Security of cyber-physical systems: an old idea
Security Issues and Mitigation in Ethernet POWERLINK

Jonathan Yung, Hervé Debar and Louis Granboulan

AIRBUS Group Innovations & Télécom SudParis

February 10th, 2017
Recurrent issues in cyber-security

- Name changes, keywords and buzzwords
  - Information systems security → cyber-security
  - Intrusion detection → anomaly detection
  - Big data
  - Artificial intelligence

- Really represent persistent issues in IT
  - Security as an afterthought
  - Security as an overkill
  - Security without expectations
  - Security without specifications
  - ...
Can we construct a better approach?

- Safety has clear pre-specified expectations.
  - Standardized constraints
- Industrial control systems have clear specifications.
  - Control loop
- Can we leverage these two old ideas for better security?
- *Old keyword: Resilience*
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Industrial Control System
System composed of devices producing data (sensors), and of devices which will act depending of this data (actuator) and of a program.
Industrial Control System

System composed of devices producing data (sensors), and of devices which will act depending of this data (actuator) and of a program.
Industrial Ethernet Protocols

- Adaptation of the Fieldbus protocols on Ethernet.
- Classified in three types:
  - Class 1 (soft real-time): MODBUS/TCP, EtherNet/IP
  - Class 2 (hard real-time): PROFINET (RT)
  - Class 3 (isochronous real-time): PROFINET IRT, Ethernet POWERLINK, EtherCAT
## Attack State of the Art

<table>
<thead>
<tr>
<th>Attack</th>
<th>MODBUS/TCP (C1)</th>
<th>EtherNet/IP (C1)</th>
<th>PROFINET RT (C2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eavesdropping</td>
<td>Huitsing et al.[5]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bristow[4]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Queiroz[7]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Paul et al.[6]</td>
</tr>
</tbody>
</table>
Goal of this presentation

- Literature already presents attacks and mitigation measures...
- ... but only for class 1 and/or 2 protocols.
- The goal of this presentation is to:
  - test the security of a type 3 protocol: Ethernet POWERLINK
  - propose security improvements
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Architecture

- It is specified by the EPSG (Ethernet POWERLINK Standardization Group).
- It uses the Master/Slave paradigm.
  - A Slave can send a message only if asked by the Master.
- It is composed of:
  - one master called Managing Node (MN)
  - up to 240 slaves called Controlled Node (CN)
- The master and slaves are connected through Hubs.
Cycle Structure

Composed of three periods:
- Isochronous period
- Asynchronous period
- Idle period

<table>
<thead>
<tr>
<th>Isochronous period</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoC</td>
</tr>
<tr>
<td>PReq CN 1  MN → all</td>
</tr>
<tr>
<td>PRes CN 1  CN 1 → all</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>PReq CN n  MN → CN n</td>
</tr>
<tr>
<td>PRes CN n  CN n → all</td>
</tr>
<tr>
<td>PRes MN    MN → all</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Asynchronous period</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoA</td>
</tr>
<tr>
<td>ASnd</td>
</tr>
<tr>
<td>MN or one CN → all</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Idle period</th>
</tr>
</thead>
</table>

Cycle
Network Management

- The Network Management (NMT) State Machine describes a device behavior.
- The master can change the NMT state of a slave through an ASnd command.
- A slave can ask the master to send an NMT command to change the NMT state of a slave or of the master.
- The NMT states of an master are (non exhaustive list):
  - init
  - not_active
  - pre_operational_1
  - pre_operational_2
  - ready_to_operate
  - operational

![NMT State Machine Diagram]

- Configuration is done
- Does not receive any SoC or SoA
- All mandatory CN are identified
- MN configuration OK
- All CN in ready_to_operate
- No error after a complete cycle
- Does not receive any SoC or SoA
The Network Management (NMT) State Machine describes a device behavior.

The master can change the NMT state of a slave through an ASnd command.

A slave can ask the master to send an NMT command to change the NMT state of a slave or of the master.

The NMT states of a slave are (non exhaustive list):

- init
- pre_operational_1
- pre_operational_2
- ready_to_operate
- operational
- stopped
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Attacker Model

- Our attacks are on the Ethernet POWERLINK protocol.
- We consider an attacker connected to the Ethernet POWERLINK network.
  - We need physical access.
  - We need a free port.
- Attacker positions
  - Most likely position: end of bus (free port)
  - Inserted after the master (requires interruption)
- Attacker capabilities
  - Listen to and analyze traffic
  - Sufficiently fast to impersonate any slave
  - With enough computational power for crypto operations
Attacks Description

- Denial of service

Send many SoCs
Attacks Description

- Denial of service
- Acyclic command insertion
 Attacks Description

- Denial of service
- Acyclic command insertion
- Slave impersonation

ASnd message insertion: stop slave

Wait for PReq

Send PRes

Wait for SoA

SoA for the CN?
- Yes: Send ASnd
- No:
Attacks Description

- Denial of service
- Acyclic command insertion
- Slave impersonation
- Master reset

```
CN impersonation

Send PRes with PR=0b111 and RS=0b001

Wait for the ASnd for this PRes

Send NMT request
```
Attacks Description

