WHITEPAPER

SICHERHEITSKONZEPT FÜR SICK REMOTE SERVICE

LIFETIME SERVICES, 2013-03

HERAUSGEBER:

SICK AG in Waldkirch / Deutschland

Lifetime Services,
SICK AG in Waldkirch / Deutschland

INHALTSVERZEICHNIS

SICK
Sensor Intelligence.

Einleitung

Die SICK Meeting Point Architektur (MPA) wurde mit dem Ziel maximaler Sicherheit bei minimaler Bedienkomplexität entwickelt. Die Architektur orientiert sich an den Empfehlungen und dem Maßnahmenkatalog des BSI (Bundesamt für Sicherheit in der Informationstechnik) zur Absicherung von Fernwartung (M5.33).

Dazu gehören:
- Fernwartungszugriffe immer nur vom lokalen IT-System initiieren
- Durchführung der Fernwartung ausreichend protokollieren
- Einhaltung des 4-Augen-Prinzips (keine Fernwartung ohne die Freigabe durch den Kunden)

Wie diese Vorgaben technisch in der Meeting Point-Architektur eingeflossen sind und welche Details eine sichere Fernwartung gewährleisten wird im Folgenden erläutert.

Eingesetzte Protokolle

Es werden zwei Kommunikationskanäle in der **MPA** verwendet:
- HTTPS (Port 443)
- SSH (Port 22)

HTTPS
HTTPS wird allgemein zur verschlüsselten Kommunikation zwischen Browser und Webserver u.a. beim Onlinebanking, bei diversen Einkaufsportalen und bei vielen Portalen, die personenbezogene Daten speichern, eingesetzt. Der Webserver setzt dabei ein x.509-Zertifikat ein, das dem Benutzer die Sicherheit gibt, dass er mit dem gewünschten Server verbunden ist. Dazu wird in dem Zertifikat der Name des Servers, unter dem er im Internet erreichbar ist, hinterlegt und diese Information von einer Certification Authority (CA) unterschrieben. Ein Browser liefert eine Liste von vertrauenswürdigen CAs mit, von denen man erwartet, dass sie nur dann unterschreiben, wenn die Information zum Servernamen korrekt ist.

Die Vergangenheit hat jedoch gezeigt, dass dies nicht immer gewährleistet ist. Etwa durch Lücken in den von den CAs genutzen Webportalen zur Verwaltung von Zertifikaten, die es erlauben, beliebige Servernamen in ein Zertifikat zu speichern. Oder durch regulär arbeitende und in den Browsern gelistete CAs, die ohne große Überprüfungen nahezu jeden gewünschten Servernamen in ein Zertifikat hineinschreiben. Ein **MPA**-Server arbeitet daher mit einer eigenen CA, d.h. der Server ist in der Lage, selbst Zertifikate auszustellen. Dadurch ist es möglich, nicht nur den Webserver mit einem Zertifikat auszustatten, sondern auch einzelne Benutzer können ein x.509-Zertifikat erhalten und sich damit gegenüber dem Webserver auf sichere Weise ausweisen. Diese Benutzerzertifikate werden in dem Kundenbrowser installiert und von diesem bei jedem Besuch des **MPA**-Servers präsentiert. Sichert man den Zertifikatsspeicher im Browser wiederum mit einem Passwort, erhält man eine Zwei-Faktor-Authentifizierung: Man benötigt das Zertifikat und das Passwort, um sich an dem Server anmelden zu können. Insbesondere bei Rechnern, die zur Maschinensteuerung verwendet werden und an denen vielfach mehrere Mitarbeiter im Wechsel arbeiten, ist die Verwendung von Benutzerzertifikaten eine bequeme und sichere Möglichkeit der Benutzerauthentifizierung.

Wird in einem Browser ein HTTPS-Webserver angesprochen, der ein Zertifikat vorweist, das nicht von einer der hinterlegten vertrauenswürdigen CAs unterschrieben ist, so werden in der Regel Warnmeldungen ausgegeben. Wird das CA-Zertifikat des **MPA**-Systems einmal in die Liste der vertrauenswürdigen CAs aufgenommen, so akzeptiert der Browser das Server-Zertifikat fortan wie alle anderen Zertifikate von vorinstallierten CAs. Prinzipiell ist die Akzeptanz von Zertifikaten eine Vertrauenssache.

