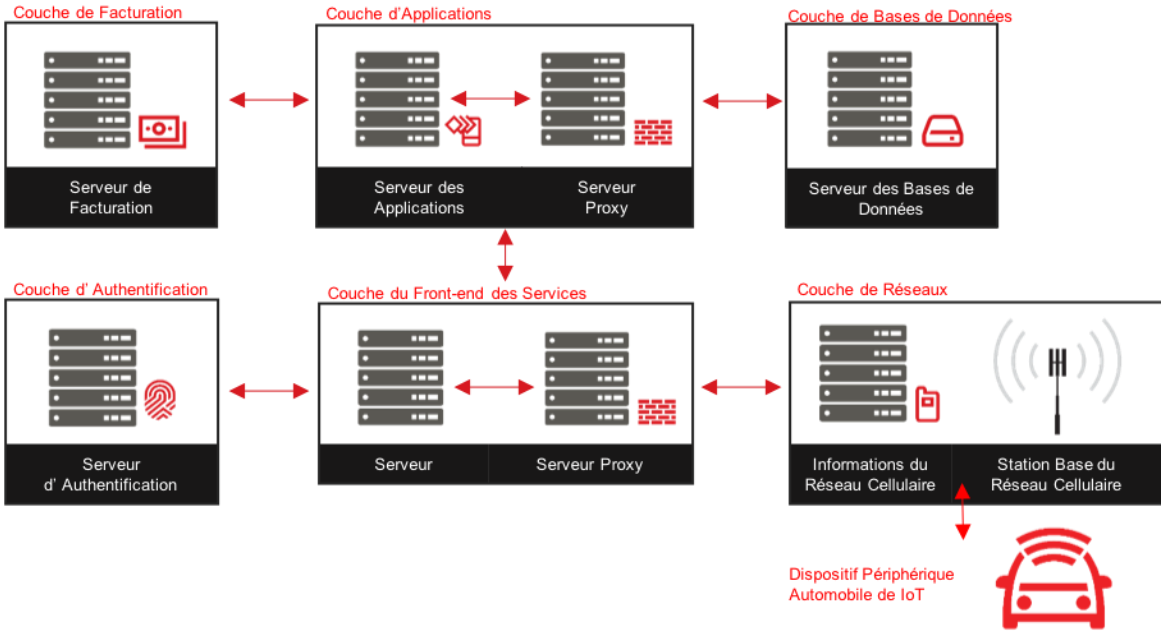

Figure 3 - Exemple d'un écosystème de services

Quelles que soient les technologies, modernes ou traditionnelles, utilisées pour mettre en œuvre ce cadre standard, les informations seront traitées, servies et authentifiées à l'aide de protocoles et de technologies éprouvés. Alors que les topologies et les abstractions pour les environnements de traitement ont subtilement évolué pour s'adapter aux exigences modernes en matière de vitesse, de puissance de calcul et de stockage, les technologies utilisées pour mettre en œuvre ces innovations sont fondamentalement identiques. Par exemple, chaque couche contient généralement un proxy ou un système de pare-feu qui gère la connectivité vers et depuis un ensemble de serveurs d'un type spécifique. Les services de facturation se trouvent dans une couche de facturation. Les serveurs d'applications résident dans une couche spécifique aux applications. Les services de bases de données doivent être gérés dans une couche de bases de données. Ces systèmes fonctionnent tous en fonction des règles d'entrée et de sortie appliquées aux serveurs proxy.

Par conséquent, le modèle de sécurité de l'Écosystème de Services peut être facilement décomposé en un ensemble de composants. Ces composants seront discutés dans ce document.

3 Le modèle de sécurité

La sécurité dans les environnements de services pour les dispositifs périphériques peut être conçue en utilisant des éléments communs d'infrastructure, de stratégies, et de politiques, quelle que soit la topologie ou les innovations utilisées pour créer une architecture d'application. Chaque aspect de l'Écosystème des Services peut être décomposé en composants. Ces composants doivent être sécurisés individuellement, mais en utilisant des méthodologies similaires.

Par exemple, considérez les composants communs dans la création d'un service simple capable de répondre aux requêtes et d'envoyer des réponses depuis et vers les dispositifs périphériques, les partenaires et les utilisateurs. Ce modèle devrait contenir, sans s'y limiter, les couches suivantes :

- Une couche de services Web
- Une couche de serveurs des applications
- Une couche de bases de données
- Une couche d'authentification
- Une couche de réseaux
- Des Couches d'applications tierces, tels qu'une couche de facturation

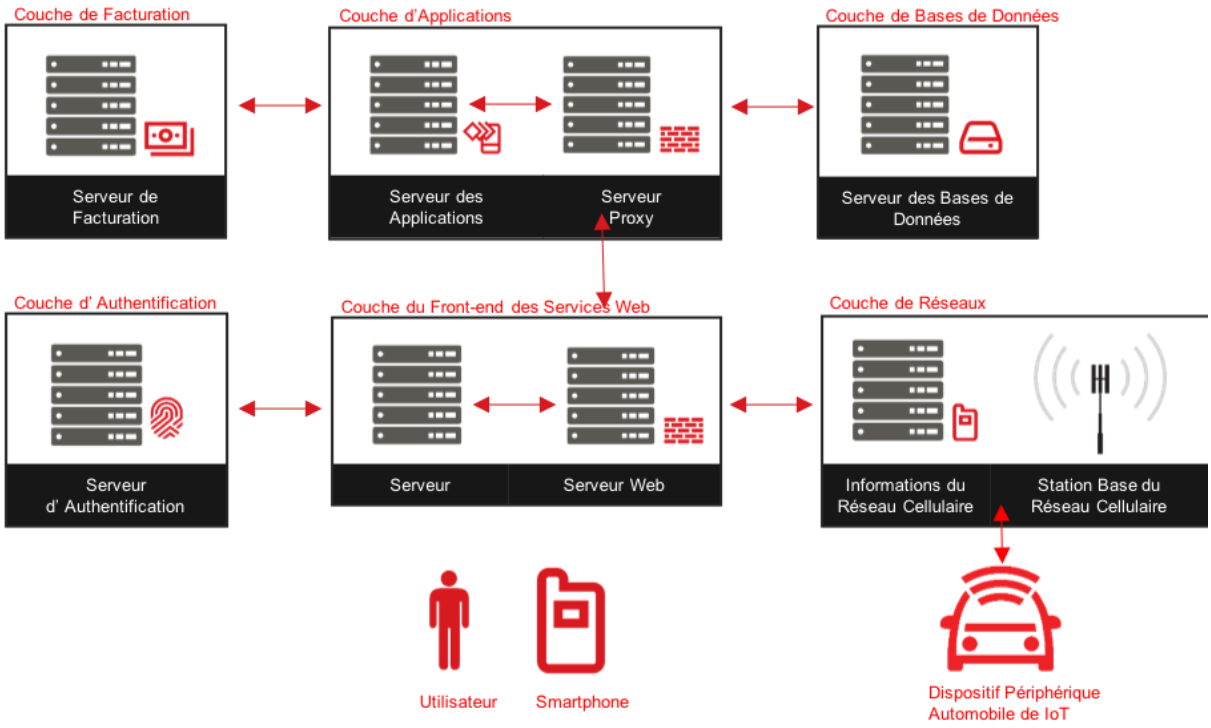

Figure 4 - Un Exemple d'écosystème de services avec des couches séparés.

Même s'il n'y a qu'un seul serveur dans chaque niveau, il est plus efficace sur le plan de l'architecture générale de séparer chaque concept logique dans sa propre couche. Cela permet également d'isoler une couche de technologie d'autres couches en cas de compromission ou si le système doit augmenter ses capacités pour répondre à plus de demandes.

Si un type de système est envisagé du point de vue d'un *type de couche*, il peut être plus facilement sécurisé, dimensionné sur demande, démanteler et remplacer. La seule exigence est une API suffisamment polyvalente pour être améliorée ou ajustée tout au long de la durée de vie de la couche. La définition de cette API est hors de la portée de ce document. Cependant, les recommandations concernant les attributs de sécurité de haut niveau de l'API que l'organisation choisit ou définit seront discutées ici.

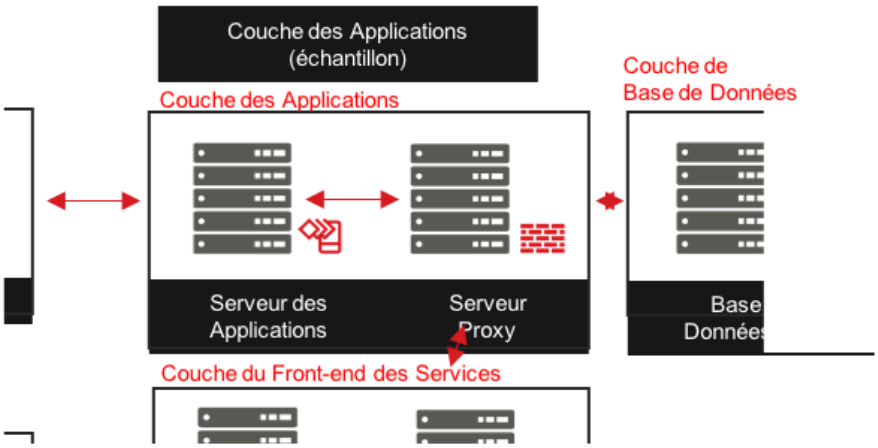

Figure 5 - Un niveau d'application protégé par la technologie de pare-feu

Dans l'exemple ci-dessus, une description de couche légèrement plus complète est fournie. La seule modification extra nécessaire pour représenter la couche dans ce cas est un serveur proxy. Ce serveur proxy est juste un descripteur représentant la technologie de sécurité actuelle qui sera utilisée dans la couche. Que le contrôle actuel soit un pare-feu HW, un pare-feu SW, des groupes de sécurité, des listes de contrôle d'accès (ACL) ou n'importe quelle autre technologie, il existe un composant qui commande les contrôles d'entrée et de sortie pour le compte de la couche en concret.

Lors du choix ou de la définition d'une API, l'organisation doit prendre en compte les spécifications existantes susceptibles de résoudre les problèmes de l'équipe d'ingénierie. L'organisation devrait notamment prendre en compte les spécifications suivantes :

- ETSI SmartM2M TS 102 690, ETSI SmartM2M TS 102 921 [14]
- oneM2M TS-0001, oneM2M TS-0003 [15]
- 3GPP TS 33.220 [16]

Pour les composants accessibles au public, tels que la couche du Front-End, la seule amélioration de l'architecture pour résoudre les besoins du modèle est d'ajouter un composant de sécurité pour :

- Protection DDoS (« Distributed Denial of Service »)
- Équilibrage de charge
- Redondance
- Possibilité de pare-feu d'applications Web (WAF) en option

Les technologies ci-dessus doivent être mises en œuvre pour que tout service fonctionne correctement et pour garantir que le service qu'elles protègent soit rendu disponible même

dans les environnements les plus limités en ressources. La définition de ces composants est hors de la portée de ce document, mais peut être approfondie en se référant aux entités suivantes :

- La Cloud Security Alliance
- NIST Cloud Computing Standards
- FedRAMP
- Guide de gestion de réseau Cisco

Les autres attributs requis pour qu'une couche fonctionne de manière sécurisée sont la définition du/des serveur/s lui/eux-même. Ceci est défini par des contrôles d'administration, d'application et de système d'exploitation internes à la plate-forme choisie par l'équipe d'ingénierie.

Bien que non exhaustive, une liste de problèmes internes à l'environnement de la plateforme sera :

- Connexion à un service de journalisation centralisé
- Authentification administrative et autorisation
- Application de la sécurité des communications
- Sauvegarde, restauration et duplication de données
- Séparation des tâches d'application
- Surveillance du système et intégrité

3.1 Attaques contre l'infrastructure des réseaux

Les adversaires qui tentent de compromettre un Dispositif Périphérique du service du point de vue du réseau supposeront qu'il existe des faiblesses dans la façon dont les entités se communiquent, et des vulnérabilités dans les services exposés à travers les points d'accès au service. Ces attaques supposent qu'une position privilégiée sur le réseau équivaut à une position de pouvoir sur le canal de communication.

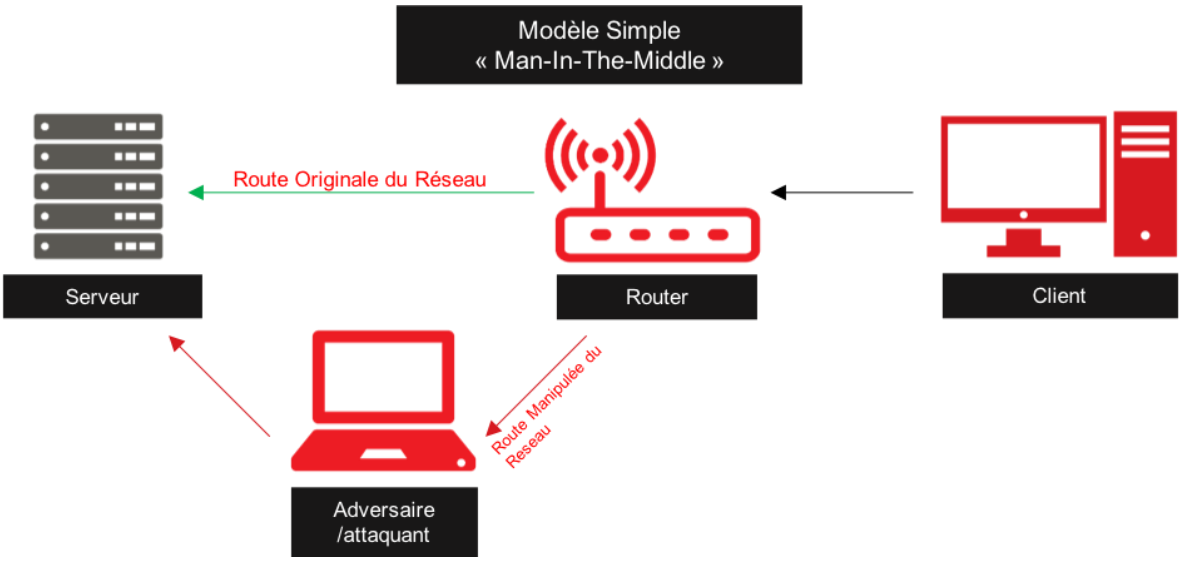

Figure 6 - Un exemple du modèle d'attaque de l'homme du milieu, « Man-In-The-Middle » (MITM)

La forme d'attaque la plus courante dans ce modèle est l'attaque de « l'Homme du Milieu » (MITM). Cette attaque suppose qu'il n'y a pas d'authentification entre les pairs qui se communiquent, d'authentification entre les pairs unilatérale ou d'authentification mutuelle rompue sur le canal de communication. L'objectif d'un adversaire est d'usurper l'identité d'un côté de la conversation pour forcer le pair à effectuer des actions pour le compte de l'adversaire. Cette attaque peut être évitée en imposant une authentification mutuelle, ce qui nécessite une Racine de Confiance Organisationnelle bien définie, une Base Informatique Sécurisée (TCB) et un modèle de communication.

D'autres attaques sont, par exemple, des attaques contre la confidentialité des données, l'analyse des communications de cryptage et les attaques par canal latéral. Ceux-ci doivent être atténués en utilisant des protocoles de cryptographie, des algorithmes et des normes appropriés.

Ces attaques sont difficiles et nécessitent l'accès à l'infrastructure du réseau, soit en interne à une organisation, dans l'infrastructure Internet de base entre une organisation et ses partenaires ou l'Écosystème de Dispositifs Périphériques, ou à l'infrastructure près de ces dispositifs. L'attaque la plus simple et la plus courante consiste à tenter de manipuler l'infrastructure réseau du dispositif périphérique, par exemple le réseau Wi-Fi, Ethernet ou cellulaire, pour obtenir une position de privilège entre le service et son homologue.

Les attaques contre l'infrastructure d'un seul dispositif périphérique sont limitées à celui-ci ou au groupe de points de terminaison disponibles dans cet emplacement physique. Les attaques contre l'infrastructure Internet de base impliquent généralement le détournement du protocole BGP ("Border Gateway Protocol"), l'attaque d'un routeur principal ou l'utilisation abusive de l'infrastructure DNS (Domain Name Service). Ces attaques fourniraient une position de privilège plus dissociée d'une cible particulière, permettant potentiellement à l'attaquant d'avoir accès à de nombreux systèmes cibles à la fois. Les attaques contre une infrastructure de réseau interne nécessitent un accès au réseau interne, ce qui implique une attaque interne ou une position de privilège existante dans l'environnement d'une entreprise, ce qui peut impliquer un compromis système plus profond.

Quel que soit le type d'attaque utilisé, ce modèle est facile à remédier en utilisant l'authentification mutuelle, le secret de transmission en avant et les protocoles et algorithmes cryptographiques appropriés. Cela annulerait la capacité d'un attaquant à abuser de cette infrastructure, ou ferait grimper le coût de ce type d'attaque à un niveau de prix impossible à implémenter par l'attaquant commun.

3.2 Attaques contre l'infrastructure du Cloud ou des conteneurs

Ces attaques supposent une position de privilège sur l'environnement d'infrastructure Cloud ou de Conteneur. Par exemple, si un adversaire est capable de compromettre un réseau de service Cloud, il peut avoir accès à des serveurs exécutant des systèmes de machine virtuelle (MV) invités. Cela permettrait à l'adversaire d'inspecter et de modifier les systèmes MV en cours d'exécution. L'adversaire peut avoir des objectifs spécifiques en tête, ou peut-être avoir eu de la chance et compromis un fournisseur de services Cloud juste pour l'accès à de nombreux types de systèmes avec des données précieuses.

