Eine Themenbeilage der
Heise Medien GmbH & Co. KG

SICHERHEIT & DATENSCHUTZ

I 2019

IoT, Verschlüsselung und Industrie 4.0

IoT-Security:
Wie sich vernetzte Geräte absichern lassen

Log- und Protokollmanagement:
Wann SIEM-Lösungen Alarm schlagen

IIoT-Sicherheitsarchitektur:
Was Servicetechnikern die Arbeit erleichtert

E-Mail-Verschlüsselung:
Warum Johnny jetzt seine Mails verschlüsselt

Digitale Signatur:
Wie die Hausbank zum ID-Provider wird

Zertifikatsmanagement:
Womit sich das MIM-Tool effektiver nutzen lässt

Quantum Computing:
Was die Quantentechnologie mit sich bringt

www.it-security-und-management.de

powered by: TeleTrusT Pioneers in IT security. iX

Geräte-Service per Tablet

Sicher und benutzerfreundlich: SSH-Authentifizierung und NFC-Smartcards

Eine ausgereifte IIoT-Sicherheitsarchitektur basiert auf erprobten und bewährten Sicherheitsprotokollen, integriert diese aber in ein modernes Anwendungsszenario mit Smartcards und Tablets. Sie ist sicher und robust und bietet dennoch eine hohe Benutzerfreundlichkeit für Servicetechniker.

Die Vision einer Industrie 4.0 kann nur durch eine übergreifende Vernetzung von Industrieanlagen umgesetzt werden. Die Einführung eines solchen „Industrial Internet of Things“ (IIoT) geht mit erhöhten Sicherheitsanforderungen an die initiale Inbetriebnahme, Administration und Konfiguration dieser Geräte einher. Da diese Tätigkeiten normalerweise von Servicetechnikern der Maschinenbauunternehmen durchgeführt werden, sollte dieser Prozess möglichst komfortabel und einfach sein.

Kritische Infrastrukturen

Mit der zunehmenden Digitalisierung von industriellen Anlagen innerhalb kritischer Infrastrukturen sind eine Reihe von Bedrohungen verbunden. So arbeiten diese Anwendungen nicht nur mit hochsensiblen, geschäftskritischen Daten der Hersteller, physische Aktoren und Effektoren können im schlimmsten Fall auch die Anlage selbst zerstören oder Menschenleben bedrohen. Einschlägige Normen wie die IEC 62443 sehen hier eine Reihe von Gegenmaßnahmen vor, um unbefugten Zugriff zu verhindern, darunter physisch getrennte Schlüsselspeicher und die Nutzung von Public-Key-Kryptografie. Im Folgenden wird eine Auswahl der Anforderungen an die Login-Mechanismen dargestellt, die für Level 2 und höher auf einer Skala von 0 (keine Sicherheit) bis 4 (gesichert gegen staatliche Akteure) notwendig sind.

Menschliche Nutzer müssen identifiziert und authentifiziert sein. So muss auf jedem Interface eine vorherige Authentifizierung stattfinden. Für Security Level 2 und höher muss jede Person individuell, eindeutig authentifiziert werden. Für Maschinenkommunikation gilt entsprechend, dass jede Maschine und jeder Softwareprozess eindeutige Login-Credentials nutzt. Identifikatoren müssen in ein Managementsystem integrierbar sein. Sie müssen zurückziehbar und erneuerbar sein. Für Level 3 gilt ein zusätzlicher Hardwareschutz wie TPMs oder SEs. Für Public-Key-Kryptografie gilt hier selbstverständlich die Nutzung von generell als sicher anerkannten kryptografischen Verfahren und Primitiven. So gilt etwa RSA mit Schlüssellängen unter 2048 Bit sowie ECC mit Schlüssellängen unter 256 Bit als nicht mehr zeitgemäß. Revozierungsmechanismen müssen vorhanden und verwendbar sein.

Industrial Internet of Things

Industrielle IoT-Geräte werden in unterschiedliche Kategorien aufgeteilt, je nachdem, ob sie an der Maschine selbst platziert sind, oder in einer speziell gesicherten Zone in einem Rechenzentrum. Generell gilt, je mehr potenziell nicht vertrauenswürdige Personen ein Gerät im physischen Zugriff haben und je höher der Schaden bei eingetretenem Sicherheitsvorfall ist, desto höher ist der Schutzbedarf. Häufig finden sich in diesem Umfeld eingebettete Linux-Varianten, die auf Industriecomputern auf Basis von x86- oder ARM-Prozessoren laufen. Als solche stehen im Allgemeinen die typischen Linux-Administrationswerkzeuge wie SSH zur Verfügung. Oft wird ein VPN-Tunnel zu einem gesicherten Backend aufgebaut, sodass bei ordentlicher Konfiguration keine Netzwerk-Ports aus dem öffentlichen IP-Netz offen sind.