- Denial of service
- Acyclic command insertion
- slave impersonation
- master reset
- master impersonation

MN
reset
Init MN
state
machine
Testbed for the implementation of the attacks

- Composed of a master, a slave, and an attacker (attacker is a computer, with Linux 3.13.0-95-lowlatency kernel).
- Two testbeds implemented:
Testbed for the implementation of the attacks

- Composed of a master, a slave and an attacker (attacker is a computer, with Linux 3.13.0-95-lowlatency kernel).
- Two testbeds implemented:
  - one using B&R components for the master and slave
Testbed for the implementation of the attacks

- Composed of a master, a slave and an attacker (attacker is a computer, with Linux 3.13.0-95-lowlatency kernel).
- Two testbeds implemented:
  - one using B&R components for the master and slave
  - one using openPOWERLINK for the master on the same computer as the attacker, and B&R components for the slave
## Results

<table>
<thead>
<tr>
<th>Attacks</th>
<th>B&amp;R components</th>
<th>openPOWERLINK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denial of service</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>Acyclic command insertion</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>CN impersonation</td>
<td>~OK</td>
<td>OK</td>
</tr>
<tr>
<td>MN reset</td>
<td>OK</td>
<td>OK</td>
</tr>
<tr>
<td>MN impersonation</td>
<td>OK</td>
<td>OK</td>
</tr>
</tbody>
</table>
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Summing Up the Attacks

- The Master/Slave paradigm simplifies any DoS attacks.
  - we do not handle mitigation against DoS attacks here
- The other attacks are due to weaknesses in the asynchronous period:
  - no basic authentication of the command
  - no verification that the ASnd and SoA are consistent
  - several ASnd can be accepted by a slave
Wait SoC trigger

SoC trigger → send SoC+PReq

last PRes → send SoA, [ASnd]
last PRes timeout → send SoA, [ASnd]

PRes → send PReq
PRes timeout → send PReq

Wait PRes

Wait ASnd

PRes → send PReq

SoC trigger → send SoC + PReq

ASnd

PRes timeout → send PReq
Wait SoC
trigger

SoC trigger → send SoC+PReq

Wait PRes

PRes → send PReq
PRes timeout → send PReq

last PRes → send SoA, [ASnd]
last PRes timeout → send SoA, [ASnd]

Wait ASnd

SoC trigger → send SoC+PReq
First correct ASnd
Incorrect ASnd → error
Extra ASnd → error

PRes timeout → send PReq
DLL State Machine Modification - slave

Wait_SoC

Wait_PReq

Wait_SoA

SoC

SoA

PReq

PRes

SoA → [ASnd]

SoC timeout

PReq

PRes

SoA

ASnd

Wait_SoA

PReq → PRes

SoC

PRes

SoC timeout

ASnd

PRes

SoC

ASnd

PReq

PRes

SoC
DLL State Machine Modification - slave

First correct ASnd
PReq
PRes
SoA
SoC timeout

Wait_SoC

SoC timeout
PReq

SoA

SoA \rightarrow [ASnd]

Wait_SoA

PRes \rightarrow PRes

PRes

SoC

Wait_PReq

PRes

SoC

ASnd
DLL State Machine Modification - slave

Incorrect ASnd
Extra ASnd
First correct ASnd
PReq
PRes
SoA
SoC timeout

Wait_SoC

SoA
SoA → [ASnd]
SoC timeout

Wait_PReq

PReq
SoC timeout
SoA

Wait_SoA

PRes
ASnd

SoC

PReq→PRes
SoC

PRes
SoC
ASnd
NMT master State Machine Modification

- **init**
  - Configuration is done

- **not_active**
  - Does not receive any SoC or SoA

- **pre_operational_1**
  - Reset: All mandatory CN are identified

- **pre_operational_2**
  - MN configuration OK
  - All CN in ready_to_operate

- **ready_to_operate**
  - No error after a complete cycle

- **operational**
  - ASnd error

- **CN lost**
NMT master State Machine Modification

- **init**
  Configuration is done

- **not_active**
  Does not receive any SoC or SoA

- **pre_operational_1**
  Reset: All mandatory CN are identified

- **pre_operational_2**
  MN configuration OK
  All CN in ready_to_operate

- **ready_to_operate**
  No error after a complete cycle

- **operational**
  ASnd error
  Unexpected NMT command
NMT master State Machine Modification

- **init**
  - Configuration is done

- **not_active**
  - Does not receive any SoC or SoA

- **pre_operational_1**
  - All mandatory CN are identified and the MN is authenticated

- **pre_operational_2**
  - MN configuration OK
  - All CN in ready_to_operate

- **ready_to_operate**
  - No error after a complete cycle

- **operational**
  - Unexpected NMT command

- **CN lost**

- **ASnd error**

- **Reset**
Summary

- **Denial of Service:**
  - not handled here

- **Acyclic command insertion:**
  - The slave only accepts one correct command consistent with SoA.
  - It is not totally perfect: an attacker can be quick enough to send such a command before the master.
  - However, even in this case, it will be detected provided authentication (protection) behaves properly.