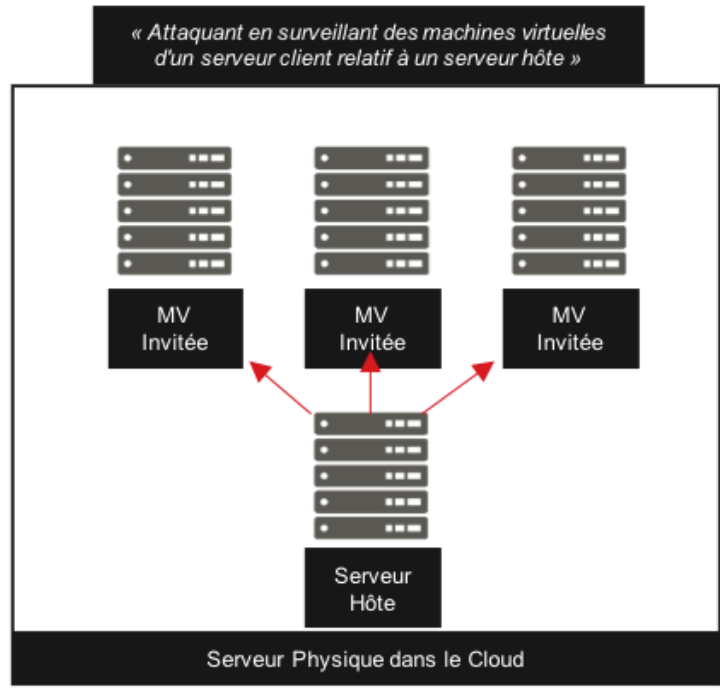

Figure 7 – Exemple d'un modèle d'attaque sur des machines virtuelles

Une autre attaque d'infrastructure Cloud ou Conteneur suppose que l'adversaire contrôle une machine virtuelle sur le même serveur physique que la machine virtuelle cible. L'adversaire peut alors utiliser plusieurs méthodes pour compromettre d'autres machines virtuelles sur un serveur physique. Ils pourraient :

- Utiliser une vulnérabilité dans l'infrastructure MV pour rompre la sécurité d'un invité et attaquer le serveur principal.
- Utilisez une attaque par canal latéral pour déduire les clés secrètes d'une autre machine virtuelle invitée.
- Consommer des ressources excessives sur le serveur physique pour forcer une machine virtuelle cible à migrer vers un serveur physique sur lequel l'attaquant a plus de contrôle.

Peu importe le modèle d'attaque utilisé, il y a peu de choses qu'une entreprise puisse faire pour se prémunir contre ce risque. Au lieu de cela, le fournisseur de services Cloud doit implémenter des fonctionnalités adéquates pour réduire la probabilité qu'un attaquant puisse subvertir l'infrastructure Cloud ou de Conteneur.

Une façon de réduire ce risque consiste à implémenter une architecture basée sur un conteneur qui limite chaque conteneur à un utilisateur spécifique et à une identité cryptographique unique. Bien que cette activité soit très gourmande en ressources et qu'elle entraîne des coûts supplémentaires, elle réduit la possibilité pour un adversaire d'abuser de l'infrastructure de la machine virtuelle, d'accéder à plusieurs utilisateurs ou à plusieurs services simultanément.

Alors qu'une position de privilège dans un environnement Cloud ou de Conteneur est une menace critique pour les applications s'exécutant dans les machines virtuelles invitées, il

faut un haut degré de compétence, de temps et de ressources pour accéder à cette position. Une fois l'accès acquis, l'adversaire doit le maintenir suffisamment longtemps pour identifier quel système contient la MV qui correspond à ses intérêts. En outre, ils doivent pouvoir surveiller ou modifier cette machine virtuelle sans être détectés par le sous-système d'incident du fournisseur de services Cloud. Cela peut représenter un défi important et devrait diminuer la probabilité d'un compromis.

Cependant, il est à noter que ce type de compromis est largement indétectable par la machine virtuelle invitée ou par une application qui s'exécute dessus. Ainsi, des données peuvent être collectées qui révèlent des anomalies dans le comportement d'une MV ou d'un conteneur Cloud en particulier, mais il peut être extrêmement difficile d'identifier si un compromis s'est effectivement produit ou non. En effet, tout adversaire ayant des privilèges suffisants sur la couche de l'hôte de l'infrastructure MV serait capable de manipuler l'invité pour qu'il soit difficile de détecter une manipulation.

Les attaques d'un invité à un autre sont exceptionnellement difficiles à détecter, même par le fournisseur de services Cloud. Il est important de noter, cependant, que ces attaques sont en grande partie de nature théorique. Bien que les attaques par canal latéral soient possibles, elles sont sujettes à discussion car elles nécessitent un niveau de cohérence dans la plate-forme d'exécution sous-jacente qui n'est pas garanti dans un environnement réel. En outre, les attaques par escalade des invités vers les hôtes dans un environnement de machine virtuelle, de conteneur ou d'hyperviseur sont difficiles à trouver et encore plus difficiles à exploiter. Cela rend beaucoup moins probable qu'une vulnérabilité entraîne l'exploitation d'une quantité massive d'invités ou d'une cible spécifique.

Par conséquent, même s'il s'agit d'une position de privilège importante pour les attaquants, la probabilité d'une attaque réussie est faible car la difficulté, le coût et l'opportunité rendent l'exploitation pratiquement impossible.

3.3 Attaques contre les services des applications

Bien que les discussions sur l'architecture d'exécution des applications soient largement hors sujet par rapport à ce document, il est important de noter que cette couche représente le plus grand risque d'attaque. Si l'écosystème de services a été configuré correctement, comme cela est recommandé dans ce guide, les attaquants migreront des attaques d'infrastructure réseau vers l'application elle-même.

L'application présente la plus grande couche de complexité dans tout produit ou service, et contient toujours le potentiel pour un adversaire d'augmenter leurs privilèges à travers plusieurs niveaux de technologie. Par conséquent, bien que le but de ce document soit de détourner l'attention de l'infrastructure du réseau, l'accent est mis sur le seul endroit où le succès est beaucoup plus probable.

Pour réduire le risque d'attaque, veuillez passer en revue bon nombre de documents bien documentés sur la sécurité des applications (par exemple le projet de conception d'applications sécurisées OWASP [5]), pour implémenter l'architecture d'exécution de l'application aussi sûrement que possible.

3.4 Protection des renseignements personnels

Alors que les systèmes partenaires sont conçus pour consommer des données et métriques ou d'autres informations du système IoT centrés sur l'utilisateur pour apporter une valeur ajoutée à l'ensemble du système, il n'y a jamais de garantie quant au niveau de sécurité mis en œuvre par le partenaire. Plutôt que de simplement transmettre des informations à un tiers, il est nécessaire d'évaluer quels types de données doivent être transmises, quel doit être le retour tangible et comment ces informations doivent être protégées.

La responsabilité légale peut être diminuée par des contrats et des clauses d'assurance, cependant, la perte de clients peut survenir en raison de l'échec d'un tiers. Plutôt que de risquer une telle perte d'activité, une organisation doit évaluer les équipes d'ingénierie tierces pour déterminer le niveau de sécurité qu'elles appliquent à leur infrastructure, leurs applications et leurs API. Si le niveau de sécurité n'est pas suffisant, il est recommandé de rechercher des partenaires alternatifs.

3.5 Objets malveillants

Les systèmes tiers sont conçus pour présenter de l'information ou du contenu multimédia aux consommateurs. Une façon évidente d'accomplir ceci est à travers la publicité. Différents types de fichiers sont complexes dans leur structure et sont difficiles à analyser par le logiciel. Les réseaux publicitaires sont un canal intéressant pour la distribution de logiciels malveillants. Les réseaux de distribution de contenu (CDN) représentent également des canaux potentiels pour la distribution de logiciels malveillants. Tout système qui offre des types multimédias complexes, ou des ensembles de codes (que ce soit sur le Web ou exécutables) dans le but de rendre des informations dynamiques, peut transmettre des logiciels malveillants.

Par conséquent, il est impératif que l'entreprise évalue les différents types d'offres technologiques qui seront transmises par un canal particulier. L'entreprise doit décider de ce qu'il faut autoriser et de ce qui est trop excessif pour pouvoir le transmettre à ses clients. Par exemple, une entreprise de publicité peut souhaiter acheminer le code Java vers les systèmes clients via une application de service proxy offerte aux partenaires par l'entreprise IoT. L'entreprise devra décider si les systèmes clients s'exécutant sur certains environnements sont plus susceptibles d'être attaqués à travers la technologie Java. Si cela s'avère être vrai, l'entreprise peut désapprouver Java, mais autoriser d'autres technologies, telles que le langage HTML (« Hypertext Mark-up Language »).

Étant donné que les malwares se présentent sous de nombreuses formes, allant des types de fichiers polymorphes aux exploits Adobe Flash, Java et multimédia, il n'existe pas de moyen unique et uniforme de garantir la sécurité de l'utilisateur final. Une solution simple consisterait pour l'équipe d'ingénierie à appliquer une politique sur les technologies utilisées sur leurs canaux et sur la façon dont leurs utilisateurs seront touchés. Des sous-systèmes de surveillance peuvent être mis en place, ainsi que des “sandbox”, pour s'assurer que tout objet rendu sur un système client est moins sujet à des abus.

3.6 Authentification et autorisation

Les partenaires proposent souvent des services spécifiques à un sous-ensemble d'utilisateurs. Cela peut inclure des services payants auxquels un utilisateur peut éventuellement s'abonner. Cela peut également représenter un moyen qu'un utilisateur peut

utiliser pour s'authentifier sur le système, mais en utilisant des informations d'identification partagées avec une technologie distincte bien connue, telles que les API d'authentification existantes des fournisseurs de services réseau, des infrastructures de réseaux sociaux et des entités de gestion M2M ou IoT existantes.

Bien qu'il s'agisse d'excellents moyens de partager la technologie entre plates-formes, les ingénieurs doivent s'assurer que la technologie ne consomme pas par inadvertance des informations d'identification pouvant être utilisées pour abuser des clés d'autorisation non expressément accordées à un service tiers. Par exemple, certaines API de plate-forme permettent de restreindre les autorisations à une classe acceptée ou refusée par l'utilisateur. Cela permet à l'utilisateur d'adapter l'expérience à celle qui convient à ses besoins spécifiques de confidentialité. Si la plate-forme est incapable d'offrir des autorisations de sécurité granulaires, elle doit répertorier les technologies auxquelles elle souhaite accéder.

Il est nécessaire que l'équipe d'ingénierie demande à ses partenaires que l'offre active des autorisations granulaires pour s'assurer que la révocation d'un service n'autorise pas par inadvertance une fenêtre d'exposition des données des utilisateurs qui continue même après la révocation de l'abonnement.

3.7 Faux positifs et faux négatifs

Bien que les services de surveillance et de consignation de données soient des moyens exceptionnels pour améliorer une infrastructure de sécurité existante, ils doivent être soigneusement évalués pour les faux positifs et les faux négatifs. Étant donné que ces systèmes interprètent uniquement les données provenant de divers écosystèmes au sein d'un produit ou d'un service IoT, et que ces systèmes ne sont pas développés par l'équipe d'ingénierie interne, ils ne peuvent qu'offrir un aperçu artificiel d'un événement. Ils peuvent cependant ne pas être en mesure de distinguer avec précision si un événement avec un objectif malveillant se produit réellement.

En conséquence, il est important de sonder les équipes informatiques et d'ingénierie pour déterminer si un événement suspect est, en fait, attribuable à un comportement malveillant. Cela aidera à annuler le potentiel pour l'équipe de surveillance d'interdire l'accès d'un utilisateur légitime au système. Si ce processus est automatisé et que le processus est incorrect, de nombreux utilisateurs peuvent être exclus de leur service légitime en raison d'un faux positif pouvant être attribué à une anomalie dans l'application ou l'infrastructure du client. Lorsqu'un événement critique est douteux, les équipes informatiques et d'ingénierie doivent examiner les données pour évaluer s'il y a bel et bien une attaque.

De plus, les ingénieurs doivent prendre soin de modéliser les informations acquises par les canaux analogiques. Les faux positifs et les faux négatifs, en particulier dans les écosystèmes où les données doivent être traitées à des taux exceptionnellement élevés, peuvent avoir des conséquences importantes si l'application n'évalue pas correctement les mesures les plus sûres dans le cas où les données acquises ne peuvent pas être entièrement fiables. Il convient de noter qu'avec suffisamment de temps, de technologie et d'expertise, toutes les données analogiques peuvent être canalisées et traitées par un système numérique.

4 Foires aux questions de sécurité

La sécurité du service 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 des Dispositifs Périphériques et les recommandations pertinentes pour atteindre ces objectifs.

4.1 Comment combattons-nous le clonage ?

La différenciation entre les dispositifs valides fabriqués par le fournisseur de services IoT et les dispositifs qui sont des reproductions ou des «rip-s» (clones) est un exploit. Aucun fournisseur de services IoT ne veut fournir de services pour les dispositifs périphériques non autorisés, car les fournisseurs de services doivent payer pour le temps CPU, la bande passante, le stockage sur disque et d'autres ressources. L'organisation doit payer, quoique l'appareil ait été fabriqué par le fournisseur de services IoT ou non.

En outre, l'organisation doit être capable de discerner si son architecture de dispositif périphérique est compromise. Cela permet à l'organisation de réagir à un périphérique qui a été cloné dans plusieurs instances du même périphérique. Cela pourrait être fait par un fabricant peu scrupuleux, ou un adversaire essayant d'usurper l'identité d'un utilisateur particulier.

Consultez les recommandations suivantes pour obtenir de l'aide sur l'utilisation du service pour lutter contre le clonage :

- Définir une racine de confiance organisationnelle
- Utiliser les services d'authentification réseau
- Forcer l'authentification via l'écosystème de services
- Définir l'authentification et l'autorisation de la couche d'applications

4.2 Comment les utilisateurs sont-ils authentifiés par un dispositif périphérique ?

L'un des concepts les plus importants dans IoT est la séparation de l'authentification de dispositifs périphériques de l'authentification de l'utilisateur. Un dispositif périphérique peut être authentifié par sa base informatique sécurisée, mais la façon dont l'utilisateur est authentifié est un processus différent qui repose sur la TCB du dispositif périphérique pour la sécurité des communications. Ce qui est le plus important dans cette abstraction est l'évaluation de la fiabilité du canal de communication pour l'authentification de l'utilisateur.

Par exemple, si la fiabilité d'un dispositif périphérique est faible car il n'y a pas de TCB ou une implémentation TCB pauvre du point de vue de la sécurité, le mécanisme d'authentification utilisateur qui repose sur le logiciel ou micro-logiciel du dispositif périphérique ne peut pas être approuvé. Cela signifie que tout utilisateur s'authentifiant via un dispositif périphérique ne peut pas être considéré comme authentifié.

D'un point de vue différent, une TCB de dispositif périphérique bien conçu peut mal authentifier l'utilisateur final si le schéma d'authentification est facilement contournable. Ainsi, l'écosystème de services doit s'appuyer sur la fiabilité du dispositif ainsi que sur la

mise en œuvre du mécanisme d'authentification pour s'assurer que l'écosystème de services peut garantir que le bon utilisateur est connecté au système.

Considérez les recommandations suivantes pour vous aider à faire face à ces complexités :

- Implémenter une base informatique sécurisée pour les services
- Définir une racine de confiance organisationnelle
- Définir un modèle d'autorisation clair
- Utiliser les services d'authentification réseau
- Forcer l'authentification via l'écosystème de services
- Appliquer la stratégie de mot de passe fort
- Définir l'authentification et l'autorisation de la couche d'application

4.3 Comment le service peut-il identifier un comportement de dispositif périphérique anormal ?

L'un des aspects les plus difficiles de la gestion des dispositifs périphériques dans un réseau IoT distribué consiste à déterminer s'ils se comportent de manière anormale ou non. Ce n'est pas seulement important du point de vue de la sécurité, mais du point de vue de la fiabilité. Souvent, un comportement anormal peut indiquer un problème avec le micro-logiciel ou le matériel et peut indiquer que l'entreprise doit se préparer à résoudre un problème inattendu. Cependant, si le comportement est isolé dans une partie du réseau qui ne peut pas être analysée par le fournisseur de services IoT, ces mesures seront perdues, ce qui fait perdre à l'entreprise une position beaucoup plus avantageuse.

La résolution de ce problème nécessite la capacité d'inspecter le comportement sur le dispositif périphérique, la couche réseau et l'écosystème de services. Cependant, si la bonne infrastructure, les services et les partenariats ne sont pas conçus pour rassembler ces points de données, l'organisation n'aura pas les informations nécessaires pour déterminer s'il y a un problème ou si un problème est lié à la sécurité ou fiabilité.

Évaluer les recommandations suivantes du point de vue de l'écosystème de services :

- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir une approche de journalisation et de surveillance des systèmes
- Définir un modèle de communication
- Utiliser les services d'authentification réseau
- Implémenter une validation d'entrée
- Mettre en œuvre un filtrage de sortie
- Utiliser les services de surveillance améliorés par les partenaires
- Utiliser un APN privé pour la connectivité sans fil
- Définir un modèle pour l'évaluation des faux négatif ou positif

4.4 Comment le service peut-il restreindre un dispositif périphérique se comportant de façon anormale ?

Une fois qu'un dispositif périphérique est identifié comme se comportant anormalement, le service devrait prendre des décisions quant aux ressources qui devraient être limitées ou restreintes. Cette question est pertinente pour chaque couche de l'infrastructure de services.

Par exemple, un dispositif périphérique cellulaire qui se connecte et se déconnecte constamment du réseau mobile dans une boucle effrénée devrait être désactivé par la force jusqu'à ce que le comportement erratique soit résolu. Un autre exemple utile est un dispositif compromis qu'un adversaire utilise pour tenter d'attaquer les services back-end. Dans ce scénario, les services au back-end doivent empêcher le dispositif périphérique abusif d'atteindre sous aucun prétexte, n'importe quel service.

La gestion de chaque scénario dépend du fournisseur de services IoT et de leurs objectifs commerciaux et de la manière dont les incidents doivent être traités. Pour aider à l'élaboration de ces lignes directrices, tenez compte des recommandations suivantes :

- Définir une racine de confiance organisationnelle
- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir un modèle de réponse aux incidents
- Définir un modèle de récupération
- Définir un modèle de temporisation
- Définir un modèle de communication
- Définir une politique de violation pour les données exposées
- Forcer l'authentification via l'écosystème de services
- Utiliser un APN privé pour la connectivité sans fil
- Définir un modèle pour l'évaluation des faux négatif ou positif

4.5 Comment peut-on déterminer si un serveur ou un service a été piraté ?

Alors que les anomalies des dispositifs périphériques sont plus ésotériques et nécessitent une grande quantité d'analyses comportementales pour détecter la majorité des attaques, l'écosystème de services est plus simple. Les services et les serveurs sont déployés dans un environnement étroitement contrôlé par le fournisseur de services IoT ou par leurs partenaires qui gèrent l'infrastructure de cloud ou de serveur. Ainsi, l'organisation et ses partenaires peuvent utiliser des systèmes de surveillance et de diagnostic facilement disponibles pour identifier et contenir les problèmes potentiels.