Aufgrund der Vielzahl vorinstallierter Zertifikate im Browser und der damit undurchsichtigen Vertrauenskette vom CA-Betreiber über den Browserhersteller und die Softwareverteilung bis zu Manipulationsmöglichkeiten nach der Installation empfiehlt es sich, die mitgelieferte Liste vertrauenswürdiger CA-Zertifikate zu löschen. Vertrauen wird dann nur individuell einzelnen Certification Authorities durch die Aufnahme in die CA-Liste entgegengebracht, die wirklich für die vom Benutzer angesprochenen Seiten nötig sind. Die Vergangenheit hat etwa mit dem Beispiel DigiTrust gezeigt, dass offiziell gelistete Certification Authorities kompromittiert werden können und damit beliebige Zertifikate herausgeben, die dann „Man-in-the-Middle"- Angriffe sehr viel einfacher machen. Individuelle Aufnahme von CAs in die Liste der vertrauenswürdigen CAs reduziert das Risiko für erfolgreiche Angriffe dieser Art erheblich. HTTPS verwendet beim Einsatz aktueller Browser den Verschlüsselungsalgorithmuns AES mit einer Schlüssellänge von 256 Bit. Im **MPA**-Server wird der weit verbreitete Webserver Apache als Kommunikationsendpunkt eingesetzt. Zur Verschlüsselung werden dabei die OpenSSL-Bibliotheken verwendet.

SSH
Secure Shell (SSH) ist ein weit verbreitetes Protokoll zum sicheren Zugang zu Unix-Servern über das Netz. Die serverseitig eingesetzte OpenSSH-Implementierung stammt aus der Entwicklungsumfeld von OpenBSD, einem sehr auf Sicherheit bedachten Unix-Derivat, das z.B. in vielen Hardware-Firewalls Einsatz findet. Neben OpenSSH wird in der Meeting Point Architektur auf Windows-Systemen auch Plink aus dem Putty-Paket eingesetzt. Das SSH-Protokoll erlaubt neben der Authentifizierung per Benutzername/Passwort auch den Einsatz von Public-Keys, die vorab ausgetauscht und dem Server beim Verbindungsaufbau präsentiert werden, und weitere wie etwa Kerberos.

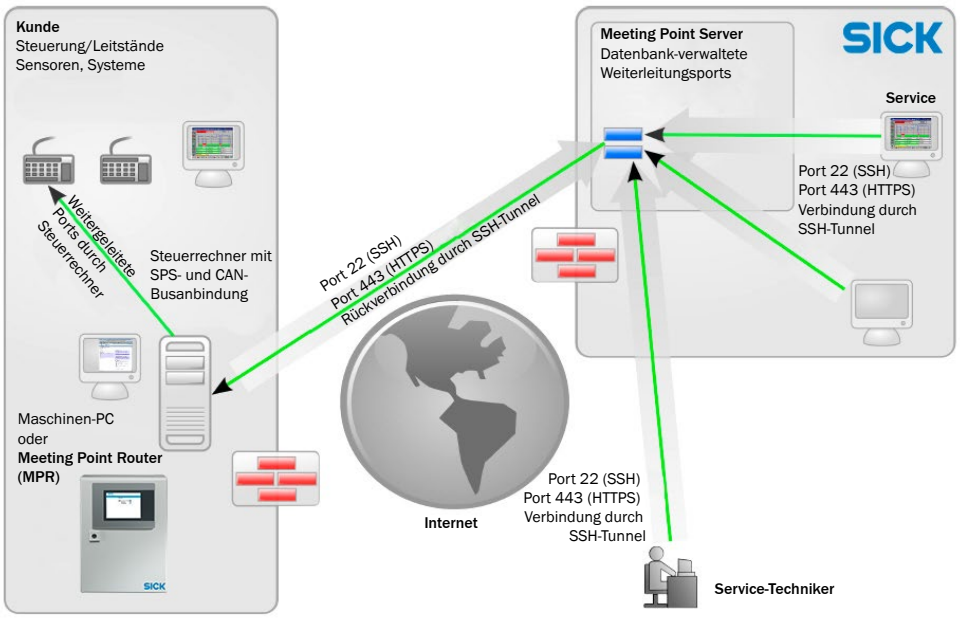
Der **MPA**-Server verwendet ausschließlich die Public-Key Authentifizierung. Die Verwendung von Benutzername/Passwort ist deaktiviert, da es im Internet Automaten gibt, die diese automatisiert durchtesten und damit eine Restwahrscheinlichkeit bestünde, zufällig ein schwaches Passwort anzutreffen. Die Public-Keys werden vom System vor dem SSH-Verbindungsaufbau dynamisch generiert und per HTTPS ausgetauscht. Die Gültigkeit eines so erzeugten Schlüsselpaares ist auf max. 10 Minuten begrenzt. Nach der Authentifizierung wird im SSH-Protokoll ein Sitzungsschlüssel erzeugt, der für den weiteren Datenaustausch verwendet und periodisch gewechselt wird. Die Verschlüsselsung wird durch AES mit einer Schlüssellänge von 128 Bit realisiert.