Dies birgt jedoch Herausforderungen, wenn ein Gerät initial deployed werden soll. Da solche Geräte häufig kein vollwertiges User-Interface besitzen, können initiale Netzwerkparameter oft nur umständlich vordeployed werden. Hier bieten moderne Mobilgeräte die Möglichkeit, z. B. mittels NFC, Bluetooth oder USB eine Verbindung zu dem Gerät aufzubauen und – über entsprechende Protokolle, Certificates und Schlüssel geschützt – ein temporäres lokales UI bereitzustellen. So können etwa Android Open Accessory und WebUSB genutzt werden, um ein Smartphone oder Tablet mit einem Industrie-Gateway zu verbinden. Das Industrie-Gerät wird als Accessory erkannt und gibt direkt einen Hinweis, welche App notwendig zur Konfiguration ist. Über einen Bulk-Transfer-Endpunkt können beliebige Daten ausgetauscht werden. Wir haben uns hier dazu entschlossen, diese Daten an den lokalen SSH-Unix-Dämon weiterzuleiten, um auf bewährte Sicherheitsmechanismen und Konzepte aufsetzen zu können.

OpenSSH User Certificates

Die Nutzerauthentifizierung erfolgt mit Public-Key-Kryptografie. Hierbei wird jedem Nutzer ein unabhängig generiertes Public-/Private-Key-Paar zugewiesen. Um Security Level 3 des IEC 62443 zu erreichen, werden diese auf Smartcards generiert und gespeichert. Diese bieten ein wesentlich höheres Sicherheitsniveau als Passwörter, da Nutzer oft schwache Passwörter wählen und komplexe Passwörter nur schwer auf Tablets eingegeben werden können.

Zur Autorisierung der Nutzer und Verwaltung der Zugangskontrolle werden OpenSSH User Certificates genutzt. Ähnlich wie X.509 Client Certificates für eine anwenderseitige TLS-Authentifizierung genutzt werden können, ist dies auch bei SSH möglich. Jedoch sind OpenSSH User Certificates einfacher umgesetzt. Es werden keine komplexen Kodierungsregeln und mehrstufigen Zertifizierungsebenen genutzt. Das vereinfacht die Implementierung und vermindert die Angriffsoberfläche. Diese einstufige CA-Infrastruktur erlaubt also zusammen mit einer Public-Key-Infrastruktur die Zugriffs-Autorisierung von Nutzern.

Viele Server- und Client-Implementierungen von SSH unterstützen User Certificates bereits. In Client-Implementierungen können sie außerdem leicht nachgerüstet werden, da die eigentliche Challenge-Response-Authentifizierung von SSH nicht geändert werden muss. Eine Anpassung des publickey-Formats reicht aus. Als konkreten Algorith-

mus empfehlen wir „ecdsa-sha2-nistp384-cert-v01@openssh.com“ für die NIST-Kurve P-384.

NFC-Smartcards

Bei den eingesetzten Smartcards wird auf bewährte JavaCard-Technologie gesetzt. Typischerweise werden Smartcards an Desktop-Systemen mit externen Smartcard-Readern genutzt, an Laptops mit dem integrierten Reader. Oft sind dies kontaktbehaftete Smartcards. Für eine Nutzung über die NFC-Schnittstelle von Tablets bieten sich kontaktlose oder hybride Smartcards an. Die JavaCard-Plattform stellt eine API bereit, um eigene Smartcard-Standards zu implementieren. Als eigentlicher Standard kann also die OpenPGP Card Specification, NIST Personal Identity Verification (PIV) oder auch ISO 7816-4 implementiert werden. Wichtig ist, dass Schlüssel immer auf den Smartcards selbst generiert werden und eine Extraktion von Schlüsselmaterial nicht möglich ist. Das zugehörige SSH User Certificate wird ebenfalls auf der Smartcard gespeichert und ist somit geräteunabhängig verfügbar.

Basierend auf den Sicherheitsanforderungen kann die Authentifizierung zwischen Smartcard und Tablet durch eine PIN gesichert werden. Durch den Hardwareschutz der Smartcard muss diese nur aus vier Zahlen bestehen. Smartcards werden mit den Namen der Techniker personalisiert und bedruckt, in einem sicheren Zustand ausgeliefert und eine PUK sicher an den Hersteller übertragen. So kann der Techniker vor Ort eine eigene PIN wählen und setzen.

Public-Key-Infrastruktur

Die Nutzung von OpenSSH User Certificates erlaubt wie bereits beschrieben den Aufbau einer einstufigen CA-Infrastruktur. Um die Sicherheit der ausgestellten Certificates zu gewährleisten, untersteht diese jedoch einem besonderen Schutzbedarf. So sollte der private Schlüssel der CA in einem HSM gesichert und der Ausstellungsprozess durch adäquate Mechanismen geschützt sein. Typische CA-Software nutzt hier eine vorgeschaltete Registration Authority. Die Kartenpersonalisierung wird in einer speziell gesicherten Umgebung nur von vertrauenswürdigem, speziell unterwiesenem Personal durchgeführt.