Passez en revue les recommandations suivantes pour obtenir de l'aide :

- Définir un modèle d'administration
- Définir une approche de journalisation et de surveillance des systèmes
- Définir un modèle de réponse aux incidents
- Implémenter la validation d'entrée
- Mettre en œuvre le filtrage de sortie

4.6 Que peut-on faire une fois qu'un serveur a été piraté ?

Lorsqu'un serveur a été identifié comme compromis, l'équipe d'administration doit résoudre le problème aussi rapidement et efficacement que possible. La complexité de ce processus découle souvent de la détermination des ressources, de l'information et des comptes qui ont été mis en péril. Dans certains environnements mal architecturés, les effets d'un compromis ne sont pas souvent quantifiables. Par conséquent, l'organisation doit mettre en œuvre un plan pour résoudre la vulnérabilité de sécurité et sécuriser les actifs à risque sur le terrain, en parallèle. Une fois l'écosystème et la vulnérabilité sécurisés, l'organisation peut procéder à un plan de reconstruction de la technologie concernée.

Passez en revue les recommandations suivantes pour plus d'informations :

- Définir un modèle de réponse aux incidents
- Définir un modèle de récupération
- Définir un modèle de temporisation
- Définir un ensemble de classifications de sécurité
- Définir des classifications pour les ensembles de types de données

4.7 Comment les administrateurs doivent-ils interagir avec les serveurs et les services ?

Le développement d'un modèle administratif qui ne met pas en péril l'écosystème de services est une partie importante de l'architecture d'un service IoT. Il existe plusieurs couches d'administration et chaque couche doit être prise en compte par les équipes d'ingénierie et de sécurité. Par exemple, les administrateurs qui administrent le serveur (indépendamment de l'utilisation d'une architecture virtuelle, micro-service ou uni-noyau) doivent pouvoir interagir avec des serveurs actifs via un canal de communication fiable et sécurisé. Les administrateurs qui régissent l'application Web interagissent souvent avec l'application sur la même couche de communication Web, mais via une application spécialisée intégrée dans le code.

Indépendamment du besoin administratif, l'interface doit être restreinte pour limiter la capacité des adversaires à interagir avec la technologie ou à en abuser. Considérez les ressources suivantes :

- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir un modèle d'administration
- Définir un modèle d'autorisation clair
- Définir un modèle de communication
- Utiliser un APN privé pour la connectivité sans fil

4.8 Comment l'architecture de service peut-elle limiter l'impact d'une compromission ?

Un attribut fascinant d'un réseau IoT est sa capacité unique à attacher des services à des consommateurs spécifiques. Dans les services Web, chaque utilisateur doit avoir la possibilité d'interagir avec le service à partir de n'importe quel type d'appareil ou, potentiellement, de partout dans le monde. Ce n'est pas vrai pour la technologie IoT. La technologie IoT nécessite généralement un dispositif périphérique spécifique pour interagir avec les services IoT. En raison de cette différence, les architectes d'écosystème de serveurs peuvent exploiter la relation un-à-un entre les dispositifs périphériques et les consommateurs pour restreindre l'accès d'un dispositif aux données principales.

Considérez le scénario dans lequel un dispositif pousse les mesures de capteur vers un service principal. Dans une architecture de micro-service, l'écosystème de services peut déployer un micro-service ou un noyau unique pour gérer un consommateur particulier. En utilisant cette architecture, l'ingénieur peut s'assurer que le micro-service est provisionné uniquement avec les ressources et les capacités d'accès requises pour fournir des données et des services spécifiques au consommateur individuel.

Cela signifie que si un service est compromis et que le dispositif périphérique est la seule technologie capable de communiquer avec ce service spécifique, il n'y a aucun avantage supplémentaire à compromettre ce service car l'accès obtenu à partir du compromis sera limité aux ressources qui seraient déjà disponible pour le dispositif périphérique. En substance, l'attaque ne peut obtenir rien d'autre.

Passez en revue les recommandations suivantes pour obtenir de l'aide :

- Implémenter une base informatique sécurisée pour les services
- Définir une méthode Bootstrap
- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir un modèle de stockage persistant
- Définir un modèle d'administration
- Définir un modèle de temporisation
- Définir un modèle d'autorisation clair
- Provisionner les serveurs là où c'est possible
- Définir un environnement d'exécution d'applications
- Compromis de machine virtuelle

4.9 Comment l'architecture de service peut-elle réduire la perte de données en cas de compromission ?

Un autre attribut intéressant de l'architecture IoT est la réduction de la perte de données. Ceci est similaire à la façon dont les services peuvent être isolés pour un utilisateur spécifique. Les données peuvent également être isolées à un utilisateur spécifique une fois que l'utilisateur a été authentifié. Cependant, le stockage de données ne peut pas être facilement implémenté par utilisateur en raison du coût de l'infrastructure de base de données et de stockage.

Au lieu de cela, les jetons uniques doivent être fournis aux services qui agissent ensuite au nom d'un utilisateur spécifique dans l'infrastructure de stockage. De cette manière, un attaquant ayant accès à l'environnement de stockage de données peut être en mesure de se connecter au service, mais ne devrait pas être en mesure d'interagir avec, de récupérer ou de modifier des données utilisateurs autres que celles de l'utilisateur compromis.

Du point de vue de la couche réseau, la réduction du flux de trafic de l'écosystème du serveur vers Internet est également une exigence. Les contrôles de sortie forcent un adversaire à transmettre des données caractérisées comme propriété intellectuelle ou spécifiques à un client via des canaux spécifiques. Cela peut augmenter la difficulté de déplacer de grandes quantités de données ou de forcer le trafic de données à travers des couches de communication capables de détecter et de couper les communications lors d'incidents.

Pour plus d'informations, considérez les recommandations suivantes :

- Définir une méthode Bootstrap
- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir un modèle de stockage persistant
- Définir un ensemble de classifications de sécurité
- Définir des classifications pour les ensembles de types de données

- Provisionner les serveurs là où c'est possible
- Définir un environnement d'exécution d'application
- Règles de pare-feu par défaut ouvertes ou non ouvertes

4.10 Comment l'architecture de service peut-elle limiter la connectivité des utilisateurs non autorisés ?

Un avantage de tirer parti des architectures IoT courantes est de limiter la possibilité pour les utilisateurs Internet non autorisés de se connecter directement aux services back-end. La plupart des applications web n'ont pas ce luxe, et doivent être disponibles pour un usage public. En IoT, cependant, parce que le dispositif périphérique est l'entité qui doit se connecter à un service particulier, le réseau privé virtuel (VPN) peut être utilisé pour restreindre l'accès aux services dorsaux. Cela peut être implémenté sur des protocoles Internet standard, ou peut-être implémenté en utilisant des services mobiles, tels que l'APN privé. Consultez les recommandations suivantes pour plus d'informations :

- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Utiliser un APN privé pour la connectivité sans fil

4.11 Comment réduire la probabilité d'exploitation à distance ?

L'exploitation à distance des applications et des services Web est une préoccupation constante des administrateurs d'infrastructure. S'assurer que les adversaires n'ont pas de route dans le réseau interne, ou simplement vers des ressources précieuses, est une bataille quotidienne. La seule façon de réduire le risque que des adversaires compromettent l'écosystème de services consiste à réduire les cibles potentielles en un ensemble gérable de services pouvant être rapidement et facilement maintenus. La deuxième plus importante amélioration apportée à l'architecture est la conception de l'architecture sous-jacente : architecture d'exécution, configuration du système d'exploitation, chaîne de déploiement, sécurité du langage de programmation et autres options qui définissent la sécurité d'une application. Ces options peuvent faire la différence entre un plantage d'application et un compromis d'infrastructure.

Pour plus d'informations sur la réduction du potentiel d'exploitation à distance, veuillez consulter :

- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir un modèle de mise à jour
- Implémenter la validation d'entrée
- Mettre en œuvre le filtrage de sortie
- Règles de pare-feu par défaut ouvertes ou non ouvertes
- Définir un environnement d'exécution d'application
- « Rowhammer » et les attaques similaires
- Compromis de machine virtuelle

4.12 Comment le service peut-il gérer la confidentialité des utilisateurs ?

Au fur et à mesure de la croissance des fournisseurs de services IoT, ils établiront invariablement des partenariats avec des organisations qui utiliseront les données des consommateurs de manière innovante. Cependant, ces données ont un coût pour la vie

privée du consommateur. Les consommateurs devraient avoir le droit de déterminer quelles données sont partagées avec les partenaires et comment elles seront utilisées. En outre, les partenaires devraient être tenus d'utiliser les données de manière spécifique. Les modèles d'autorisation peuvent aider à cela, mais cela implique une discussion beaucoup plus large sur la vie privée, les répercussions juridiques, l'assurance des entreprises, et plus encore.

Pour commencer la discussion au sein de votre organisation, veuillez passer en revue les recommandations suivantes :

- Définir un ensemble de classifications de sécurité
- Définir des classifications pour les ensembles de types de données
- Définir un modèle d'autorisation clair
- Définir une politique de violation pour les données exposées
- Évaluer le modèle de confidentialité des communications
- Définir une politique de distribution de données tierce
- Construire un filtre de données tiers
- Créer une API pour les utilisateurs afin de contrôler les attributs de confidentialité

4.13 Comment un service peut-il améliorer sa disponibilité ?

Les attaques par déni de service (DoS) ou par déni de service distribué (DDoS) sont si répandues dans l'Internet moderne que chaque entreprise devrait être prête à faire face à une attaque majeure de cette classe et devrait pouvoir rester en ligne même en cas d'attaques prolongées. La raison pour laquelle ces attaques sont devenues si banales est qu'elles nécessitent que de peu de compétences à exécuter et que les outils pour mettre en œuvre une telle attaque sont facilement accessibles en ligne. En fait, il existe des services en ligne où une tierce partie malveillante peut payer un attaquant pour mettre en œuvre une attaque DDoS contre une cible particulière.

En conséquence, des modèles entièrement nouveaux pour la disponibilité des services ont été construits pour lutter contre cette menace. Tenez compte des recommandations suivantes lors de la création de l'écosystème de services :

- Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public
- Définir une approche de journalisation et de surveillance des systèmes
- Définir un modèle de réponse aux incidents
- Définir un modèle de récupération
- Définir un modèle de communication
- Règles de pare-feu par défaut ouvertes ou non ouvertes

5 Recommandations critiques

Lors du développement d'un dispositif périphérique sécurisé, les recommandations suivantes doivent toujours être mises en œuvre. Les recommandations critiques suivantes définissent une architecture de dispositif périphérique sécurisée. Sans ces recommandations, ce dernier aura un profil de sécurité incomplet qui sera abusé par un adversaire.

5.1 Implémenter une base informatique sécurisée pour les services

Une base informatique sécurisée (TCB) est un ensemble de matériels, de logiciels, de protocoles et de stratégies. Une TCB doit être la base d'une plate-forme informatique donnée et doit définir l'environnement dans lequel une application peut fonctionner de manière fiable, sécurisée et de haute qualité.

Une TCB peut être construit et déployé pour n'importe quelle classe de système, telle que smartphones, terminaux IoT, et même serveurs fonctionnant dans un écosystème de services. Les TCB sont tous construites avec des technologies similaires. Pourtant, selon la classe de système, ces technologies peuvent adopter des caractéristiques très différentes. Par exemple, l'amorçage d'une TCB dans un serveur de cloud sera très différent de l'amorçage dans un dispositif périphérique.

Construire une TCB dans un Écosystème de Services signifie définir la façon dont une image d'application doit être déployée. Une image dans ce contexte représente les données binaires brutes qui comprennent un exécutable d'application, ses fichiers de configuration et ses métadonnées. Ces composants ensemble sont communément appelées une image d'application, ou simplement une image. Dans la plupart des écosystèmes de services modernes, les systèmes seront répliqués, mis sous tension ou adaptés fonctionnellement à la demande pour s'adapter de manière réactive aux changements de l'environnement informatique. Cela signifie qu'une TCB doit définir un moyen de permettre aux systèmes d'évoluer efficacement tout en conservant un modèle de sécurité persistant.

Pour le faire correctement, l'équipe doit :

- Standardiser la plateforme informatique :
  - Choisir un ensemble de modèles de serveurs physiques
  - Sélectionner un ensemble de plates-formes Cloud ou d'images de machine virtuelle (MV)
- Définir l'ensemble des applications, bibliothèques et fichiers de configuration à exécuter sur la plate-forme informatique :
  - Définir un environnement de conteneur, le cas échéant
- Générer une image d'application, composée de l'ensemble défini ci-dessus
- Signer cryptographiquement une archive de l'image à l'aide de la clé de signature Tiers TCB
- Stocker en toute sécurité l'archive et la signature

L'exécution de cet ensemble de tâches se traduira par une image d'application approuvée pouvant être déployée dans une couche spécifique. Chaque couche aura un matériel et un modèle d'application différent qui fonctionnent le mieux pour cette couche spécifique. Par exemple, le matériel de base de données a des besoins de performances et de stockage

très différents de ceux d'une couche d'application. Une couche de stockage aura des exigences de stockage matériel similaires à celles d'une couche de bases de données, mais aura des exigences de performances différentes. Après avoir normalisé la définition de chaque couche, le résultat est une image qui peut être déployée et vérifiée sur chaque plate-forme matérielle.

La difficulté à déployer une TCB vient de :

- Configuration d'une racine de confiance organisationnelle pour gérer la signature cryptographique des images
- Mise en place d'une procédure de signature de chaque image
- Mise en place d'une procédure de vérification de chaque image
- Mise en place d'une procédure de déploiement automatique des images, mais avec vérification d'image

S'il vous plaît envisager d'utiliser le matériel des organisations suivantes pour aider à cette recommandation :

- Spécification de la carte GlobalPlatform [11]
- Spécification TPM du « Trusted Computing Group » [6]
- Spécification de l'API centrale interne de GlobalPlatform TEE [12]

5.1.1 Risque

Sans une TCB bien définie, les plates-formes informatiques sont incapables de vérifier si elles fonctionnent actuellement dans une configuration approuvée par l'équipe d'ingénierie. Ceci est important car le sous-système d'application doit être capable de déterminer s'il a été compromis par un adversaire. Une TCB peut être utilisé pour remédier ce risque et fournir une couche de sécurité pour toutes les communications réseau.

5.2 Définir une racine organisationnelle de confiance

Une Racine Organisationnelle de Confiance est un certificat ou un système basé sur une clé publique pour authentifier des entités de plate-forme informatique dans une organisation. Chaque plate-forme informatique d'un Écosystème de Services doit être authentifiée de manière cryptographique lors des communications réseau. Cela réduit la possibilité pour une personne interne de l'entreprise, ou quelqu'un dans une position de réseau privilégié, d'usurper l'identité ou d'abuser de la confiance d'un système privilégié.

Pour créer une Racine Organisationnelle de Confiance, effectuez simplement les actions suivantes :

- Construire ou acquérir, par exemple, un module de sécurité hardware (HSM) pour stocker le secret racine de l'organisation
- Générer un secret racine et / ou un certificat
- Assurez-vous que la facette privée du secret est stockée en toute sécurité
- Générer un ensemble d'une ou plusieurs clés de signature à utiliser comme clé de signature pour chaque TCB dans une couche en particulier du système
- Signer la facette publique de la clé de signature avec la racine de l'organisation
- Assurez-vous que ces clés ne peuvent pas être utilisées sans l'authentification et l'autorisation des responsables commerciaux et d'ingénierie

Chaque fois qu'une nouvelle couche dans un système est définie, sa clé ou certificat cryptographique unique peut maintenant être signé par la clé de signature. Si un autre système se connecte à ce nouveau système, il peut valider l'identité du système en vérifiant la chaîne de confiance définie par la racine de l'organisation.

Il validera cryptographiquement que les messages ont été signés par la clé publique représentant le système. Ensuite, il vérifiera la signature de la clé de signature générée de la clé publique unique de ce système. Ensuite, le client doit vérifier que la clé de signature était bien la clé de signature authentifiée par la racine de l'organisation.

Parce que chaque ensemble de certificats ou de secrets est limité à de moins en moins d'individus dans l'organisation, et les politiques et procédures définies devraient restreindre qui peut utiliser ces secrets et quand, chaque degré de confiance devrait augmenter au fur et à mesure que le client descendra dans la hiérarchie du système.

Un service doit être défini qui présente des capacités d'authentification aux pairs autorisés au sein de l'Écosystème de Services. Par exemple, l'authentification à l'aide du certificat ou de la chaîne secrète ne peut pas être utilisée seule pour garantir la sécurité. Un service doit être disponible qui vérifie si les certificats ont été révoqués ou sont actuellement valides. Il se peut qu'un autre service doive être utilisé pour authentifier les identités des serveurs ou des services ayant une durée de vie courte, en fonction des besoins de l'infrastructure sous-jacente.

Au cours de la définition de la Racine de Confiance, considérez que :

- Chaque secret doit être protégé contre les abus
- L'utilisation interne de chaque secret doit être suivie et contrôlée de manière vérifiable
- Chaque personne autorisée à utiliser un secret doit utiliser une authentification multifactorielle lors de l'accès au (x) secret (s)
- Définir un ensemble de politiques et de procédures qui imposent une utilisation cohérente et sécurisée peut être difficile
- Construire un processus pour annuler ou révoquer un certificat peut être difficile
- Identifier si une clé a été abusée peut-être difficile
- Le choix du bon ensemble d'algorithmes cryptographiques peut être non intuitif

Pour en savoir plus sur le concept de la Racine de Confiance, veuillez tenir compte des sources d'information suivantes :

- Groupe d´informatique Fiable (« Trusted Computing Group »)
  - Spécification TPM [6]
  - Conseils pour sécuriser IoT TCG [7]
  - ISO 11889
- Spécifications de l'ICP (« infrastructure à Clés Publiques »)
  - RFC 2510
  - RFC 3647

5.2.1 Risque

Le risque de ne pas utiliser une Racine Organisationnelle de Confiance est que tout compromis avec une seule clé peut entraîner une compromission de l'ensemble de l'écosystème. En séparant l'organisation en une hiérarchie et en déployant des clés séparées pour chaque élément de la hiérarchie, les clés peuvent être cyclées à intervalles réguliers et en fonction de la priorité de l'application ou de la sous-organisation à laquelle la clé se rapporte.