Benutzer-Authentifizierung

Die Benutzer-Authentifizierung erfolgt über clientseitige x.509- Zertifikate, die im Browser hinterlegt werden können. Das Verfahren führt dazu, dass dem Benutzer ein zufälliger Sessioncookie übergeben wird, der im Weiteren für die Identifikation des Benutzers verwendet wird. Dieser Cookie wird in regelmäßigen Abständen ausgetauscht, damit im Falle eines „Man-in-the-Middle"-Angriffs nur ein begrenztes Zeitfenster für das Ausnutzen der gewonnenen Information bleibt. Um Brute-Force Angriffe auf Benutzername/Passwort-Kombinationen abzuwehren, werden Accounts nach 5 Fehlversuchen automatisch gesperrt und müssen durch einen Administrator wieder freigeschaltet werden. Fehlversuche werden unabhängig von der Applikation im System geloggt und können durch den System- Administrator analysiert werden.

Verbindungs-Struktur

MPA setzt auf eine sternförmige Verbindungsstruktur, d.h. alle Verbindungen werden über den zentralen MPA-Server aufgebaut. Nur dieser Server stellt die o.g. Dienste HTTPS und SSH im Internet bereit, alle anderen Komponenten nutzten diese und verwenden daher ausschließlich ausgehende Verbindungen. Insbesondere gibt es keine Direktverbindungen zwischen Servicetechnikern und Maschinen. Vorgänge wie der Verlust von Laptops, die Veränderung von Mitarbeitern, die Berechtigungsänderungen nach sich ziehen oder aufgedeckte Sicherheitslücken lassen sich zentral behandeln. Die Notwendigkeit, an Clientgeräten administrativ tätig zu werden, entfällt mit dieser Architektur.



Vorteile

SSH-Authentifizierung durch dynamische Einmalschlüssel

HTTPS-Authentifizierung durch x509-Zertifikate

DB-basierte Verwaltung

Kaum Anforderungen an Router und Firewall

Bild 1: Meeting-Point Architektur

Maschinen-PC

Wenn Steuer-PCs an Maschinen als Bedienterminal für **MPA** verwendet werden, ist dafür der Accounttyp „Maschinenaccount“ vorgesehen. Dieser bietet Zugang zu genau einer Maschinenakte und erlaubt den Aufbau einer Remoteverbindung. Die im **MPA**-Server generierte SSH-Verbindungskonfiguration erlaubt für Maschinenaccounts lediglich SSH-Reversetunnel, d.h. es können damit Ressourcen auf dem Maschinenrechner angesprochen werden, nicht jedoch in die Gegenrichtung.

