SMARTGRID SECURITY CHALLENGES AND BEST PRACTICES

Bruno Rohée
Security Architect

CoESSI

AGENDA

- Smartgrid overview
- Hardware challenges and best practices
- Key management challenges and best practices
- Patch management and vulnerability mitigation for connected object

SMARTGRID

Property supply management network using IT to adapt and to optimize networks management.

The three main functions: dematerializing service provision, hypervision and maintenance of networks, metering infrastructure management

All actors benefit from it: smoothing usages means better stability, demand can match offer better, less spare capacity is needed.

SMARTGRID

Lots of sensors/actuator

- Flood detector
- Environmental sensors
- Voltage sensors
- Trip devices
- Smart meters

SMARTGRID SECURITY CHALLENGES AND BEST PRACTICES

- Easy for an attacker to steal one of your objects
  - **Assume some will be stolen and reverse engineered**
    - Make reverse engineering harder
    - Make key extraction hard
    - Reduce impact of key extraction
- Cost of installation is a big component of total cost
  - **Everything must be done to reduce the need for manual intervention**
- General immaturity on security subject in the industry
  - **New subject, safety is well known, security is different**

FIGHTING REVERSE ENGINEERING

- Tamper evident enclosure
- Alarm on enclosure opening
- Disabling of any debugging interface on production hardware
  - JTAG disabled at end of manufacturing for hardware dedicated to production use
  - Serial console disabled
  - Any other debug interface
- Unmarked IC
  - Makes chips harder to identify : longer => more expensive
- Prefer integrated SoC, preferably BGA
  - RAM snooping is real
- Encrypt firmware in flash

MAKING KEY EXTRACTION HARD

- Secure Element, TPM chips, software based TPM
  - If budget permits
  - Every high end ARM MCU includes a TrustZone

REDUCE IMPACT OF KEY EXTRACTION

- *Assume all secrets on a device can be compromised*
  - **Such compromise should not give any edge to an attacker**
    - **⇒ Diversify all symmetric keys**

IMMATURITY

- This is a new domain, security culture not yet ingrained
  - Connected objects and their challenges are new
  - Manufacturers have little if any culture, many pain points
    - Cryptography : mainly RNG, key management
    - Vulnerability management
    - Devices attack surface
      - Manufacturing may need to be able to securely handle secrets
    - Network isolation, hardening
- Make security contractual
  - Audit facilities and process, right to audit part of the contract
  - Ask for formalized security policies
  - Identified people and channels to handle security matters
- Third party evaluations (CSPN, Common Criteria…)

KEY MANAGEMENT

- Two main categories of connected objects, two set of constrains
  - Low power
    - Low speed, energy and bandwidth constraints
    - Symmetric cryptography only (AES, ChaCha20…) using either AEAD or MAC for authentication
  - High power, fast connection
    - Decent speed, reliable energy source, sometime even good bandwidth
    - ⇒ Asymmetric cryptography possible (RSA, Elliptic Curves)
    - Standard protocols (IPsec, TLS…) a possibility

SYMMETRIC KEY MANAGEMENT

- One object, a set of keys (e.g. SCP03 ‘s ENC, DEK, MAC)
  - Bad strategy : randomly generated and stored
  - Good strategy : diversification, using KDF(device_id|key_version, master_key)

    From needing to protects millions of keys, to needing to protect only a few master keys ⇒ can use a HSM
- Key rolling issues
  - Need to rotate keys after they have been disclosed
  - Often no assurance that the key roll is effective
  - Need to identify keys
- Key distribution woes
  - You may need

PKI: ASYMMETRIC KEY MANAGEMENT

- One object, one certificate
  - When to enroll?
    - At manufacturing time ; needs automated data exchange between manufacturer and operator
      - Device is ready to use, possibly headless installation
      - What about expiring certificates in stored devices?
    - When first installing the device
      - Needs access to a registration authority at installation time
      - What if network is unavailable?

Each strategy has its challenges, there is a need for auxiliary processes to handle corner cases either way.

PATCH MANAGEMENT AND VULNERABILITY MITIGATION FOR CONNECTED OBJECTS

- Firmware updates are slow and failure is expensive
  - Limited bandwidth, shared with data gathering
  - Failure needs manual intervention, possibly remotely
- ⇒ Park managers are understandably squeamish

- But cost of security incident is usually underestimated
  - Need to be able to prove that security fixes won't break anything when deployed
    - Not that hard when the vulnerability is in some code you don't actually use, case for e.g. Heartbleed
    - Which begs the question, why did you ship this code in the first place
- A solid risk analysis (ISO27005, EBIOS…) with impact analysis helps make your case

CoESSI

PATCH MANAGEMENT BEST PRACTICES

- Know your dependencies
- Do watch for new vulnerabilities
- Need to have a dedicated, accelerated, validation procedure for security fixes
  - And you need to be able to make your case to other stakeholders
- Work upfront to lower the need to ship fixes, embark minimal, hardened programs. No one needs OpenSSL in a thermometer!.
- Audit
- Do impact analysis on new vulnerabilities to assess exposure
  - Skilled people able to do that are rare

PATCH MANAGEMENT BEST PRACTICES

- Know your dependencies
- Do watch for new vulnerabilities
- Need to have a dedicated, accelerated, validation procedure for security fixes
  - And you need to be able to make your case to other stakeholders
- Work upfront to lower the need to ship fixes, embark minimal, hardened programs. No one needs OpenSSL in a thermometer!
- Prefer LTS branches
- Audit the software and the configuration
- Do impact analysis on new vulnerabilities to assess exposure
  - Skilled people able to do that are rare

Thank you!

Questions?

herve.daussin@coessi.fr
bruno.rohee@coessi.fr

www.coessi.com

Twitter.com/coessi_fr

Linkedin/company/coessi