5.3 Définir une méthode bootstrap

Pour qu'une application fonctionne correctement, elle doit être chargée et exécutée de manière cohérente sur une plate-forme fiable, de haute qualité et sécurisée. La TCB définit comment formuler cette plate-forme, mais le modèle Bootstrap définit comment l'application doit être exécutée par-dessus.

Pour définir efficacement un modèle Bootstrap, il faut prendre en compte les éléments suivants :

- Définir une API permettant à l'application de s'identifier cryptographiquement à ses pairs
  - Envisagez d'utiliser une API existante définie par un leader du secteur de confiance
- Définir comment l'application authentifiera les dispositifs périphériques, les points de communications entre services et les partenaires
- Définir à quoi devrait ressembler la configuration de l'application
- Assurer que chaque application différente ait une identité unique, en particulier les applications exécutées sur des couches différentes

Bien qu'il puisse sembler intuitif qu'une application doit s'identifier de manière cryptographique à ses pairs, et qu'elle n'a pas besoin d'une API pour le faire, le processus en production n'est pas très intuitif. En effet, dans le modèle bootstrap, il faut considérer comment l'identité cryptographique est provisionnée pour l'application. Comment l'application acquiert-elle son identité? L'identité est-elle acquise en toute sécurité? Quel est le processus de révocation des secrets que l'identité utilisera dans le cas où les secrets doivent être mis à jour ou modifiés?

Lors de l'exécution, les applications requièrent certaines ressources pour s'exécuter efficacement. L'application doit être capable de communiquer et d'effectuer une authentification mutuelle avec tous les services externes, les dispositifs périphériques et les partenaires impliqués dans ce processus.

La configuration d'une application détermine souvent la sécurité qu'elle aura en production. Une configuration doit être appliquée et présentée en mode de « seule-lecture » à une application. L'application, ou quelqu'un abusant de l'infrastructure de l'application, ne devrait pas pouvoir modifier simplement la configuration d'une application.

Utilisez la Racine de Confiance Organisationnelle pour définir des modèles de confiance pour chaque couche déployée dans l'écosystème global. Cela permettra à chaque application distincte d'avoir une identité cryptographique unique. Cela permettra aux homologues de différencier entre un service de base de données et un service d'application, par exemple.

5.3.1 Risque

Sans un modèle Bootstrap bien défini, le système n'aura aucun moyen de vérifier chaque couche dont il a besoin pour fonctionner. En substance, il ne doit pas y avoir de superposition de confiance sur une seule couche dans le système IoT globale. Ce manque de couches de confiance introduit une complexité qui peut entraîner des lacunes qui peuvent être exploitées par les adversaires.

5.4 Définir une infrastructure de sécurité pour les systèmes exposés à l'Internet public

Pour les services accessibles au public, plusieurs éléments de sécurité et de fiabilité sont nécessaires pour maintenir la disponibilité, la confidentialité et l'intégrité du service :

- Infrastructure résistante aux attaques DDoS
- Infrastructure d'équilibrage de charge
- Systèmes de redondance
- Pare-feu d'application Web (facultatif)
- Pare-feu traditionnel

Ces technologies supplémentaires doivent être placées à l'entrée de la couche d'application pour s'assurer qu'elles ne peuvent pas être manipulées par des attaquants publics. Bien que le modèle de sécurité des communications corrigera ou atténuera le risque qu'un tiers anonyme accède au système, ces technologies réduiront la capacité de l'adversaire à rendre le système indisponible.

La sécurité frontale doit être appliquée à tous les protocoles mis en œuvre par les services. Par exemple, si le service est disponible sur IPv4 et IPv6, les mêmes contraintes de sécurité doivent être appliquées au service sur les deux protocoles. Si un service est accessible via TCP ainsi que le protocole SCTP (Stream Control Transmission Protocol), les contraintes de sécurité doivent également être appliquées à ces deux protocoles. Les ports qui n'offrent pas de services publics et utilisés par un produit ou un service IoT ne doivent pas être accessibles.

Assurez-vous que le filtrage d'entrée et de sortie est administré, dans la mesure du possible. Alors que le filtrage d'entrée stoppe toute une série d'attaques, toute attaque contre un service accessible publiquement peut toujours entraîner une compromission de l'écosystème de services. À ce stade, le filtrage de sortie est impératif pour garantir qu'un composant compromis de l'écosystème de services ne puisse pas être utilisé par un attaquant pour se déplacer latéralement dans l'écosystème. En outre, le filtrage de sortie complique la capacité des pirates à exfiltrer les données critiques de l'écosystème vers des serveurs contrôlés par l'adversaire, laissant ainsi plus de temps aux administrateurs pour identifier et isoler l'attaquant.

Plusieurs organisations offrent ces services dans un modèle d'API simple qui peut être intégrée dans une technologie donnée. Cela permet à la technologie d'être utilisée avec peu d'effort. Peu d'efforts d'ingénierie sont nécessaires au-delà de la signature et de la configuration de l'application dans le système du fournisseur de services. Consultez votre fournisseur de services pour déterminer la meilleure façon d'implémenter leur technologie de sécurité pour votre environnement.

S'il vous plaît envisager d'utiliser le matériel des organisations suivantes pour aider à cette recommandation:

- Amazon Best Practices pour la résilience DDoS :
  - https://d0.awsstatic.com/whitepapers/DDoS_White_Paper_June2015.pdf
- Meilleures pratiques d'atténuation DDoS d'Arbor Networks :
  - https://www.arbornetworks.com/images/documents/Arbor%20Insights/AI_DDoSMitigation_EN2013.pdf
- Guide de défense Cisco DDoS :
  - http://www.cisco.com/web/about/security/intelligence/guide_ddos_defense.html

5.4.1 Risque

Une infrastructure sécurisée pour les services et les applications destinés au public est impérative en raison de la nature volatile d'Internet. Les attaques DDoS aléatoires se produisent souvent, et pour n'importe quelle raison. Des services DDoS peuvent être achetés dans le «marché clandestin» pour plusieurs centaines de dollars. Ainsi, les adversaires de l'entreprise, ou le client d'une entreprise, ne seront pas les seuls auteurs de telles attaques. Des attaques aléatoires peuvent se produire juste pour voir s'il est possible d'arrêter un système. Il est préférable de se préparer à de telles attaques afin de s'assurer que les services IoT critiques ne soient pas supprimés de manière inattendue. La disponibilité est un élément essentiel d'un produit ou service IoT.

5.5 Définir un modèle de stockage persistant

Les environnements d'application dans l'informatique moderne sont souvent éphémères, tels que les systèmes basés sur des conteneurs ou les environnements Cloud. En conséquence, le stockage alloué à ces systèmes n'est pas assez important (quantité de mémoire assignée), ni conçu pour être rendu disponible à long terme, pour que l'application utilise ces technologies en tant que stockage persistant. De plus, ces systèmes peuvent être définis comme des entités à la demande et ne semblent pas être normalement centralisés. En d'autres termes, les autres systèmes n'ont aucun moyen de définir quel système a suffisamment de mémoire pour une utilisation persistante.

C'est pourquoi les systèmes de stockage centraux sont impératifs, et pourquoi ils doivent être soigneusement sécurisés. Étant donné que les systèmes de stockage doivent être accessibles à tout système temporaire donné dans ce type d'environnement, tout serveur ou service éphémère compromis peut accéder à une entité de stockage persistante (ou couche) utilisée par de nombreux autres serveurs ou services. C'est souvent un moyen efficace pour les adversaires de compromettre latéralement (ou potentiellement, verticalement) un réseau donné.

Pour limiter cela, chaque serveur ou service doit avoir accès à un stockage persistant, mais doit stocker des informations basées sur l'application qu'il représente et, plus important encore, sur l'unique dispositif périphérique, partenaire ou utilisateur pour lequel l'application agit. La dernière partie de ce point est le point le plus essentiel, car l'application d'un accès persistant au stockage pour le compte d'une identité donnée limite l'accès éphémère du serveur ou du service aux données.

En d'autres termes, un adversaire qui a compromis un système éphémère ne peut affecter que les données stockées au nom de l'identité liée à ce même système éphémère. Si ce système n'a accès qu'aux données d'une seule identité, l'adversaire ne peut pas utiliser le compromis de ce système pour migrer latéralement vers d'autres comptes. Ils sont limités à l'accès à l'information pour cette identité unique. Cela limite considérablement la capacité de l'adversaire à exploiter une vulnérabilité pour compromettre le système d'une manière plus importante impliquant plus de ressources.

5.5.1 Risque

Si un modèle de stockage persistant sécurisé n'est pas défini, aucune architecture n'impose des attributs uniques par utilisateur à séparer en toute sécurité des autres actifs. Le résultat peut être que toute compromission d'un jeton qui accorde un accès adversaire à un périphérique de stockage peut entraîner la compromission des données de plusieurs utilisateurs. Toutefois, un modèle de stockage persistant peut isoler le compromis pour un seul utilisateur ou une technologie de stockage unique avec des données chiffrées. Dans les deux cas, la portée du compromis est considérablement réduite, accordant à l'organisation plus de temps pour réagir et combattre la menace à la fois pour les utilisateurs et pour l'entreprise.

5.6 Définir un modèle d'administration

Chaque système doit être accessible par l'administration pour dépanner et diagnostiquer les défauts d'application. Cela peut être difficile dans les environnements où les services ou les serveurs sont de courte durée, si un modèle administratif n'est pas suffisamment bien conçu.

Pour ce faire, identifiez comment l'équipe administrative communiquera avec chaque système de chaque couche. Il devrait y avoir des limites d'authentification, telles que les VPN, qui séparent les systèmes différents les uns des autres. Assurez-vous que l'équipe administrative doit s'authentifier auprès de chaque couche.

Identifiez également comment l'administrateur interagira avec le système. Est-ce que le système peut être copié avec des images instantanées, semblable à une MV ? Est-ce qu'un terminal est utilisé ? Un Secure Shell (SSH) distant est-il utilisé pour interagir avec le système ? Existe-t-il des APIs pour la surveillance et l'analyse des métriques du système, telles que l'utilisation du processeur, l'utilisation du disque et l'utilisation du réseau ? Peut-on les utiliser pour dépanner ou identifier des anomalies ?

Quel que soit le modèle, il y a certaines choses à définir :

- Comment les administrateurs s'authentifieront auprès de l'environnement
- Comment les administrateurs quand ils s'authentifient peuvent être attribués à des identités physiques :
  - Utilisation de l'authentification à deux facteurs (2FA)
- Comment les images instantanées de systèmes peuvent être faites
- Comment les changements peuvent être faits et doivent être suivis

5.6.1 Risque

Les environnements sans un architecture d'accès bien conçu pour la gestion administrative des systèmes finissent généralement par utiliser des moyens ad hoc pour y accéder en production. Cela conduit souvent à des ports administratifs ouverts à la connectivité publique ou à des services offrant des diagnostics, mais qui ne sont pas limités à l'utilisation par des tiers. Un modèle administratif clair réduit les possibilités que les attaquants puissent finir pour avoir un accès privilégié aux ressources critiques de l'IoT.

5.7 Définir une approche de journalisation et de surveillance des systèmes

Chaque système doit être surveillé pour permettre aux administrateurs et aux technologies de l'information (TI) de détecter et de diagnostiquer les anomalies. La surveillance doit être effectuée à plusieurs dimensions. Par exemple, la surveillance réseau dans la couche de l'infrastructure permet de diagnostiquer les attaques d'applications ou les attaques DDoS contre des composants réseau. La surveillance par couches identifie si des applications spécifiques ou des éléments d'infrastructure ont été violés. La surveillance au niveau du système définit si des applications individuelles ou des plates-formes d'applications sont attaquées ou compromises.

Cela nécessite évidemment plusieurs niveaux de surveillance et consolide l'information dans une ressource qui peut être transmise à une équipe de surveillance. Il existe plusieurs applications professionnelles qui fournissent cette technologie et convertissent les métriques en systèmes visuels utilisables par les professionnels de l'informatique et les ingénieurs système.

Les anomalies qui indiquent un comportement contradictoire peuvent inclure, mais ne sont pas limitées à :

- Augmentation du trafic réseau
- Augmentation du trafic réseau dans une direction étrange (en particulier sortie)
- Générer un trafic réseau d'une ressource qui ne devrait pas avoir besoin d'être transmis
- Utilisation anormale du processeur
- Utilisation du GPU pour les systèmes sans interface visuelle, mais avec un GPU intégré à la CPU
- Utilisation du disque ou du stockage réseau
- Modifications anormales de l'heure système sur un hôte particulier

Alors que les systèmes de surveillance pour la détection des anomalies sont facilement disponibles, le contexte peut être spécifique à l'application ou à l'infrastructure utilisée par l'organisation. Consultez l'entreprise fournissant le système de surveillance pour déterminer comment capturer et interpréter les métriques de la manière la plus efficace pour la mise en œuvre spécifique.

Les couches séparées peuvent avoir des disparités dans certaines anomalies indiquant une attaque ou un compromis. Évaluer quels sont ces indicateurs pour chaque couche.

S'il vous plaît envisager d'utiliser la documentation des organisations suivantes pour aider à cette recommandation :

- Documentation de surveillance Amazon EC2
  - http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/monitoring_ec2.html
- Motorisation Google Cloud
  - https://cloud.google.com/monitoring/
- Motorisation Microsoft Azure
  - https://azure.microsoft.com/en-us/documentation/articles/best-practices-monitoring/
- Didacticiels de surveillance DigitalOcean (Général)
  - https://www.digitalocean.com/community/tags/monitoring?type=tutorials

5.7.1 Risque

La technologie de surveillance des systèmes est un attribut clé du modèle de sécurité IoT. Sans surveillance, il n'existe aucun moyen de déterminer si une vulnérabilité a été détectée dans les composants de services critiques. La surveillance permet aux administrateurs de diagnostiquer rapidement les points critiques dans le service et l'infrastructure, et peut aider à différencier les incidents de sécurité des bogues logiciels.

5.8 Définir un modèle de réponse aux incidents

Il ne suffit pas de détecter un compromis potentiel ou une attaque en cours. L'entreprise doit être capable de réagir et de combattre l'attaque. Si un système est compromis, le nettoyage ou l'arrêt de ce système ne suffit pas. L'entreprise doit, au contraire, être en mesure de diagnostiquer la source du compromis, de patcher le système et de déployer le correctif sur toute l'infrastructure existante.

Cela peut être difficile si un environnement basé sur conteneur est utilisé lorsque les applications clonées s'exécutent avec une configuration vulnérable. Le système d'application doit être capable de détecter un événement de «redémarrage» ou de «mise à jour», lorsque la connexion de l'application est transférée d'une manière frappante à un autre système du Cloud ou que l'utilisateur est déconnecté de force pour permettre la mise à jour.

Cependant, quel que soit le modèle d'exécution, l'équipe d'ingénierie doit être capable de capturer des métriques de manière à permettre une analyse légale. Ces politiques et procédures doivent être figées et approuvées par un juriste (et éventuellement l'équipe d'assurance) pour valider si l'information est contenue d'une manière qui convient aux agents d'application de la loi (LEO). La conformité aidera à faire en sorte que l'entreprise se conforme non seulement à la loi locale et fédérale, mais fournisse des exemples de compromis qui peuvent être utilisés devant les tribunaux.

Une fois les échantillons capturés, chaque aspect du système global doit être évalué pour les journaux, les métriques et autres données pouvant corroborer l'événement en question. Ces données devraient toutes être capturées et stockées dans un système sécurisé pour un examen juridique.

S'il vous plaît envisager d'utiliser la documentation des organisations suivantes pour aider à cette recommandation :

- Recommandations du CERT pour la création d'un CSIRT
  - http://www.cert.org/incident-management/products-services/creating-a-csirt.cfm

5.8.1 Risque

Les organisations qui n'ont pas de modèle de réponse aux incidents prendront beaucoup plus de temps pour organiser leurs ressources, identifier les systèmes compromis, mettre en quarantaine ces systèmes et examiner les systèmes pour obtenir des informations. Cela ralentit également considérablement les efforts qui devraient être faits pour patcher et restaurer un système donné. Cette impréparation donne aux adversaires une grande fenêtre d'opportunité pour tirer parti d'un compromis dans un mouvement latéral ou vertical dans un environnement donné. Cela peut entraîner un compromis beaucoup plus important en raison de l'augmentation du temps de réponse. Les organisations devraient être prêtes à réagir à un incident presque immédiatement afin de réduire le temps dont dispose un adversaire pour contrôler des parties critiques du service.

5.9 Définir un modèle de récupération

Indépendamment du fait qu'un utilisateur ou une application soit affecté en raison d'une compromission de sécurité ou d'un problème du hardware, une récupération doit avoir lieu. Une procédure devrait être mise en place pour la récupération des informations et des capacités au niveau de l'application. La procédure doit être adaptée au contexte de chaque application et couche du système.

Par exemple, si une application a collecté des informations à partir d'un dispositif périphérique concernant le résultat d'une action particulière et qu'une erreur de stockage empêche l'application de consolider la sortie de ces données dans un stockage persistant, l'application peut :

- Essayez à nouveau de stocker jusqu'à ce qu'elle réussisse (peut être infini)
- Tentative de stockage pour un nombre limité de tentatives jusqu'à ce que les seuils de réussite ou d'échec soient atteints
- Échouer immédiatement, en perdant peut-être les métriques
- Demander à nouveau au dispositif périphérique les mêmes données (peut ne jamais être disponible)

La méthode la plus adaptée à l'application et aux besoins de l'entreprise doit être choisie. Ceci, encore une fois, dépendra du contexte de l'application et peut ne pas être facile à modéliser en dehors d'un système donné.