Meeting Point Router MPR

Der MPR basiert auf einem einfachen Industrie-PC, der mit zwei Netzwerkschnittstellen ausgestattet ist und zur Trennung von Maschinennetz und Kundennetz dient. Durch eine integrierte Paketfirewall werden gezielt nur die benötigten Kommunikationswege zwischen dem Kundennetz und dem Maschinennetz, aber auch von dem Maschinennetz in das Internet freigeschaltet. Für den Remote-Zugang über den **MPA**-Server werden auch hier nur ausgehende Verbindungen per HTTPS und SSH eingesetzt. Zur Authentifizierung wird beim Rollout des MPR eine eindeutige Kennung auf dem System hinterlegt, die beim Ansprechen des MPR mitgesendet wird. Diese Kennung ersetzt die Verwendung von Benutzername und Passwort für den MPR und muss daher geheim gehalten werden. Für den Fall, dass dennoch einmal eine MPR-Kennung aus dem MPR entwendet wurde, kann damit ein Zugriff auf die MPR-spezifischen Seiten des **MPA**-Servers erfolgen. Diese erlauben es, eine Remote- Verbindung aufzubauen, die jedoch ausschließlich Reverse-Ports umfassen kann (durch serverseitige Einschränkung sichergestellt), so dass ein potentieller Angreifer lediglich in der Lage wäre, eigene Ressourcen zugänglich zu machen, insbesondere aber keinen Zugriff auf Ressourcen anderer gerade verbundener MPR-Geräte erlangen kann.

Bild 2: Meeting Point Router: Sicherheit durch Trennung von Maschinennetz und Kundenentz

Neben den normalen per SSH weitergeleiteten Ports bietet der MPR die Möglichkeit, bei Bedarf auch eine vollständige VPN-Verbindung zwischen Servicetechniker und Maschinennetz aufzubauen. Dazu wird durch einen SSH-Tunnel einer Remote-Verbindung zusätzlich mit OpenVPN eine Netzwerkkopplung auf Layer 2 aufgebaut, so dass auch etwa Broadcast- oder UDP-basierte Dienste vom Servicetechniker genutzt werden können. Die OpenVPN-Option ist ausschließlich durch eine bestehende SSH-Remoteverbindung nutzbar, so dass die Authentifizierung und Verschlüsselung der VPN-Verbindung keinen neuen Angriffspunkt darstellt.

Servicetechniker
Servicetechniker haben Zugang zu allen Maschinenakten und können sich sich zu allen gerade per Remote-Verbindung verbundenen Maschinen verbinden. Eine Verbindung durch einen Servicetechniker wird stets in der Maschinenakte nachrichtlich vermerkt. Die serverseitige Generierung der SSH-Konfiguration erlaubt in diesem Falle ausschließlich Forward-Tunnel, d.h. ein Servicetechniker kann Dienste auf Seiten des Kunden ansprechen, nicht jedoch in der Gegenrichtung. Die Einschränkung der Port-Weiterleitungen auf die für die Maschine vorgesehenen Ports erfolgt zweistufig: Zum einen wird die SSH-Clientkonfiguration mit diesen Ports versehen, parallel dazu wird für den betreffenden SSH-Schlüssel auf dem **MPA**-Server serverseitig eine Liste erlaubter Ports hinterlegt, die durch den SSH-Server beim Verbindungsaufbau geprüft wird.

Potenzielle Angriffs-Szenarien

Durch die Beschränkung auf genau zwei Verbindungsmechanismen ist auch die Angriffsmöglichkeit von Außen auf diese beschränkt. Da lediglich der **MPA**-Server aktiv für Verbindungen auf diesen Ports ist, reduziert sich die Notwendigkeit regelmäßiger Sicherheitsupdates weitestgehend auf diesen Server.

Angriffe auf die Verbindungstechnik
SSH wird in einer maximal gesicherten Konfiguration betrieben (Zugang nur für die benötigten Benutzer möglich, ausschließlich per Public-Key-Authentifizierung, kein Root-Zugang, ausschließlich SSH2-Protokoll), was weltweit in vielfacher Weise für administrative Fernzugänge so genutzt wird. Erfolgreiche Angriffe auf so konfigurierte Server sind nicht bekannt. Zusätzlich verwenden wir dynamisch generierte, nur zeitlich eingeschränkt gültige Schlüsselpaare, so dass auch Brute-Force Angriffe auf die Schlüssel nicht erfolgversprechend sind.