Kryptografische Algorithmen

Die Wahl der asymmetrischen Kryptografie ist in dieser Architektur mehreren Einschränkungen unterlegen. Die meisten Smartcards – insbesondere die kontaktlosen und hybriden – unterstützen RSA nur bis 2048 bit. Das Bundesamt für Sicherheit in der Informationstechnik (BSI) empfiehlt diese Schlüssellänge nur noch bis 2022. Danach sollte RSA mit mehr als 3000 bit genutzt werden. Das National Institute of Standards and Technology (NIST) empfiehlt für den aktuellen Zeitraum 2016–2030 auch 3072 bit. Zudem benötigt die Generierung eines 2048-bit-RSA-Schlüssels auf der Smartcard im Durchschnitt drei Sekunden bei einer Stromversorgung über NFC.

Elliptische Kurven sind somit eine sinnvolle Alternative. Selbst kosteneffektive Smartcards unterstützen oft Schlüsselgenerierung und Signaturerstellung mit ECDSA. Da hier nur eine Challenge signiert werden muss und keine Entschlüsselung stattfindet, benötigt es kein ECDH, was von vielen Smartcards nicht unterstützt wird. Eine aktuell von BSI und NIST empfohlene Schlüssellänge ist 256 bit. Für höhere Sicherheitsniveaus, wie z. B. in der Commercial National Security Algorithm (CNSA) Suite, wird 384 bit empfohlen.

Nicht gewählt wurden die vom BSI empfohlenen Brainpool-Kurven oder modernere Kurven und Signaturalgorithmen wie Ed25519. Brainpool-Kurven sind aktuell nicht für OpenSSH User Certificates spezifiziert. Ed25519 wird aktuell von den meisten Karten noch nicht unterstützt. Dies ist aber eine sinnvolle Wahl, sobald JavaCard 3.1 eine höhere Verbreitung gefunden hat und kosteneffektive Smartcards zur Verfügung stehen. Zusammenfassend fällt die Wahl somit auf die Kurve NIST P-384 unter Nutzung von ECDSA. Als Hash-Algorithmus wird SHA-256 verwendet.

Smartcards und Tablets

Ist das Tablet mit dem IIoT-Gerät verbunden, wird der Nutzer über eine passende Konfigurations-App informiert. Nach deren Installation kann der Nutzer sich gegenüber dem Gerät authentifizieren, indem er die PIN eingibt und die Smartcard für NFC-Kommunikation gegen die Rückseite des Gerätes hält. Die Smartcard ermöglicht somit ein portables System, d. h. es müssen keine Konfigurationen oder Schlüssel auf ein neues Tablet kopiert werden.

Die Smartcard-Kommunikation findet mit einem eigenen SDK statt. Es implementiert und abstrahiert die Smartcard-Protokolle und detektiert Smartcards automatisch über NFC. Ein User Interface bietet PIN-Abfrage und Hilfestellung zur Positionierung der Smartcard. Im Hintergrund wird das User Certificate von der Smartcard abgerufen und beim SSH-Challenge-Response-Verfahren als „pubkey“ genutzt. Die SSH-Server-Challenge wird über USB empfangen und per NFC an die Smartcard weitergegeben. Auf der Smartcard findet die ECDSA-Signaturerstellung statt. Die signierte Challenge bietet somit den Beweis, dass sich zu dem gesendeten User Certificate der Private Key im Besitz des Nutzers befindet.

Nutzlose Angriffe

Die Sicherheit gegenüber Angreifern im Netzwerk, die sich nicht physisch Zutritt verschafft haben, ist sehr hoch. Durch die Nutzung einer Public-Key-Infrastruktur zusammen mit SSH User Certificates gibt es praktisch keine bekannte Möglichkeit, Gerätezugriff durch Brute-Force-Angriffe zu erlangen. Durch die Nutzung von externen Smartcards wird außerdem ausgeschlossen, dass Private Keys durch Malware auf dem Tablet gestohlen werden können.

Ein Angreifer, der sich in physikalischer Nähe befindet, könnte versuchen die NFC-Kommunikation abzuhören. Das einzige sinnvolle Datum, das abgegriffen werden könnte, wäre die signierte Challenge. Zum einen ist das Abhören von NFC durch die Beschaffenheit der NFC-Antenne nur bis auf wenige Meter möglich. Zum anderen passt diese signierte Challenge nur zu der aktuellen Verbindung zwischen Gerät und Tablet. Einen erweiterten Schutz bietet eine NFC-Verschlüsselung, die mit einem Key Agreement initiiert wird.

Fazit

Eine benutzerfreundliche und dennoch sichere Konfiguration von IIoT-Geräten ist möglich: Servicetechniker benötigen vor Ort nur ein handelsübliches Smartphone oder Tablet und ihre persönliche Smartcard. Da die Smartcard SSH User Certificate und Private Key speichert, ist das Verfahren portabel und unabhängig vom Mobilgerät. Dennoch erreicht das Verfahren ein hohes Sicherheitsniveau, das sich auf Empfehlungen des BSI, NIST und verschiedenen Standards wie z. B. IEC 62443 stützt.

Dr. Jó Bitsch
exceet Secure Solutions GmbH
Dr. Dominik Schürmann
Cotech – Confidential Technologies GmbH