Engager à la fois l'ingénierie et le leadership commercial afin de déterminer comment une application défaillante ou compromise devrait être récupérée, en particulier dans le contexte de l'activité de l'utilisateur.

Pour les systèmes qui ont été prouvés compromis par un adversaire, il doit y avoir un modèle en place pour valider que l'application ou le système a été suffisamment corrigé avant la récupération. Sans la définition de cet ensemble de politiques et de procédures, un système vulnérable peut simplement être redéployé dans l'écosystème de services, ce qui facilite d'autres compromis.

5.9.1 Risque

Les modèles de récupération garantissent que les informations, les applications et les configurations sont correctement restaurées. Sans un modèle de restauration, l'équipe peut involontairement redéployer des sous-systèmes vulnérables vers des serveurs ou une infrastructure. En outre, des données corrompues qui auraient pu être manipulées par un adversaire dans une base de données ou un environnement de stockage pouvaient être répliquées sur plusieurs systèmes, propageant involontairement des logiciels malveillants ou simplement des données modifiées. Les processus de récupération réduisent la capacité des adversaires à abuser des faiblesses dans la récupération d'un incident, ce qui s'ajoute à un événement déjà coûteux.

5.10 Définir un modèle de caducité

Chaque système déployé par une organisation et chaque couche utilisé a une durée de vie. Même si le même produit ou service est déployé par l'organisation depuis des décennies, les technologies utilisées pour piloter ce produit ou service vont changer. Ainsi, il ne doit pas seulement y avoir un plan pour la conception et la mise en œuvre du produit ou du service, il doit y avoir un plan pour retirer du marché ce produit ou service.

Ce processus permet de garantir que toutes les technologies sont révoquées et désaffectées de telle sorte qu'un adversaire ne peut pas prendre l'identité de la technologie donnée ou en utiliser les installations. Par exemple, un cas simple est un domaine d'Internet pour un produit particulier après l'acquisition d'une entreprise par une société mère. Si le produit est renommé et que le domaine est migré vers le domaine de la société mère, un adversaire peut être en mesure de s'approprier le domaine désormais disparu. Si l'adversaire peut émettre des certificats cryptographiques pour ce domaine et continuer à interagir avec la technologie déployée sous cet ancien domaine, il y aura un important manque de sécurité causé par un manque de procédure dans la caducité de ce produit ou service.

Chaque technologie utilisée dans l'architecture, la mise en œuvre et la gestion d'un produit ou service donné doit ensuite être cataloguée et évaluée pour sa facilité d'utilisation. Une fois que cette technologie n'est plus utilisable, elle peut être retirée du marché selon son modèle. Cela permet aux ingénieurs et aux chefs d'entreprise de migrer la technologie vers un ensemble plus approprié d'innovations sans lacunes dans les plates-formes sous-jacentes. Cela garantit également qu'un produit qui ne sera plus offert aux partenaires et aux utilisateurs finira sa vie sans risque d'exploitation par des adversaires après la fermeture de l'entreprise.

5.10.1 Risque

L'absence d'un processus de caducité peut entraîner la compromission des dispositifs périphériques et des services par des concurrents ou des adversaires. Cela est possible légalement parce que si une organisation libère l'accès à certains objets, tels que les noms de domaine, numéros de téléphone et autres services renouvelables, un adversaire ou un concurrent a le droit d'acquérir ces objets, même si cela semble contraire à l'éthique. Cela peut soumettre des appareils ou des services à des abus peu scrupuleux ou même un comportement malveillant.

5.11 Définir un ensemble de classifications de sécurité

Pour gérer efficacement les interactions avec les organisations partenaires, les classifications de sécurité doivent être définies. Cela donnera le ton pour non seulement la politique organisationnelle interne sur la sécurité des données, mais aidera à définir le niveau de sécurité que les organisations partenaires doivent appliquer aux données de l'entreprise et service principal, à leurs propres données et aux données des clients.

Bien que ce processus doive être étudié et adapté à l'organisation, la plupart des stratégies de classification de sécurité des données doivent commencer par les classes suivantes :

- Public - Toute entité ayant accès
- Classé - L'utilisateur doit autoriser la diffusion
- Secret - Données spécifiques à l'utilisateur
- « Top Secret » - Données spécifiques à l'organisation, à ne jamais publier

Après avoir défini les classes de base, l'organisation doit évaluer comment chaque classe de sécurité doit être attribuée à une classe de données. En d'autres termes, évaluez comment la classification devrait être utilisée dans la pratique, pas seulement en théorie. Déterminer quelles politiques et procédures devraient être mises en place du point de vue des entreprises et de l'ingénierie.

Cela permettra à l'organisation de non seulement élaborer une politique technique, mais aussi d'adopter une politique d'entreprise qui prend en charge les exigences techniques. Cela permet à l'équipe d'ingénieurs de transmettre plus facilement ces exigences à des partenaires et à des organisations internes qui chercheraient à enfreindre la politique, intentionnellement ou non.

Une fois les classifications de sécurité normalisées, il est important d'évaluer comment le modèle de classification de sécurité peut être affecté par les exigences de confidentialité de l'entreprise et de ses utilisateurs. L'organisation doit prendre le temps d'appliquer un modèle de confidentialité aux classifications de sécurité, de donner du sens aux données des utilisateurs et de protéger leur vie privée dans le cas où un partenaire souhaite accéder à des ressources spécifiques susceptibles d'exposer les utilisateurs. En contextualisant la confidentialité dans le contexte des classifications de sécurité, les partenaires devront obtenir l'approbation du chef d'entreprise et des utilisateurs, lorsque les partenaires souhaitent acquérir certains types de données axées sur la confidentialité. Les utilisateurs doivent avoir la possibilité de protéger leurs données confidentielles et doivent pouvoir limiter l'exposition de leurs données à des tiers.

5.11.1 Risque

Il est impératif de classer les modèles pour la sécurité afin de concevoir des solutions qui utilisent la sécurité de manière efficace. Afin de protéger l'information, cette information doit être quantifiée afin que les contrôles appropriés puissent être formulés en fonction des politiques et procédures correspondantes. Sans ces modèles, les ingénieurs ont tendance à mettre en œuvre la sécurité soit trop intensément, soit pas du tout, en fonction de leur perception des risques encourus. Toute l'équipe, y compris les ingénieurs et les chefs d'entreprise, doit identifier ce que les données signifient pour l'entreprise et comment elles doivent être sécurisées dans le cadre d'une gamme appropriée de contrôles rentables.

5.12 Définir des classifications pour les ensembles de types de données

Après avoir défini les classifications de sécurité, l'organisation doit définir les types de données à utiliser par le produit ou le service IoT global. Cela permettra à l'organisation de définir clairement quels types d'informations sont acquis, générés et diffusés aux « acteurs » dans le système IoT, et comment l'organisation doit traiter ces types de données. Ces données fourniront un contexte et une valeur aux composants globaux utilisés dans l'environnement IoT.

Bien que ce document ne tente pas de modéliser toutes les variations de données pouvant être pertinentes pour une organisation spécifique, certains types peuvent être les suivants :

- Utilisateurs
- Opérations
- Images
- Documents modifiables
- Informations personnelles identifiables
- Information sur la santé protégée

Une information peut être attribuée à un ou plusieurs types. Mais, les données elles-mêmes ne devraient être attribuées qu'à une classe de sécurité. Alors que le type identifie ce que les données représentent et comment elles doivent être traitées, la classe de sécurité représentera comment, où et quand les informations peuvent être utilisées, et à qui elles peuvent être partagées.

Définir les différents types de données et leur attribuer des classifications est un processus long. Cela établit une norme organisationnelle pour l'entreprise et permet à l'équipe d'ingénierie d'exécuter des contrôles techniques sur les données et leurs classifications. Cela aide grandement les équipes d'ingénierie et de direction des affaires plus tard lors de la négociation avec les partenaires sur la façon dont les données peuvent être partagées et traitées.

5.12.1 Risque

Comme pour les classifications de sécurité, les contrôles ne peuvent pas être implémentés autour de données sans quantifier ce que ces données sont et quelle relation ces données ont avec l'entreprise. Ces classes définissent comment les informations doivent être utilisées dans le système et quelles protections doivent être appliquées aux données afin de maintenir une posture de sécurité appropriée. Sans ces classes, les ingénieurs ont tendance à appliquer des mesures de sécurité trop strictes ou trop faibles. Les mesures de sécurité devraient être approuvées par l'équipe d'ingénierie et les chefs d'entreprise, afin d'équilibrer les contrôles avec l'importance des données pour l'entreprise.

6 Recommandations de haute priorité

Les recommandations de haute priorité représentent l'ensemble des recommandations qui doivent être mises en œuvre, mais seulement si l'architecture de dispositifs périphériques l'exige. Par exemple, toutes les architectures de dispositifs périphériques ne nécessitent pas un boîtier de produit résistant aux manipulations. Ces recommandations devraient être évaluées pour déterminer si l'analyse de rentabilité les considère comme une exigence.

6.1 Définir un modèle d'autorisation claire

Alors que le modèle de confidentialité traite de la manière dont les informations de l'utilisateur sont proposées aux partenaires, le modèle d'autorisation définit la manière dont l'entreprise ou les partenaires agiront au nom d'un utilisateur. Ceci, par exemple, serait utile pour un système domotique où les paramètres d'un partenaire pourraient optimiser l'utilisation du chauffage ou du refroidissement dans une maison donnée. Le modèle d'autorisation accorderait au partenaire la possibilité de modifier les commandes de chauffage ou de climatisation pour le domicile de cet utilisateur lorsque certaines mesures ont été détectées par le partenaire.

Pour ce faire, ayez une interface graphique similaire qui décrit les capacités d'autorisation granulaires et comment elles seront distribuées aux partenaires. Permettre à l'utilisateur d'accorder l'accès ou de révoquer l'accès à certaines fonctionnalités à la demande. Assurez-vous que les capacités révoquées agissent immédiatement, afin de réduire le risque d'abus.

Le système doit faire l'objet d'une surveillance étroite pour s'assurer que les partenaires ne prennent pas les mesures qu'ils ne sont pas autorisés à prendre. Le contrôle granulaire du modèle d'autorisation doit permettre aux utilisateurs de configurer lorsque les partenaires ont accès à certaines fonctionnalités et à quelle fréquence. Des attributs comme celui-ci amélioreront si un utilisateur peut prendre le contrôle de son système à partir d'un partenaire potentiellement abusif ou compromis (piraté).

6.1.1 Risque

Sans modèle d'autorisation, les tiers n'auront pas un accès restreint aux capacités d'un utilisateur. Cela peut permettre à un tiers malveillant ou compromis d'acquérir un accès complet à la technologie ou aux données d'un utilisateur. En créant un modèle d'autorisation, l'accès est limité aux seuls attributs qu'un utilisateur autorisera. Cela permet à l'utilisateur d'avoir un meilleur contrôle sur les capacités et les données mises à la disposition des tiers, et réduit le risque du fournisseur de services IoT en atténuant le potentiel de compromis généralisé.

6.2 Gérer l'architecture cryptographique

Toute la technologie déployée dans un environnement IoT doit utiliser la cryptographie, que la technologie soit un dispositif périphérique léger à faible consommation ou un service Cloud robuste. Pour mettre correctement en œuvre la sécurité dans un produit ou service IoT, la cryptographie utilisée doit être bien définie à niveau de l'architecture, gérée et ajustée pour répondre aux spécifications changeantes au fil du temps.

L'équipe d'ingénierie doit identifier si :

- Leurs algorithmes cryptographiques ont été dépréciés
- Ils utilisent des clés cryptographiques avec des longueurs de bits adéquates
- Les algorithmes de hachage sont sujets à des attaques de collision
- Un générateur de nombres aléatoires puissant est utilisé
- Les messages sont complétés suffisamment avec des données aléatoires
- Les protocoles cryptographiques, tels que TLS, sont à jour avec les meilleures pratiques
- Des concepts centrés sur la confidentialité, tels que la sécurité itérative, sont utilisés

- Les mots de passe plain-texte ou les codes PIN sont transmis sur le réseau
- Un algorithme cryptographique personnalisé a été utilisé

Chacun de ces points, et bien d'autres, sont importants pour maintenir une architecture cryptographique de haute qualité dans le produit ou le service IoT. La réussite du déploiement d'une solution cryptographique est étroitement liée à la capacité de l'équipe d'ingénierie à tirer parti des solutions de cryptographie les plus résilientes pour déployer des correctifs sur des technologies utilisant des solutions moins résilientes.

Par exemple, l'algorithme RC4 a récemment été découvert pour avoir des failles de sécurité importantes. Si un correctif peut être distribué de manière sécurisée aux clients configurés pour utiliser RC4, cela remplace RC4 par AES-256, alors RC4 devient moins préoccupant. Si l'authentification mutuelle est effectuée en utilisant une technologie plus résiliente, telle que l'échange de clés « Ephemeral Diffie Hellman » et des clés asymétriques, ou un jeton de sécurité avec un UICC, le correctif peut être vérifié sans utiliser l'algorithme cryptographique vulnérable.

Les mots de passe et les Pins utilisés par un utilisateur ou un dispositif périphérique ne doivent jamais être transmis en plein texte sur le réseau, même si le canal de communication est sécurisé par chiffrement. Au lieu de cela, le hachage cryptographique du mot de passe ou du Pin doit être utilisé, pour s'assurer que toute mauvaise configuration dans le tunnel cryptographique n'expose pas le mot de passe lui-même. Le hachage doit être généré par le mot de passe et au moins un jeton ponctuel unique. Bien qu'il soit courant que ce jeton soit extrait de la session réseau, il est plus sûr de prendre la valeur d'un « code roulant » stocké à la fois sur le dispositif périphérique et sur l'infrastructure de service. De cette façon, un attaquant avec une position de privilège sur le réseau ne peut pas semer le hachage avec des valeurs bénéfiques, ce qui peut entraîner une attaque par signature forcée.

Les algorithmes cryptographiques personnalisés (algorithmes conçus en interne) ne devraient jamais être utilisés. Toujours utiliser les algorithmes recommandés développés par les cryptographes et recommandés par des organismes de surveillance spécialisés dans la sécurité cryptographique. Évitez toujours l'utilisation d'algorithmes mal conçus, d'algorithmes obsolètes, ou de compression, binaire-texte, ou d'autres algorithmes généralement pris pour des algorithmes cryptographiques, tels que LZO, base64, ROT13 et XOR.

Veuillez consulter les guides et références suivants pour plus d'informations sur ce sujet :

- ISO 18033-1: 2015 - Algorithmes de chiffrement
- ISO 18033-2: 2015 - Chiffrages asymétriques
- ISO 18033-3: 2015 - Chiffrement de bloc
- www.owasp.org/index.php/Guide_to_Cryptography
- csrc.nist.gov/publications/nistpubs/800-57/sp800-57_part1_rev3_general.pdf
- csrc.nist.gov/groups/ST/toolkit/key_management.html

6.2.1 Risque

Le déploiement correct d'une solution avec une architecture cryptographique garantit que les algorithmes, protocoles et secrets utilisés répondent aux recommandations actuelles. De plus, les recommandations changent avec le temps. Sans une architecture cryptographique,

il sera plus difficile d'identifier toutes les technologies obsolètes, ce qui crée une opportunité pour des failles de sécurité.

6.3 Définir un modèle de communication

Chaque système de l'écosystème de services doit être capable d'une authentification mutuelle. Aucune plate-forme informatique au sein de cet écosystème ne devrait être accessible aux utilisateurs publics anonymes. Chaque dispositif périphérique, partenaire ou utilisateur communiquera avec l'écosystème de services au moyen de technologies nécessitant une authentification mutuelle. Étant donné que les services qui composent l'interface utilisateur sont généralement déployés et gérés dans un environnement différent, l'interface accessible au public doit être confinée à cet espace. L'écosystème de services, cependant, comprend l'ensemble de tous les systèmes utilisés pour déployer le service sur toutes les ressources authentifiées.

Cela inclut les dispositifs périphériques qui n'ont pas encore été provisionnés par le système, car le processus de personnalisation et de fabrication du matériel doit configurer le matériel de manière suffisamment précise pour qu'il puisse être authentifié en tant que ressource déployée par l'entreprise.

Par conséquent, le modèle de communication doit fournir :

- Authentification mutuelle
- Confidentialité
- Intégrité

Pour y parvenir efficacement, le modèle de communication doit également fournir :

- Une racine de confiance centralisée ou, alternativement, une racine de confiance décentralisée
- Provisionnement et révocation d'identité
- Confidentialité persistante itérative (“Perfect Forward Secrecy”)

Une racine de confiance doit être utilisée pour garantir que chaque entité dans le modèle de communication est autorisée par la même organisation. Cela permet de s'assurer que toutes les entités ont été provisionnées et autorisées par une organisation centrale. La technologie utilisée pour garantir cette racine de confiance peut être centralisée (similaire aux certificats TLS) ou décentralisée (similaire aux modèles IoT basés sur la chaîne de blocs Bitcoin, par exemple le projet ADEPT d'IBM / Samsung, Tilepay et autres). Quoi qu'il en soit, une entreprise centrale doit être le propriétaire du modèle et garder le système d'approvisionnement.

L'approvisionnement et la révocation doivent faire partie du modèle de communication, afin de garantir que tout secret ou identité compromis peut être retiré du système avec un minimum d'effort. Des technologies telles que le protocole OCSP (« Online Certificate Security Protocol ») aident à ce processus.

Le protocole de communication doit utiliser une technologie qui atténue le risque de compromettre les communications du passé. Ceci est fait en créant des clés cryptographiques asymétriques éphémères qui sont utilisées pour échanger un secret de communication. Si un certificat est compromis, le secret éphémère ne sera pas. Cela

garantit que le stockage des messages cryptés pendant une longue période ne conduira pas plus tard un adversaire à les décrypter si le secret privé du certificat est compromis ou exposé.

Le défi de la sécurité des communications réside dans la mise en œuvre et la longévité de la technologie. Des algorithmes de cryptage peuvent être sélectionnés et conservés avec un degré de confiance élevé par des entités qui fonctionnent comme autorité de certification, ce qui réduit le potentiel d'échec.

