Er du hackersikret?

Af Flemming Jensen, Adm. Dir.
Nohau Danmark A/S

Du kan næsten ikke åbne et teknisk tidskrift eller en nyhedsmail indenfor IT, uden at læse om, at det og det produkt er blevet hacket. Indenfor den sidste uge - her i slutningen af august - er det gået hårdt ud over bilindustrien, hvor både Tesla og Chrysler har været i medierne med historier om, at de er blevet hacket enten bevidst som et forsøg fra producenten selv eller som et såkaldt fjendtlig attack.

Men det er jo bestemt ikke kun bilindustrien, der har dette problem. Man behøver kun at google "IoT hacking" eller "Embedded device hacking" eller lignende ord for at se, at hackerne nu også går efter embeddede devices og ikke længere kun store administrative datasystemer. På dette års DefCon, som er den største "hacker konference" i USA, vil der være en speciel "IoT Village" hvor man vil have fokus på IoT devices.

IoT (Internet of Things) er jo rigtig meget oppe i tiden og en netop offentliggjort undersøgelse viser, at hvis man lægger en IoT device ud i Cyberspace, vil den indenfor 18 timer have haft det første angreb og indenfor en lille måneds tid, vil den være angrebet 30 – 40 gange fra et utal af forskellige lande.

Prøv og tænk på de nye "SmartCities", hvor man måske har kablet alle lyskurve i byen sammen. Hvis man overtog styringen og slukkede for dem alle samtidig med gadelyset. Konsekvenserne af den slags handlinger er vanskelige at forholde sig til, og der er helt sikkert scenarier, der har væsentlig værre konsekvenser.

Hvad gør vi så ved det? Vi må erkende, at fremtiden kun vil betyde flere angreb. Selv om langt de fleste vil være ganske harmløse, så kan konsekvenserne af et egentligt fjendtligt angreb være ganske katastrofal.

Der er naturligvis flere måder at forsøge at undgå denne type af udfordringer.

Man kan skriver sin software "rigtig", så risikoen mindskes

Det er derfor nok en god ide at følge egne eller generelle kodningsstandarder (MISRA, HIC++, JSF ...). Det er dog ikke altid let at sikre, at standarderne overholdes, når der er mange, der koder på samme projekt, og man anvender 3. parts komponenter osv. Der findes dog forskellige værktøjer, der kan gøre dette arbejde mere effektivt end det er muligt med opslidende og tidskrævende review-møder.

Ved brug af statisk kodeanalyse værktøjer kan man sikre, at man følger de valgte standarder, ligesom det giver mulighed for, at analysere applikationerne for sårbarheder og brug af ikke valideret data.

Links:

*1 Study
*2 MISRA
*3 GrammaTech
*4 Lattix
*5 Sysgo PikeOS
*6 IconLabs FloodGate Firewall

Perfekt kod

Når man så har gjort, hvad man kan, for at koden er fornuftig skrevet og "fejlfri", så kan arkitekturen stadig være så kompleks, at afhængigheder mellem komponenterne bliver uigennemskuelige, at de giver anledning til skjulte sårbarheder – især når der skal ændres i koden over tid. Der findes naturligvis også værktøjer på markedet, der kan bruges til at analysere og optimere arkitekturen.

En anden måde at opnå en mindre sårbar arkitektur, er ved at vælge et "sikkert" og præcertificeret operativsystem som f.eks. PikeOS, hvor man kan isolere de kritiske Software komponenter fra de mere generelle ved hjælp af virtualisering, som vist i fig. 1.