Der HTTPS-Server bietet eine End-To-End-Verschlüsselung zwischen Browser und Webserver, die mit üblichen Mitteln nicht aufzubrechen ist. Allerdings besteht wie bei jeder HTTPS-Kommunikation die Möglichkeit, mit einer „Man-in-the-Middle“-Attacke die Kommunikation zwischen Browser und Server aufzubrechen. Die dazu nötigen Manipulationen erfordern jedoch stets einen Eingriff in die grundlegende IT-Infrastruktur wie DNS-Server oder Proxy-Infrastruktur. Daneben muss bei einem solchen Angriff entweder die Liste der vertrauenwürdigen Zertifizierungsstellen im Browser oder aber eine vertrauenswürde Zertifizierungsstelle direkt manipuliert werden. Im Interesse der IT-Sicherheit im Unternehmen sollte dies jedoch ohnehin administrativ verhindert werden. Einige Firmen nutzen diese Möglichkeit jedoch vermeintlich im Interesse der eigenen IT-Sicherheit aus (was relativ einfach ist, da alle o.g. Komponenten durch die eigene IT beeinflussbar sind), um per HTTPS verschlüsselten Datenaustausch auf Schadsoftware zu untersuchen. So gut dies gemeint ist, so nachhaltig durchbricht es jedoch das Konzept der Ende-zu-Ende-Verschlüsselung und unterwandert damit das Vertrauen der Anwender, die von einer sicheren Kommunikation über HTTPS ausgehen.

Angriffe auf die Applikation
Neben Angriffen auf die Verbindungstechnik verbleibt die Möglichkeit des Angriffs der Webapplikation selbst. Die Softwarekomponenten und die Implementierung der Web-Applikation auf dem **MPA**-Server sollten dazu gut gewartet und regelmäßig auf bekannte Sicherheitslücken hin überprüft werden. Mit einem Service Level Agreement stellen Sie sicher, dass in diesem Falle schnell und fachkundig reagiert wird. Die interne Sicherheitsarchitektur der Webapplikation teilt die einzelnen Bereiche strikt gegeneinander ab. Bereiche, die benutzerabhängig Zugang zu Nutzdaten gewähren, sind durch Gatekeeper geschützt, die bei Aufruf eines entsprechenden Pfades zuerst die übergebenen Parameter auf Gültigkeit im Kontext des Benutzers überprüfen. Hierzu gehören u.a. Maschinenakten, Dokumente und Verbindungen.

Risikoabschätzung

Der zentrale Aufbau der Meetingpoint Architektur konzentriert die Angriffsmöglichkeiten auf einen Server sowie die Verbindungen zu diesem. Im Vergleich zu vielfach eingesetzten dezentralen VPN- oder Einwahl-Architekturen kann etwa bei verloren gegangenen Laptops schnell zentral reagiert und der Zugang für den Benutzer gesperrt werden. Insbesondere die Vermeidung direkter Kommunikation zwischen Servicetechniker und Kunden und die dazu nötige Verteilung von Zugangsdaten an die Servicetechniker verringert die denkbaren Angriffsszenarien erheblich. Die genannten Angriffsmöglichkeiten auf die HTTPS-Verbindungen würden es erlauben, dem Anwender manipulierte Seiten zu präsentieren oder eingegebene Information abzugreifen. Es ist jedoch nicht möglich, dadurch den Server über die eigene Identität zu täuschen und damit Information abzugreifen, die nicht für den Benutzer bestimmt ist. Durch das Verfahren, Benutzeraccounts nach 5 Fehlversuchen zu sperren, ist ein Denial-of-Service Angriff denkbar, der versucht, möglichst viele Accounts zu sperren. In diesem Fall muss der Administrator diese Accounts wieder freischalten und sollte bei Häufung weitere Maßnahmen (etwa Firewall-Sperren für die Quelle des Angriffs) ergreifen.

Fazit
Die verbleibenden Risiken sind beherrschbar. Die denkbaren Angriffsszenarien konzentrieren sich auf wenige Punkte, die sich aufgrund des zentralen Aufbaus gut überwachen lassen. Umfangreiches Systemlogging stellt sicher, dass Angriffe protokolliert werden und bei den regelmäßigen Systemdurchsichten im Rahmen des Service Level Agreements nicht unbemerkt bleiben.

8016207/2013-03 · DIV03 (15.04.13) · WB USmod de37

REFERENCES

Maßnahmenkatalog des BSI (Bundesamt für Sicherheit in der Informationstechnik) zur Absicherung von Fernwartung (M5.33)

SICK
Sensor Intelligence.