Les bibliothèques et les implémentations d'algorithmes conçues ou approuvées par les autorités techniques doivent être utilisés. Les implémentations personnalisées d'algorithmes ne doivent pas être utilisées. Cela diminue non seulement le travail de l'équipe d'ingénierie, mais aussi le potentiel d'un algorithme à être cryptographiquement affaibli par un système mal conçu ou incorrectement mis en œuvre.

S'il vous plaît envisager d'utiliser le matériel des organisations suivantes pour aider à cette recommandation :

- Guide d'utilisation de l'authentification mutuelle CafeSoft Apache :
  - http://www.cafesoft.com/products/cams/ps/docs32/admin/ConfiguringApache2ForSSLTLSMutualAuthentication.html

6.3.1 Risque

La sécurité des communications est la pierre angulaire de l'IoT. Sans la sécurité des communications, il n'y a aucune garantie que les périphériques intégrés communiquent avec les services back-end appropriés. Ceci est impératif pour les services critiques qui guident, configurent et envoient des commandes à des périphériques tels que des systèmes télématiques, des dispositifs médicaux et des systèmes de contrôle industriels. Sans la sécurité des communications, il n'y a aucune garantie que les commandes sont envoyées au bon dispositif périphérique. Appliquez la sécurité des communications pour vous assurer que les messages sont envoyés et reçus par l'homologue prévu.

6.4 Utiliser les services d'authentification réseau

Les opérateurs de réseau, lorsqu'ils sont utilisés en tant que partenaires, permettent aux utilisateurs d'être authentifiés à l'aide de jetons spécifiques à l'opérateur de réseau. Alors que ces jetons, présents dans l'UICC de l'opérateur réseau, authentifient un utilisateur sur la couche réseau, ils n'authentifient pas nécessairement l'utilisateur au niveau de la couche d'application. L'utilisation des technologies suivantes peut faciliter l'authentification du réseau :

- Architecture Bootstrap générique (3GPP TS 33.220)
- M2M SM (ETSI TS 102 921)

Évaluer si la technologie d'authentification créera une signification au niveau de la couche d'application, à des fins d'authentification. Si le jeton peut être utilisé comme magasin de sécurité, déterminez si le périphérique peut être utilisé comme couche d'authentification pour que le dispositif périphérique physique crée une TCB à l'aide du jeton.

Alors que de nombreux opérateurs de réseau imposent l'authentification basée sur le réseau, l'accès à cette API pour authentifier les utilisateurs ou les points de terminaison est une technologie relativement nouvelle. Évaluez si l'opérateur réseau avec lequel vous travaillez crée une expérience significative dans cet espace. Si tel est le cas, envisagez d'utiliser cette technologie plutôt qu'un jeton d'authentification de couche réseau, car il peut être plus facile d'utiliser une technologie de magasin de sécurité au lieu de plusieurs technologies.

6.4.1 Risque

Lorsque les services d'authentification réseau intègrent des ancres de confiance tels que l'UICC, ne pas utiliser ces services pour sécuriser la couche d'application limitera la capacité de l'application à authentifier les utilisateurs de manière fiable et augmentera les coûts de la plate-forme du dispositif périphérique sous-jacente. Cela augmente le coût de déploiement et diminue également les informations disponibles pour l'organisation de l'opérateur de réseau.

6.5 Fournir des serveurs dans la mesure du possible

L'approvisionnement du serveur implique la définition, la configuration, la personnalisation et le déploiement d'un serveur dans un environnement de production. Le processus d'approvisionnement, du point de vue du service, garantit qu'un serveur est sécurisé et prêt à être déployé dans un environnement potentiellement hostile.

Que le serveur soit déployé dans une infrastructure Cloud, un fournisseur d'hébergement dédié ou dans l'espace de rack personnel d'une entreprise, un serveur sera vulnérable aux menaces internes et externes. Le serveur doit ensuite être protégé contre n'importe quelle attaque avant d'être déployé dans l'infrastructure de service.

Pour ce faire, identifiez les services qui devraient être accessibles à l'environnement. Définissez si l'environnement dans lequel le serveur va fonctionner sera public ou privé, et ce que cela signifie dans le contexte de la sécurité du serveur. Déterminez si chaque service exécuté sur le serveur doit être accessible au public ou si seuls les clients authentifiés doivent se connecter au service.

Évaluez le cycle de vie du système d'exploitation qui s'exécutera sur le serveur. Déterminez comment gérer correctement les mises à jour des logiciels pour vous assurer que les correctifs de sécurité seront rapidement déployés et affectés aux serveurs travaillant en production. Évaluer un modèle de restauration dans le cas où les mises à jour échoueraient ou causeraient des problèmes inattendus avec les services de production, car certaines mises à jour de bibliothèques ou d'applications pourraient entraîner des effets secondaires inattendus.

Enfin, évaluez le modèle de caducité du serveur provisionné afin de déterminer le moyen le plus sûr de supprimer des actifs du système. Cela inclut les journaux système qui peuvent être nécessaires pour évaluer le comportement anormal du service ou du client.

Cette recommandation implique qu'un processus de gestion des correctifs doit être implémenté par l'organisation pour identifier les services vulnérables, déployer des correctifs et contrôler le succès de la mise en œuvre de ces correctifs.

Veuillez consulter le matériel suivant sur la gestion des correctifs :

- http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r3.pdf

6.5.1 Risque

Le provisionnement de serveur est un élément impératif de la sécurité globale d'un environnement IoT. Sans cela, le contrôle de l'organisation sur l'architecture du serveur sera considérablement affaibli. Cela peut entraîner des lacunes de sécurité en raison d'un manque de spécification de l'architecture. Sans spécification, l'organisation ne peut pas vérifier si les technologies déployées sont conformes aux meilleures pratiques modernes. En outre, l'amélioration de ces technologies nécessitera une étude de chaque système déployé pour évaluer les deltas entre les actifs déployés. C'est inefficace et une grande préoccupation dans le cas où une mise à jour de sécurité critique doit être déployée. S'il n'y a pas de cohérence et pas d'architecture pour définir les services, il n'y aura aucun moyen de suivre facilement les systèmes nécessitant une attention immédiate sans vérifier manuellement chacun d'eux.

6.6 Définir un modèle de mise à jour

La mise à jour d'un environnement d'exécution, d'une image d'application ou de TCB est un processus difficile.

Considérez le modèle d'exemple suivant qui simplifie le processus global :

- Pour chaque couche de la plate-forme d'exécution, définissez une ressource réseau telle qu'une URL unique pour la nouvelle image de l'application.
- Générer une clé de signature pour chaque couche spécifique
- Pour toutes les nouvelles versions autorisées de chaque couche, générez une image de cette couche
- Inclure des métadonnées décrivant l'image (version, horodatage, identité, etc.) dans l'image de la couche
- Signez l'image de la couche avec la clé de signature
- Rendre l'image, la signature et la clé publique disponibles, éventuellement via la ressource réseau unique ou via un service de mise à jour

Lorsqu'un nouveau système est déployé, il doit :

- Pour chaque couche:
  - Récupérer la ou les version (s) à déployer
  - Vérifier cryptographiquement l'image
  - Déployer la couche d'image sur le système

Aucun secret privé ne doit être stocké dans une couche d'application. Au lieu de cela, les secrets doivent être provisionnés dynamiquement au fur et à mesure que chaque système est déployé, pour personnaliser chaque système. Ces identités doivent être révoquées lorsque le système est mis hors service, quelle que soit la durée de vie de ce système.

Cette recommandation implique qu'un processus de gestion des correctifs doit être utilisé pour maintenir les services et les technologies au sein de l'infrastructure.

Veuillez consulter la documentation suivante pour plus d'informations :

- http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-40r3.pdf

6.6.1 Risque

Sans un modèle de mise à jour bien défini, les services et les applications risquent d'être compromis par l'utilisation abusive de la procédure de mise à jour. Les adversaires peuvent être en mesure d'injecter des applications personnalisées dans le processus de mise à jour et de déployer leur propre logiciel sur les systèmes Cloud et d'autres serveurs. Si l'infrastructure de sécurité des communications n'est pas sécurisée, cela peut facilement être effectué simplement en manipulant des services réseau tels que DNS (« Domain Name Service »). Des attaques plus avancées contre le routage, telles que les attaques BGP (« Border Gateway Protocol »), ont été mises en œuvre à plusieurs reprises dans le passé pour compromettre des services non sécurisés.

6.7 Définir une politique de violation pour les données exposées

Définir des politiques et des procédures pour la classification des données ne suffit pas. Il doit également y avoir un modèle pour détecter si les données ont été exposées par un partenaire. L'entreprise doit avoir un plan en place pour évaluer si un partenaire a été impliqué dans des pratiques commerciales qui enfreignent les contrôles technologiques ou les politiques mises en place pour protéger les données et la vie privée des utilisateurs.

Pour ce faire, l'équipe d'ingénierie doit définir les technologies de surveillance et de journalisation qui s'appliquent aux classifications de sécurité et pas simplement aux données utilisateur. Cela permettra à une piste d'audit de s'appliquer non seulement à l'information, mais à la classification de cette information. Cela aidera l'entreprise à se défendre en cas d'exposition d'informations utilisateur. L'entreprise sera en mesure de montrer que ses classifications de sécurité et les contrôles techniques mis en place pour gérer ces classes, géré, stocké et diffusé les données conformément à la politique.

Il est avantageux que l'organisation utilise la technologie de surveillance et d'enregistrement pour prouver lorsqu'un partenaire a enfreint les règles des classifications de sécurité. Les dirigeants devraient, à ce moment-là, décider si le Partenaire devrait être assujetti à des amendes, de licenciement ou de toute autre conséquence.

6.7.1 Risque

Sans une politique de violation, il y a peu de gardes juridiques pour protéger l'organisation de la responsabilité des données qui ont été exposées par un tiers. Si l'entreprise est la source des données exposées, le tiers peut avoir perdu les données, mais l'entreprise est responsable des données qu'elle transmet à ses partenaires.

Les politiques de violation garantissent que les partenaires doivent maintenir un niveau de sécurité adéquat pour les données qui leur sont fournies. Toute violation de cette sécurité permet de supprimer la responsabilité du fournisseur de services IoT tant que le fournisseur de services IoT respecte ses propres exigences de sécurité. Ensuite, il appartient au partenaire d'adhérer à la politique.

Ces politiques devraient être examinées par les équipes juridiques et d'assurance pour s'assurer que les modèles réduisent, en fait, la responsabilité de l'entreprise en adhérant à des politiques et procédures de sécurité rigoureuses. Certaines entreprises, en raison de la

nature des produits ou des services qu'elles offrent, peuvent ne pas être exemptées en raison de règlements, de lois ou d'autres problèmes.

6.8 Forcer l'authentification par l'intermédiaire de l'Écosystème de Services

Une interface utilisateur ne doit jamais authentifier directement un utilisateur. Le système doit toujours pouvoir authentifier l'utilisateur en utilisant le service disponible de manière centralisée. La seule exception à cette règle est si une application s'exécutant sur un périphérique mobile est protégée par un code d'accès local. Ce code peut être utilisé pour accéder à l'application locale. Toutefois, l'accès aux services et ressources distants doit être vérifié par un jeton d'authentification distinct.

Bien que, pour la facilité d'utilisation, l'équipe d'ingénierie peut choisir de réduire ces deux schémas d'authentification en un si l'utilisateur reçoit suffisamment d'informations qui lui décrivent les risques liés à l'utilisation de cette méthode d'authentification. Une telle méthode permettrait au mot de passe de l'application locale d'un utilisateur authentifié de déchiffrer une base de données locale contenant un jeton d'authentification qui fonctionne sur le service distant. Ce modèle d'authentification en plusieurs étapes peut être suffisant pour la plupart des utilisateurs.

Quoi qu'il en soit, le service d'authentification central doit d'abord authentifier l'utilisateur auprès de l'application locale, puis appliquer des stratégies et des procédures qui spécifient comment ce jeton d'authentification peut être utilisé et pour quelle durée. Les métriques doivent également être collectées pour déterminer si l'utilisateur a migré vers une plate-forme informatique alternative, mais utilise le même jeton. Ou, si l'utilisateur a migré vers un autre emplacement dans un court laps de temps, mais utilise le même jeton. Selon le type et la vitesse de déplacement, ces mesures peuvent indiquer une compromission potentielle du jeton. À ce stade, le jeton doit être invalidé et l'utilisateur doit être obligé de se reconnecter, éventuellement en utilisant l'authentification multi-facteur, le cas échéant.

6.8.1 Risque

En raison des abus possibles dans les systèmes de dispositifs périphériques, quelle que soit la sécurité de l'architecture, l'authentification d'un utilisateur sans confirmation à partir d'un système back-end est toujours peu fiable. Cela suppose que l'utilisateur n'a pas mis à jour ses informations d'identification ou peut fragmenter les informations d'identification sur plusieurs types de périphériques. Ceci est inefficace et peut ouvrir une brèche où un périphérique compromis utilise une ancienne version des informations d'identification de l'utilisateur.

6.9 Validation de l'entrée d'implémentation

Toutes les données acquises à partir d'un dispositif périphérique, d'un utilisateur ou d'un utilisateur présumé doivent être analysées à la recherche d'anomalies. La voie d'attaque la plus facile pour un adversaire est toujours d'abuser de l'entrée de l'application Web dans les services qui composent l'interface utilisateur. En effet, cette technologie doit fournir des informations de manière dynamique, en fonction des variations de localité, d'encodage et d'autres paramètres qui changent d'un utilisateur à l'autre. Les utilisateurs qualifiés peuvent manipuler certains attributs des codages pour provoquer des effets secondaires inattendus bénéfiques pour un adversaire à différentes couches des sous-systèmes de traitement.

Par exemple, une attaque fascinante inclut l'encodage d'un octet nul dans des messages traités comme des chaînes par des langages de plus haut niveau. Certains langages de haut niveau acceptent les octets nuls dans le cadre d'une chaîne binaire, plutôt que de le voir comme un délimiteur. Lorsque cette chaîne binaire est transmise aux bibliothèques de niveau inférieur, l'octet nul incorporé est interprété comme un délimiteur de chaîne, tronquant la chaîne pour signifier quelque chose de complètement différent de ce que l'application a interprété comme étant la chaîne. Dans le passé, cela a été une manière intelligente d'accéder aux ressources du système de fichiers qui seraient autrement indisponibles pour un utilisateur particulier.

Bien qu'il existe un nombre infini de variantes pour la saisie malveillante, les ingénieurs n'ont pas besoin de tester tous les cas possibles. Au lieu de cela, le processus est assez simple :

- Identifier comment les données doivent être utilisées en interne
- Appliquer une stratégie sur les types de codage et de caractères qui adhèrent au modèle d'utilisation interne
- Concevoir une API qui analyse les données conformément à cette politique
- Générer une exception lorsque des données ont été identifiées qui casse le modèle
- Connecter l'événement en interne avec des métadonnées concernant la session pour aider à détecter le comportement contradictoire

Toutes les données stockées dans le système doivent d'abord être traitées et composées dans un modèle statique. Une technique efficace consiste simplement à encoder toutes les données avec l'algorithme base64, puis à les placer dans une base de données. Cela garantit que les données ne peuvent en aucun cas manipuler la base de données.

6.9.1 Risque

Les systèmes qui n'utilisent pas la validation d'entrée sont soumis à un éventail d'attaques possibles, y compris les problèmes référencés dans le Top 10 d'OWASP tels que « SQL Injection » (SQLi), et même les attaques d'exécution de code à distance. Parce que la gamme des abus potentiels est si large, le risque ne peut pas être entièrement quantifié ici. La validation des entrées est un attribut essentiel de toute application sécurisée, qu'il s'agisse d'un service cloud ou d'une application exécutée sur un dispositif périphérique.

6.10 Mettre en œuvre le filtrage de sortie

Le filtrage des résultats est le complément de la validation des entrées. Ce processus sécurise non seulement la couche de présentation de la manipulation contradictoire, mais empêche également le système de déchiffrer des informations à un utilisateur qui devrait être considéré comme privilégié.

Dans le premier cas, toutes les données à restituer par la couche de présentation doivent être évaluées avant de quitter la couche de service. Cela garantira que les données codées dans la couche de présentation, par exemple dans les messages JSON ou le code JavaScript codé, ne contiennent pas de formatage susceptible de briser ou d'invalider la présentation des données. Cela signifie que tous les caractères stockés dans le système qui, s'ils sont rendus, risquent de casser le modèle de présentation doivent être filtrés ou encodés de manière à ne pas altérer la présentation de manière inattendue.

Une méthode pour résoudre ce problème consiste à filtrer les caractères restreints, à appliquer un codage sur tous les caractères afin que la présentation de ces caractères ne modifie pas l'interface graphique (les caractères ne sont pas interprétés par un moteur de rendu comme des codes de contrôle), ou simplement n'affiche pas le message. Bien que l'une de ces méthodes fonctionne, certaines sont plus appropriées dans certaines applications. En passant en revue l'exemple du forum de messages, il serait tout aussi mauvais qu'un adversaire puisse placer des scripts que d'autres utilisateurs peuvent copier et exécuter sans savoir ce qu'ils font. Par conséquent, au lieu de simplement restituer les informations d'une manière qui n'injecte pas HTML ou d'autres scripts dans la couche de présentation, les informations doivent être filtrées afin que les autres utilisateurs ne soient pas affectés.

Dans le cas où les données ne doivent pas être présentées à l'utilisateur, ceci ne concerne pas les données stockées et restituées par un adversaire. Au contraire, cette question concerne la manifestation de données qui ne sont pas destinées à la présentation directe ou publique et devrait être réservée aux administrateurs et aux ingénieurs. Par exemple, si une erreur interne est générée dans le traitement des informations, cette erreur ne doit pas être présentée à l'utilisateur avec ses données de débogage complètes. Cela peut permettre à un utilisateur d'identifier et d'instrumenter un bug dans le but d'exploiter les faiblesses de l'application. Ces informations doivent être consignées en interne et une erreur générique doit être signalée à l'utilisateur qui ne fournit pas suffisamment de contexte pour que l'utilisateur abuse du bogue. Même si l'utilisateur peut reproduire le bogue, l'utilisateur ne devrait pas être en mesure d'évaluer une différence dans la sortie de l'application qui indique des améliorations dans la méthodologie de l'exploit.