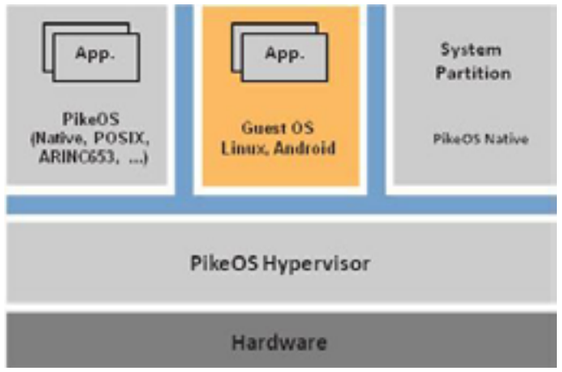
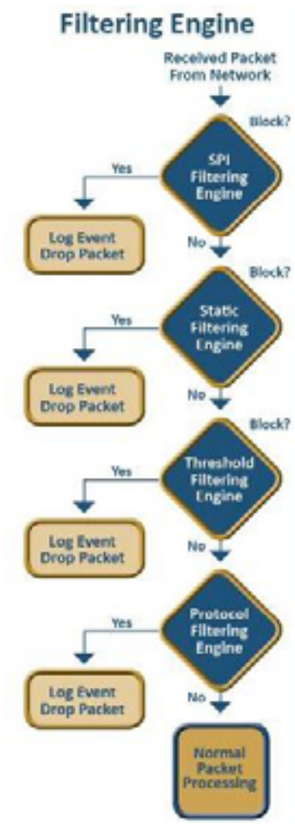


Fig. 1
Opdeling, som det bør gøres, hvis man skal overholde ARINC 653 standarden. Embedded virtualisering tillader opdeling af de forskellige applikationer og deres respektive RTOS, API og RTE i hver deres partitions.

De enkelte devices kan naturligvis også beskyttes med en embedded firewall. Den slags findes sikkert fra flere leverandører, men her vil jeg beskrive Iconlabs FloodGate™ Firewall som har vundet adskillige priser på det amerikanske marked. Floodgate™ Firewall er en komplet embedded firewall, som tilbyder flere forskellige niveauer af sikkerhed for et IoT device. Dens unikke design tilbyder flere typer af filter beskyttelse mod internetbasserede trusler. Nærmere beskrevet i fig. 2.



- Static/rules-based filtering blocks packets based on configurable rules.

- Dynamic filtering/stateful packet inspection (SPI) blocks packets based on connection state.

- Deep packet inspection for industrial automation protocols allowing control and validation of each individual field within the message, and filtering of messages based on message type, contents and message source.

- Threshold-based filtering blocks packets based on threshold crossings to protect against denial of service (DoS) attacks, broadcast storms and other packet flood conditions.

Fysiske industrielle firewalls

Der findes naturligvis også fysiske industrielle firewalls på markedet, der i lighed med de Firewalls, der anvendes til vores normale IT-netværk, kan beskytte de enheder, der befinder sig bag ved firewallen. Forskellen på de industrielle firewalls og almindelige IT-firewalls er, at de industrielle f.eks. kan anvendes til forskelige typer af SCADA netværk og de – ved fjernkonfiguration – let kan opdele og segmentere eller isolere et system efter behov.

Det sidste vigtige krav, i forhold til at lave sikker software og IoT device, er validering og test af sikkerhedsniveauet. Skal en applikation, device eller et system certificeres eller blot have et dokumenteret sikkerhedsniveau, så kræver det ofte, at man søger "ekspert hjælp" da det "Set-Up", der skal til for at gennemføre en sådan test, vil være alt for omfangsrigt for almindelige virksomheder. Der gennemføres en "blackbox-test" hvor enheden bliver bombaderet med alle former for påvirkninger, samtidig med, at man validerer, at den opfører sig som ønsket.

Følges disse retningslinjer giver det en øget sikkerhed, når man vil sende sine enheder eller systemer ud i dét "Cyberspace", som vi på mange måder er utrolig lykkelige for, og på ingen måde ønsker at slippe af med:

- Skriv koden "rigtigt" ved at følge nogle guidelines.
- Tjek ved hjælp af statisk kodeanalyse, at disse regler overholdes, samtidig med andre fejl og sårbarheder findes.
- Opbyg en overskuelig arkitektur med styr på afhængighederne, så fremtidig vedligeholdelse lettes.
- Isolér de kritiske softwarekomponenter fra "house keeping" kode.
- Benyt embeddede firewalls eller eventuelt dedikerede industrielle firewalls til perimetrisk beskyttelse af systemer.
- Test robustheden.

Så konklusionen må vel være at hvis man bruger fornuften, de rigtige værktøjer og lidt sund skepsis, så skal det nok gå.

www.nohau.se