6.10.1 Risque

La validation de sortie est un attribut essentiel de la sécurité IoT. Les systèmes qui n'effectuent pas de validation de sortie risquent d'exposer les données critiques de l'utilisateur, les données liées à la confidentialité, les données de diagnostic, les messages d'erreur trop verbeux, etc. Ces messages peuvent être utilisés pour exposer des informations utilisateur ou peuvent être utilisés pour créer un exploit fiable contre un service en réseau.

6.11 Appliquer la stratégie de mot de passe fort

Il est impératif que tous les systèmes d'authentification appliquent des mots de passe forts lorsque des mots de passe sont requis pour l'authentification de l'utilisateur. La complexité du mot de passe a été une bataille constante entre les chercheurs en sécurité de l'information, les ingénieurs et les chefs d'entreprise. Les chefs d'entreprise veulent souvent que les utilisateurs puissent facilement se souvenir de leurs mots de passe. Les ingénieurs doivent réduire la complexité des interfaces, en particulier pour les concepteurs de la couche de présentation. Les chercheurs en sécurité de l'information surestiment souvent les compétences d'un attaquant et insistent sur la complexité inutile d'une technologie donnée.

La réponse, cependant, est quelque part entre toutes les exigences de chaque groupe. Les mots de passe doivent être longs, mais ne doivent pas être complexes. Alors que les mots de passe à huit caractères étaient la norme et que certains systèmes autorisent même 6 caractères au moment de l'écriture de ce document, la longueur du mot de passe devrait être supérieure à la norme de meilleure pratique. En imposant une longueur de mot de

passe plus longue, l'exigence de complexité est réduite. Au lieu d'imposer des assortiments bizarres de différents jeux de caractères, l'utilisateur peut simplement se souvenir d'une phrase. Parce qu'ils peuvent choisir d'utiliser des espaces blancs, des majuscules, des chiffres et des signes de ponctuation, la complexité augmente automatiquement pour tout attaquant appliquant la force brute.

N'oublions pas qu'il y a généralement quatre façons pour un pirate de compromettre un mot de passe :

- En volant la base de données de mots de passe et en craquant les mots de passe individuels
- En appliquant la « force brute » sur le service d'authentification de l'application
- En installant des logiciels malveillants
- En utilisant des mots de passe codés en dur ou par défaut

Forcer les mots de passe longs aide à diminuer le premier risque. Mais, la sécurité à la couche de l´ Écosystème des Services est beaucoup plus bénéfique. L'attaquant ne devrait pas être en mesure de récupérer la base de données de mot de passe en premier lieu, ce qui nous amène au deuxième point.

L'application de force brute sur les mots de passe est alors le moyen le plus efficace pour un attaquant d'abuser des mots de passe. Ce potentiel est considérablement réduit par un service d'authentification correctement conçu. Si un mauvais mot de passe a été employé, le système devrait automatiquement commencer à augmenter la quantité de délai nécessaire entre les suppositions suivantes. Ensuite, un seuil doit être défini pour limiter le nombre total de tentatives. Si l'attaquant atteint ce seuil, le compte doit être verrouillé et une authentification à deux facteurs, ou un autre modèle, doit être utilisée pour que l'utilisateur déverrouille et vérifie son compte. Ce type de sécurité réduit considérablement le bénéfice d'une attaque basée sur le réseau, ce qui nous amène au point final.

Les logiciels malveillants sur le système client doivent être traités par la plate-forme informatique ou par l'utilisateur qui installe la technologie combative appropriée. Ce n'est généralement pas quelque chose qui peut être sécurisé par l'application elle-même. Comme il n'y a rien ou presque que l'application peut faire pour combattre ce risque, l'ingénieur d'application aura réduit de manière satisfaisante la surface de menace des attaques par mot de passe contre le système d'authentification si c'est la seule solution viable pour l'adversaire.

Il faut noter, cependant, que la récompense pour la mise en œuvre de cette recommandation n'est pas élevée. En effet, quelles que soient les technologies utilisées pour réduire le risque d'attaque par authentification par mot de passe, les mots de passe sont essentiellement une ressource intangible. Ce ne sont pas des jetons physiques qui ne peuvent être capturés que par un seul individu. Au contraire, c'est un objet abstrait qui peut être copié à l'infini à travers les systèmes informatiques et à travers l'observation visuelle. Par conséquent, ils constituent une source d'authentification significativement faible qui n'indique en aucune manière de manière adéquate un utilisateur particulier. Par conséquent, les mots de passe, eux-mêmes, sont une faiblesse, et toute technologie utilisant des mots de passe est soumise aux risques inhérents aux mots de passe.

Les mots de passe ne doivent jamais être codés en dur dans le système. Pour les terminaux, des clés cryptographiques uniques doivent être générées. Reportez-vous au document sur les dispositifs périphériques pour plus d'informations sur l'approvisionnement de dispositifs périphériques. Pour les services et les interfaces utilisateur, le mot de passe doit être défini par l'utilisateur lors de son inscription. Le mot de passe, à ce moment-là, doit respecter les exigences strictes de sécurité par mot de passe. Ne permettez jamais à un utilisateur d'utiliser un mot de passe par défaut, faible ou mal conçu.

Assurez-vous que l'utilisateur a toujours la possibilité de changer son mot de passe à tout moment. Appliquer les exigences d'authentification forte et la sécurité des communications pour que l'utilisateur puisse changer son mot de passe. Lorsque cela est possible, activez l'authentification à deux facteurs (2FA) pour vérifier l'identité de l'utilisateur avant de permettre un changement de mot de passe. Toujours forcer un utilisateur à entrer à nouveau leur mot de passe d'origine lorsqu'ils soumettent un nouveau mot de passe au système. Cela garantit qu'un autre utilisateur n'a pas usurpé une application Web ouverte en profitant d'un ordinateur portable déverrouillé, ou volé un jeton de session d'application Web.

6.11.1 Risque

Les systèmes qui n'appliquent pas les contrôles de mot de passe adéquats risquent d'empêcher les adversaires de deviner facilement les mots de passe des utilisateurs du système.

6.12 Définir l'authentification et l'autorisation de la couche d'applications

Alors que la racine de confiance organisationnelle et ses services définissent les technologies d'authentification qui sécurisent la couche de communication des réseaux, les technologies d'utilisateur, d'administration et d'autorisation des partenaires doivent être configurées séparément. Alors que les canaux de communication de ces entités sont sécurisés avec la racine de confiance organisationnelle, leurs actions et identités doivent être authentifiées à l'aide d'un système distinct.

Généralement, cette authentification de couche d'applications sera facilitée par le même service. Cependant, l'information sera recueillie à partir d'une ressource distincte. Par exemple, il est préférable de placer les données d'authentification utilisateur et administrative dans des bases de données distinctes. Cela garantit que s'il existe un moyen de manipuler la base de données via la couche d'applications (par exemple, en utilisant une injection SQL), les attaquants ne peuvent que se déplacer latéralement dans la base de données utilisateur. Ils ne peuvent pas se déplacer verticalement, élevant leurs privilèges à ceux d'administrateur, sans compromettre la base de données, elle-même. C'est une amélioration significative de la sécurité organisationnelle.

Si possible, définissez des systèmes de stockage distincts pour :

- Les Identités de dispositifs périphériques
- Les Utilisateurs
- Les Informations d'identification de l'administrateur
- Les partenaires

Cela créera une séparation logique des tâches pour les applications et l'infrastructure, mais au sein de la même API d'authentification gérée par le service de la Racine Organisationnelle de la Confiance.

S'il vous plaît envisager d'utiliser le matériel des organisations suivantes pour aider à cette recommandation :

- OAuth 2.0 [8]
- OpenID Foundation [9]
- GSMA Mobile Connect [10]

6.12.1 Risque

Sans une méthodologie pour appliquer l'authentification et l'autorisation de la couche d'applications, il n'y a aucun moyen pour le système de confirmer que les actions censées provenir d'un utilisateur sont effectivement autorisées par cet utilisateur. L'implémentation de cette recommandation garantit que chaque action est traçable avec un utilisateur authentifié et une autorisation. Ces métriques peuvent être stockées et revues ultérieurement dans le cas où un compromis est suspecté. Sans ces étapes, il n'y aura pas de mesures de protection qui minimisent le risque d'abus.

6.13 Règles de pare-feu par défaut, ouvertes ou non ouvertes, et durcissement du système

Dans certains environnements d'infrastructure de service, les mécanismes de protection d'entrée et de sortie ne sont pas configurés par défaut. Cela signifie que les ingénieurs doivent utiliser eux-mêmes des règles de pare-feu ou des règles de trafic réseau. Ces règles doivent être définies dans l'infrastructure avant tout déploiement de service au public.

Pourtant, il arrive que ces technologies ne soient pas suffisantes pour protéger l'infrastructure de service. Parfois, les pares-feux et autres systèmes de protection du trafic réseau échouent. Lorsque ces systèmes échouent, ils échouent souvent avec le pare-feu ouvert au trafic. La raison en est que si le système tombe en panne, le trafic doit toujours pouvoir fonctionner, car le trafic pour d'autres environnements informatiques sera acheminé via l'infrastructure avec le trafic du fournisseur de services IoT. Ainsi, le trafic ne peut pas s'arrêter brusquement. Par conséquent, le système échoue souvent avec les ports de service ouverts pour permettre à autant de services que possibles de continuer à travailler.

L'équipe d'ingénierie doit utiliser le renforcement du système d'exploitation pour s'assurer que les effets d'une infrastructure défaillante n'entraînent pas un événement de sécurité catastrophique. Au lieu de cela, cela signifie simplement que plus de connexions peuvent être établies avec l'infrastructure de service existante.

Par exemple, les services qui doivent être cachés ne doivent pas être placés derrière des technologies telles que les pares-feux. Au lieu de cela, des réseaux privés virtuels (VPN) ou d'autres protections de haute sécurité peuvent être utilisés pour sécuriser les services contre des adversaires.

Notez que les pares-feux logiciels comportent un risque supplémentaire, en ce sens qu'ils peuvent être manipulés par un attaquant avisé. Si un pare-feu logiciel est utilisé, toute infrastructure de serveur mal durcie peut être manipulée par un attaquant. En d'autres

termes, si un service public exécuté sur un serveur porte des privilèges superflus (tels que des privilèges de super-utilisateur) et est compromis, l'attaquant sera probablement capable de désactiver le pare-feu logiciel. Ainsi, l'équipe d'ingénierie doit évaluer si un pare-feu logiciel présente un risque trop élevé pour l'architecture choisie.

6.13.1 Risque

Sans utiliser de stratégies pour compenser les défaillances dans les systèmes de sécurité du trafic réseau, l'environnement sera soumis à des attaques inutiles qui pourraient être facilement évitées grâce à des stratégies standard de renforcement des services.

6.14 Évaluer le modèle de confidentialité des communications

La confidentialité des communications est un sujet légèrement différent de celui de la confidentialité des applications (décrit ci-dessus) ou de la sécurité des informations de communication. Alors que la confidentialité est largement évaluée à partir de la capacité des tiers à lire ou à intercepter efficacement les données, la confidentialité et l'intégrité ne représentent pas toute la portée de la confidentialité des communications.

Les autres problèmes qui affectent la confidentialité des communications incluent :

- Unicité cryptographique de chaque message
- Modèles de transmission
- Métadonnées en clair
- Adresses matérielles ou numéros de série attribuables

Même si chaque message doit être confidentiel et avoir une intégrité vérifiable, il doit également être unique sur le plan cryptographique. Si certains messages sont envoyés en réponse à des événements prévisibles par un adversaire, toutes les réponses qui ne sont pas cryptographiquement uniques peuvent être retransmises par l'attaquant. Chaque message doit être unique pour interdire à l'attaquant de capturer et retransmettre des messages qui soient effectifs.

Les modèles en transmission peuvent permettre à un adversaire d'identifier un utilisateur particulier, ou assimiler un comportement à une certaine action attribuable. Par exemple, une technologie qui émet un message contenant des empreintes digitales, lorsqu'un utilisateur rentre dans une certaine zone dans un édifice par exemple, peut être capturé par des « renifleurs » qui sont capables de recevoir ces messages lorsqu'ils sont transmis sur un réseau sans fil. Bien que cela ne soit pas intuitif, il s'agit d'une responsabilité légale potentielle à tenir en compte si un adversaire peut identifier qui se trouve dans un emplacement physique et où il se trouve à cet endroit. Les schémas de réseau doivent être évalués pour déterminer s'il existe un moyen simple permettant aux adversaires de transformer les schémas de transmission en données exploitables.

Les métadonnées ont longtemps été utilisées par les services de renseignement pour évaluer le contexte des systèmes de messagerie sans nécessiter de mandat ou d'autre accès légal aux données cryptées. Souvent, les métadonnées ont suffisamment d'informations pour qu'une organisation puisse créer une série de données exploitable. Cependant, maintenant les amateurs, les organisations criminelles et les utilisateurs curieux sont en mesure d'utiliser des métadonnées pour le suivi et d'autres fins potentiellement néfastes. Par conséquent, il est plus important que jamais de réduire la quantité de

métadonnées disponibles pour les tiers. Dans la mesure du possible, limitez la quantité de métadonnées à seulement suffisamment d'informations requises pour qu'un pair de communication évalue si le message leur est destiné.

Dans cette optique, l'adresse physique du module de communication et tous les numéros de série uniques doivent être protégés ou randomisés, si possible. Par exemple, Apple a modifié le modèle iOS pour examiner les points d'accès Wifi. Au lieu d'utiliser l'adresse matérielle statique, ils ont changé leur technologie pour utiliser une adresse physique randomisée, ce qui réduit le potentiel pour quelqu'un de suivre l'emplacement d'un utilisateur en fonction des analyses des réseaux Wifi actifs qu'il a visité. La technologie IoT fonctionnera de la même façon, mais aura un plus grand nombre de technologies de communication affectées par ce problème. Certaines technologies ne seront pas capables de générer des adresses matérielles aléatoires, comme pour les réseaux cellulaires. Mais d'autres, tels que 802.15.4, Wifi et Bluetooth, peuvent en être capables en fonction des fonctionnalités du micro-logiciel.

6.14.1 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 peut pas lire les données. Cela garantit également :

- L'identité d'un dispositif périphérique ne peut pas être usurper
- L'identité d'un service critique ne peut pas être usurpé
- Les messages abusés 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.

7 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 sécurité au niveau du système d'exploitation n'est valide que si un système d'exploitation s'exécute sur le dispositif périphérique. Si ce dernier est composé d'une application de noyau monolithique ou d'un système RTOS (Real-Time Operating System) intégré avec une seule application intégrée, la recommandation peut ne pas être d'application. Lorsque des recommandations s'appliquent à la conception du dispositif périphérique, elles devraient être mises en œuvre.

7.1 Définir un environnement d'exécution d'application

Plusieurs points doivent être faits sur les environnements d'exécution des applications :

- Le langage de programmation utilisé peut avoir un lien direct avec la sécurité :
  - Des langages comme PHP et Ruby peuvent présenter des problèmes de sécurité
  - Des langues comme GoLang et Erlang peuvent réduire les risques

- Les bibliothèques tierces doivent être surveillées, gérées et auditées en fonction des risques :
  - Certaines bibliothèques ne sont pas bien entretenues
  - Certaines bibliothèques n'ont jamais été auditées pour des failles de sécurité
  - Certaines bibliothèques nécessitent des dépendances obsolètes qui présentent des failles de sécurité connues
- Toujours exécuter une application en tant qu'utilisateur non privilégié :
  - Si l'application requiert une ressource privilégiée, utilisez une fonction d'enveloppe (« wrapper ») pour approvisionner cette ressource avant de supprimer les privilèges et d'exécuter l'application complète.
- Utilisez un modèle TCB et Bootstrap bien défini :
  - Les applications qui ont des environnements bien définis sont plus fiables et plus sécurisées

S'il vous plaît envisagez d'utiliser le matériel des organisations suivantes pour aider à cette recommandation :

- OWASP [5]

7.1.1 Risque

Les applications déployées avec une architecture sécurisée peuvent faire l'objet de compromis qui ne peuvent pas facilement être retracés vers une source spécifique. Les outils et les techniques pour compromettre les services et les applications ont progressé au cours de la dernière décennie. Certaines technologies « open source », telles que « Metasploit », permettent le développement et l'intégration d'exploits personnalisés dans une plate-forme d'attaque qui peut fournir des technologies pour augmenter la furtivité d'une attaque.

Un environnement d'exécution d'application sécurisé peut lutter contre ce risque en sécurisant la façon dont les applications sont exécutées, interagissent les unes avec les autres et les types de technologies utilisés pendant l'exécution. Ces attributs peuvent non seulement réduire la probabilité d'un compromis, mais ils peuvent également ajouter des fonctionnalités de traçabilité et d'enregistrement critique pour suivre et diagnostiquer la vulnérabilité abusée.

7.2 Utiliser les services de surveillance améliorés par les partenaires

Si le partenaire utilisé est un opérateur de réseau mobile, indiquez s'il est en mesure d'offrir des services de surveillance. Certains opérateurs de réseau sont capables d'analyser le comportement des dispositifs périphériques qui se communiquent à travers de leur réseau. Les opérateurs disposant de ce type de capacité ont l'expérience d'évaluer si les mesures faites peuvent représenter un comportement anormal et contradictoire.

Cela permettra à l'entreprise IoT d'identifier plus rapidement si un utilisateur ou dispositif périphérique spécifique est une menace ou a été compromis par un adversaire. En conséquence, les entreprises peuvent réagir plus efficacement pour prévenir les attaques contre d'autres éléments de l'infrastructure de l'entreprise.

La complexité de ce service provient de la capacité de l'opérateur de réseau à fournir l'information dans un délai significatif. Si l'opérateur de réseau ne peut fournir les

informations une fois que l'adversaire a déjà attaqué l'écosystème IoT, les systèmes de surveillance et de journalisation placés dans l'infrastructure de l'entreprise IoT doivent être capables de détecter le comportement. Cependant, si l'opérateur de réseau est capable de notifier un comportement suspect d'attaque dans la couche de réseau et peut identifier quel abonné a émis un trafic réseau anormal, l'entreprise peut être en mesure de limiter l'exposition de l'écosystème IoT en isolant et à la limite interdisant ce trafic de l'utilisateur.

7.2.1 Risque

Le fournisseur de services IoT peut compter sur certaines technologies qui ne peuvent pas être surveillées par le fournisseur de services IoT. Une telle technologie est le réseau de communication qui relie un dispositif périphérique aux écosystèmes de services et de réseaux. Sans surveillance des services, le fournisseur de services IoT n'aura pas de manière d'inspecter les événements survenant dans le réseau lui-même. Ainsi, si une identité au niveau de l'application A tente de compromettre un service, l'organisation ne pourra pas identifier que l'extrémité B est en fait l'unité qui s'est connectée au réseau de communication. Cette lacune dans les informations est critique, car l'organisation peut attribuer l'attaque à l'identité A plutôt qu'à un dispositif périphérique B compromis.

7.3 Utiliser un APN privé pour la connectivité cellulaire

Le nom du point d'accès (APN) est un composant de communication cellulaire qui connecte le réseau sans fil à Internet. Ce point agit essentiellement comme un réseau privé virtuel (VPN) entre le dispositif périphérique cellulaire et l'infrastructure de service avec laquelle il doit interagir. Un APN privé (parfois appelé APN sécurisé) est une version d'un APN qui a été sécurisé pour implémenter plusieurs contrôles souhaitables :

- Accès limité restreint aux clients authentifiés
- Pare-feu
- La communication entre dispositifs périphériques est désactivée à la force
- Services de surveillance pour la détection d'anomalies
- Services de sécurité ou de surveillance en option

En restreignant l'accès à l'APN, une organisation peut s'assurer que seuls les dispositifs périphériques authentifiés sont autorisés à se connecter à l'infrastructure de services rendue disponible via l'APN. Cela diminue la possibilité que n'importe quels clients sans fil, pirates ou non, se connectent à l'APN et accèdent aux services restreints. En outre, cela permet à l'organisation d'identifier les clients qui se comportent de manière anormale, ce qui permet à l'organisation de lier un comportement négatif à un dispositif spécifique ou à un utilisateur spécifique.

Le pare-feu garantit que les entités attachées à l'APN à la fois du côté client (Écosystème de Dispositifs Périphériques) et du côté service (Écosystème de Services) ne peuvent pas se communiquer en utilisant des canaux non approuvés. Cela limite également la possibilité pour un dispositif périphérique d'abuser de l'APN en tant qu'un possible chemin vers l'Internet ouvert, et de couper le trafic vers un ensemble spécifique de services approuvés.

Les restrictions de communication de dispositifs périphériques garantissent que ceux que soient pirates ne puissent pas attaquer d'autres dispositifs en utilisant l'APN en tant qu'un réseau ouvert. Au lieu de cela, toute communication doit passer par des services approuvés

par l'organisation. Si vous le souhaitez, l'organisation peut interdire entièrement la communication point à point.

Les services de surveillance améliorent les renforcements de la sécurité qui seront apportées par l'organisation dans la surveillance de l'infrastructure Cloud ou de services existante. En associant ces services de surveillance existants aux technologies APN et de surveillance réseau offertes par l'opérateur de réseau, l'organisation peut plus facilement détecter la source d'un comportement anormal. Cela permet à l'organisation d'inspecter plus profondément les incidents qui se produisent sur son dispositif périphérique ou son infrastructure de service. Par exemple, si la couche d'applications indique que l'utilisateur A peut être compromis, mais que l'équipement de l'utilisateur B établit la connexion authentifiée avec l'APN, l'organisation pourra utiliser les services de surveillance APN pour identifier que l'utilisateur B a potentiellement compromis l'utilisateur A, ou un adversaire a compromis à la fois l'utilisateur A et B.

Les opérateurs de réseau ont des services supplémentaires qui peuvent être superposés aux services décrits ci-dessus. Ces services aideront à mettre en liste noire les mauvais acteurs du réseau, à surveiller des utilisateurs ou des groupes d'utilisateurs spécifiques et à rediriger certains types de trafic pouvant indiquer des anomalies. D'autres options peuvent être disponibles. Contactez l'opérateur réseau pour déterminer quels services sont adaptés à votre organisation.

Bien que l'utilisation simultanée de tous ces services puisse sembler difficile, travailler avec l'opérateur de réseau simplifiera le processus et simplifiera l'intégration de ces offres dans l'infrastructure existante de l'entreprise. La complexité proviendra de l'utilisation efficace des données et nécessite une équipe d'ingénierie capable de traiter et de gérer les données de manière raisonnable. Certains services peuvent entraîner des frais supplémentaires. Déterminez quel modèle de tarification et quels services fonctionneront le mieux pour votre entreprise.

7.3.1 Risque

Sans APN privé, un dispositif périphérique peut se connecter à presque n'importe quel service ou technologie, y compris en établissant des connexions directes avec d'autres dispositifs périphériques sur l'APN, ou un service arbitraire sur Internet. Comme cela permettrait à un dispositif compromis d'interagir avec presque n'importe quel service sur Internet, et pourrait faire de celui-ci une cible pratique pour agir comme un proxy pour attaquer des réseaux ou services plus sécurisés, cette recommandation devrait être appliquée pour restreindre la capacité des dispositifs périphériques à faire des connexions arbitraires et non autorisées. Il est beaucoup plus précieux pour l'entreprise et pour la sécurité de l'ensemble de l'écosystème IoT lorsque les dispositifs sont forcés de se connecter uniquement aux services approuvés.

7.4 Définition d'une stratégie de distribution de données tiers

Une fois que les classifications de sécurité ont été définies et qu'une classification valide a été attribuée aux types de données et qu'une stratégie de violation a été adoptée, une stratégie de distribution de données doit être établie. Une politique de distribution de données décrit comment celles-ci doivent être traitées avec des contrôles techniques et transmises aux applications des services ayant reçu l'autorisation d'accéder aux données.

Le modèle d'autorisations fait partie de la stratégie de distribution de données et correspond à la capacité de l'utilisateur à créer des autorisations de données granulaires.

Bien qu'une politique de distribution de données puisse être hautement descriptive, plusieurs éléments clés peuvent aider à définir une stratégie réussie :

- Quel niveau d'authentification mutuelle est requis pour le trafic de ces données
- Quelle est la confidentialité et l'intégrité des données requises
- Quelle est la capacité de l'entreprise à conserver les données
- Quelle est la capacité du partenaire à conserver les données ?
- Si la conservation est autorisée, quelle période de temps les données peuvent-elles être conservées
- Quel niveau de sécurité de stockage doit être appliqué aux données
- Quelle classification de sécurité d'accès doit être appliquée aux données

7.4.1 Risque

Les stratégies de distribution de données imposent des exigences de sécurité aux partenaires qui peuvent ne pas adhérer au même niveau de sécurité interne que le fournisseur de services IoT. Étant donné que le fournisseur de services IoT ne peut pas contrôler la sécurité qu'un partenaire a implémentée dans ses services internes et son réseau, le fournisseur de services IoT peut uniquement appliquer les données fournies à un partenaire de manière sécurisée. Sans cette définition, le partenaire peut appliquer des configurations non sécurisées susceptibles d'exposer des données utilisateur à des adversaires alors que les données sont toujours sous le contrôle du fournisseur de services IoT. En imposant des contrôles de sécurité stricts pour le canal de communication, le fournisseur de services IoT prouve qu'il fait tout ce qu'il peut pour assurer la sécurité jusqu'à ce que les données soient hors de son contrôle.

7.5 Construire un filtre de données tiers

L'acceptation de données générées dynamiquement, telles que la publicité, d'un partenaire nécessite un certain niveau de présomption concernant la qualité et la sécurité des données. Au lieu de faire des présomptions et d'appliquer les données à la couche de présentation, l'équipe d'ingénierie doit prendre des mesures pour s'assurer que les données distribuées depuis l'application de service vers ou depuis un partenaire sont bien formées et ne contiennent pas de contenu potentiellement malveillant.

Pour ce faire, l'équipe d'ingénierie doit prendre en compte le modèle suivant :

- Les données correspondent-elles au format défini par le partenaire pour le modèle de données ?
- Les données sont-elles bien formées ?
- Les données représentent-elles un objet polymorphe pouvant être mal interprété par le client ?
- Les données affecteront-elles la façon dont le client présente la couche de présentation ?
- Les données affecteront-elles la façon dont le client interprète la couche de présentation ?

- Les données incitent-elles ou demandent-elles à l'utilisateur à adopter un comportement qui affaiblirait la sécurité ?
- Est-ce que les données « parodient » ou usurpent l'identité d'un composant (champ de saisie de mot de passe) de l'interface graphique du client?

Rejeter toutes les données qui ne correspondent pas à un modèle approuvé. Informer l'administration dès la détection de ces données et inclure autant de mesures que possible concernant l'origine et le format des données. Consignez un échantillon, si possible, dans une base de données sécurisée.

7.6 Risque

Les données générées dynamiquement par des tiers pourraient contenir des logiciels malveillants, du contenu inapproprié ou d'autres données indésirables, intentionnellement ou non. En l'absence d'un filtre d'entrée orienté vers la définition du service tiers, l'organisation risque de permettre accidentellement à des utilisateurs malveillants ou à d'autres contenus malveillants d'atteindre l'utilisateur final. Cela peut entraîner des compromis du système, ou tout simplement à la perte de clients, en raison des effets secondaires de ces données.

8 Recommandations de basse priorité

Les recommandations de basse priorité englobent l'ensemble de 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 l'information détaillée dans les recommandations soit importante, 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 considérer ces risques, appliquez les recommandations.

8.1 Attaques « Rowhammer » et similaires

Certaines implémentations de la technologie RAM moderne telles que la mémoire DRAM (« Dynamic Random Access Memory ») et la SRAM (« Static Random Access Memory ») sont vulnérables aux erreurs qui peuvent être induites par certaines séquences d'accès mémoire. Abuser de ce type d'erreur peut entraîner la modification d'un ou plusieurs bits spécifiques dans des zones de mémoire prévisibles. Un exploit réussi de cette condition peut altérer des bits dans la mémoire qui représentent des types de privilèges désignés par logiciel.

En d'autres termes, s'il est exploité correctement, un adversaire peut élever ses privilèges d'un utilisateur à un autre en manipulant une faille matérielle dans les implémentations modernes de DRAM ou de SRAM. De nombreuses implémentations modernes de DRAM et de SRAM ont été trouvées exploitables grâce à cette vulnérabilité. Cependant, il nécessite la possibilité d'exécuter du code sur le système local afin de créer les séquences d'accès mémoire capables de déclencher ce bogue.

Et pourtant, il peut être possible de déclencher ce type de comportement à distance via des langages d'exécution tels que GoLang en « sandbox », Python, Erlang, etc. Cependant, la précision de ces types d'attaques n'a pas encore été documentée, et il est hautement improbable qu'ils fonctionnent efficacement comme un exploit.

Cette attaque doit être résolue au niveau matériel. Cependant, les ingénieurs peuvent réduire le risque d'abus en interdisant aux clients d'exécuter du code, même via une machine virtuelle ou en temps d'exécution, sur un service donné. En limitant cette capacité, les ingénieurs pourront empêcher les adversaires de créer les séquences d'accès à la mémoire requises pour cette attaque.

8.1.1 Risque

Sans gardes suffisantes contre ce type d'attaque, les adversaires peuvent être en mesure d'élever à distance le privilège ou d'exécuter du code arbitraire contre un hôte ciblé. Il convient toutefois de noter qu'une attaque réussie nécessite une connaissance extrêmement approfondie du matériel, du système d'exploitation, du vecteur d'attaque et d'autres facteurs qui rendent cette attaque improbable et rare.

8.2 Compromis de machine virtuelle

L'infrastructure de service moderne utilise souvent des machines virtuelles pour déployer des services à la demande. Bien que ce modèle se soit avéré extrêmement pratique et facile à déployer, le problème avec cette méthodologie est la sécurité de l'infrastructure globale. Alors que l'équipe d'ingénierie peut réussir à déployer une architecture bien pensée, l'organisation qui gère et déploie l'infrastructure virtuelle peut ne pas réussir.

L'une des préoccupations majeures du déploiement dans les environnements de serveurs virtuels est la possibilité de compromettre les hôtes ou de permettre aux serveurs (invités virtuels) d'intercepter les données d'autres invités s'exécutant sur la même infrastructure.

Bien que ces attaques soient des préoccupations valides qui devraient être évaluées par le fournisseur de services IoT, elles nécessitent souvent beaucoup de temps et de compétences pour se perfectionner. Ainsi, il est possible qu'une attaque survienne, mais ce sera probablement un événement rare. Cependant, si l'infrastructure de service n'est pas bien protégée, il est possible que des adversaires puissent compromettre l'accès administratif aux machines virtuelles. Ce type de violation peut ne pas exiger une grande quantité de compétences pour réussir.

Un moyen de lutter contre ce problème consiste à utiliser un service de provisionnement pour les serveurs. Ce processus garantit que chaque serveur est codé avec un ensemble unique de clés cryptographiques. Si ce processus est suivi, toute compromission à un seul serveur peut être limitée à ce seul serveur.

8.2.1 Risque

Le risque de ne pas s'adapter à ce type d'attaque peut rendre l'infrastructure de service vulnérable à de nombreux types d'attaques. L'usurpation de serveur à l'aide de clés accessibles à partir de l'infrastructure de service, l'exfiltration de données, la compromission de la confidentialité et l'usurpation d'identité de l'utilisateur peuvent être possibles.

8.3 Créer une API pour les utilisateurs afin de contrôler les attributs de confidentialité

Tous les utilisateurs doivent être en mesure de contrôler les informations qu'ils transmettent à des tiers via des API de service. Les informations doivent être classées en types de données et attribuées avec des classifications de sécurité. Les utilisateurs devraient être en

mesure de récupérer les types de données et les classifications qui sont utilisés dans la modélisation de leur compte d'utilisateur. L'utilisateur devrait pouvoir appliquer des contraintes aux types de données, pour leur permettre d'accorder ou de révoquer l'accès à ces données aux partenaires.

Cela peut prendre la forme d'une API authentifiée ou d'une interface graphique qui permet de simples contrôles « Oui ou Non » de manière générale et par partenaire.

8.3.1 Risque

Sans la possibilité pour les utilisateurs de contrôler les données qu'ils contribuent à un fournisseur de services IoT, ils courent le risque de voir leurs données exposées en cas de violation de sécurité au niveau du fournisseur de services ou de l'un des partenaires utilisés par le fournisseur de services. Étant donné que certains utilisateurs courent un risque beaucoup plus élevé que d'autres, chaque utilisateur devrait être en mesure de régler ses restrictions de confidentialité en fonction de ses besoins personnels. Rendre cette interface disponible permet de s'assurer que la capacité est là. L'utilisateur a la reponsabilité d'ajuster les contrôles en fonction de ses besoins. Par exemple, oneM2M (via TS-0003) permet à l'utilisateur de définir les préférences de confidentialité pour un fournisseur de services.

8.4 Définir un modèle d'évaluation de faux négatif ou faux positif

Alors que l'analyse des faux positifs est un sujet extrêmement complexe, il existe un moyen simple d'identifier si une technologie est plus susceptible de présenter des faux positifs. C'est en évaluant les éléments suivants :

- La source de données est-elle fiable ?
- La source de données peut-elle être falsifiée ou usurpée ?
- La source de données est-elle du domaine analogique ?
- Les données peuvent-elles être corroborées à partir de plusieurs points d'origine ?
- Les sources de données corroborées existent-elles sur le même système de dispositifs périphériques ?
- Les sources de données corroborées sont-elles faciles à altérer ou à usurper ?
- Les outils sont-ils facilement disponibles pour manipuler la source de données ?
- Quel niveau d'expertise ou quel coût est nécessaire pour manipuler la source de données ?
- L'appareil connecté à la source de données est-il digne de confiance ?

Tous ces attributs, et plus, peuvent être utilisés pour évaluer si les données sont fiables. Ceci est extrêmement important, car les décisions critiques qui affectent le monde physique peuvent entraîner des effets potentiellement nocifs. Il est impératif que l'équipe d'ingénierie crée un modèle de fiabilité et l'applique à chaque source de données impliquée dans la prise de décisions critiques. Si le poids de la source de données est tel qu'il ne peut pas être fiable, l'action la plus rationnelle et la plus sûre doit être prise.

Il est important de noter que l'équipe d'ingénierie n'est pas la seule entité à prendre ce genre de décision. Le chef d'entreprise, l'équipe d'avocats et l'équipe d'assurance devraient être impliqués dans la détermination de la réaction correcte dans des scénarios potentiellement dangereux. Les ingénieurs doivent ensuite encoder le processus de prise de décision correct dans la technologie d'une manière vérifiable et reproductible.

Ce processus est très difficile car il exige l'attention de toute l'entreprise sur la façon dont la technologie devrait réagir dans des scénarios critiques. La fiabilité est un attribut difficile à appliquer à une technologie, en particulier à une technologie embarquée.

8.4.1 Risque

Sans un modèle d'évaluation des faux positifs, les ingénieurs peuvent passer trop de temps à analyser des événements bénins alors que d'autres événements plus dangereux se produisent. Cela peut entraîner un risque accru que les mesures analysées par l'organisation n'indiquent pas clairement quels types d'événements se produisent dans la production. Cela dévalue l'infrastructure de journalisation et de surveillance, et annule la capacité de l'organisation à utiliser ces ressources coûteuses à son avantage.

9 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 éléments intelligents 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 questions de sécurité fréquemment posées 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 sa définition de l'architecture du système IoT, elle peut réviser des recommandations spécifiques lorsque leurs questions et préoccupations de sécurité deviennent plus claires par rapport à leur propre implémentation.

Dans l'ensemble, chaque équipe d'ingénierie devra faire face à des risques très similaires. Il est impératif que l'organisation choisisse de partager ses préoccupations avec ses collègues afin de construire une base de connaissances commune pour les risques et les stratégies de remédiation. Ensemble, les organisations de l'entreprise 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 Gestion du document

A.1 Historique du document

Version	Date	Brève description du changement	Autorité d'approbation	Éditeur / Société
1.0	08-Feb-2016	Nouvelle version PRD CLP.12	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 sur OneM2M additionnelle ajoutées	IoT Security Group	Rob Childs GSMA

A.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.