- **slave impersonation:**
  - The master checks the ASnd sent on the wire; the attacker can’t send an NMT command without being spotted by the master.

- **master reset:**
  - This attack needs the impersonation of a slave.

- **master impersonation:**
  - The authentication phase during start-up blocks this attacks.
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Industrial Safety Protocols

- Specific protocols used on top of an Industrial Ethernet protocol to ensure the safety of a system
  - For example: PROFIsafe, FSoE, CIP Safety, OpenSafety
- In an Industrial Safety network, “critical” sensors or activators are changed to become Safe Sensors and Safe Actuators.
- A Safety Master is also added.

![Diagram of Industrial Safety Protocols]

- HMI
- PLC
- Safety Master
- Sensor
- Actuator
- Safe Sensor
- Safe Actuator
- Actuator
Yes, it can be real
Message security

- Context: OpenSafety
  - Message size limitation
  - Message duplication

- Requirement: mutually authenticate messages
- Solution: CMAC-based authentication (space and timing constraints)
- Modifications to message structure:
  - Header (6 bytes)
    - Frame identifier for message identification
    - Security identifier for component identification
    - Sub-protocol message length
    - Sequence number to prevent replay attacks
  - Sub-protocols (adapted from standard IEC mechanisms):
    - Secure cyclic: authentication and integrity
    - Secure acyclic control: NMT messages security (e.g. for secure slaves)
    - Error reporting and heartbeat: reporting from the security master to the HMI
    - Key management
    - Extensions for monitoring only
Timings computed using the characteristics of the B&R testbed.

<table>
<thead>
<tr>
<th>Nb slaves</th>
<th>Data</th>
<th>No Security</th>
<th>Isolated SecMaster</th>
<th>Monitored master</th>
<th>Monitored slave</th>
<th>Full security</th>
<th>OpenSAFETY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>102.12µs</td>
<td>+36.7%</td>
<td>+37.6%</td>
<td>+39.4%</td>
<td>+43.1%</td>
<td>+36.75%</td>
</tr>
<tr>
<td>1</td>
<td>200</td>
<td>127.72µs</td>
<td>+49.4%</td>
<td>+59.3%</td>
<td>+60.7%</td>
<td>+79.3%</td>
<td>+106.2%</td>
</tr>
<tr>
<td>1</td>
<td>1490</td>
<td>334.12µs</td>
<td>+80.7%</td>
<td>+101%</td>
<td>+101%</td>
<td>+141%</td>
<td>(+231.25%)</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>815.00µs</td>
<td>+4.60%</td>
<td>+5.76%</td>
<td>+5.98%</td>
<td>+10.9%</td>
<td>+4.62%</td>
</tr>
<tr>
<td>20</td>
<td>200</td>
<td>1327.0µs</td>
<td>+4.76%</td>
<td>+14.7%</td>
<td>+14.9%</td>
<td>+33.7%</td>
<td>+62.1%</td>
</tr>
<tr>
<td>20</td>
<td>1490</td>
<td>5455.0µs</td>
<td>+4.94%</td>
<td>+17.8%</td>
<td>+17.8%</td>
<td>+43.2%</td>
<td>(+102%)</td>
</tr>
<tr>
<td>238</td>
<td>1</td>
<td>8994.4µs</td>
<td>+0.42%</td>
<td>+1.61%</td>
<td>+1.63%</td>
<td>+6.67%</td>
<td>+0.44%</td>
</tr>
<tr>
<td>238</td>
<td>200</td>
<td>15087µs</td>
<td>+0.42%</td>
<td>+10.4%</td>
<td>+10.4%</td>
<td>+29.3%</td>
<td>+57.9%</td>
</tr>
<tr>
<td>238</td>
<td>1490</td>
<td>64210µs</td>
<td>+0.42%</td>
<td>+12.8%</td>
<td>+12.8%</td>
<td>+37.4%</td>
<td>(+94.1%)</td>
</tr>
</tbody>
</table>
Outline

Introduction to Industrial Control protocols

Ethernet POWERLINK Protocol

Attacks

Defense and Mitigation

Detection and Protection

Lessons learned
Lessons learned for ICS security

- Ethernet POWERLINK communications are not secure.
  - Unsurprising...
- They can be secured at equivalent cost to safety.
  - (Upcoming work) We can obtain both security and safety with the same add-on.
- This can be extended to less constrained protocols.
Lessons learned for Intrusion Detection and Network Security

- Clear specifications provide opportunity for extension.
- Intrusion detection may benefit from additional support.
- The more security can be built-in the more benefits we obtain.
- Better detection benefits from the process knowledge.
  - **Software** Control flow for malicious code detection
  - **Protocols** Coherence of transport (flows, routing, ...)
  - **Industrial process** Control loop for logical-physical dependencies
  - **Business process** Interactions between physical, virtual and business assets
  - ...

- Need for a better equilibrium between protection, detection and mitigation
- Need for time-tested solutions: better awareness and training
Thank you for your attention!
Security Issues and Mitigation in Ethernet POWERLINK

Any questions?

AIRBUS Group Innovations & Télécom SudParis
Bibliography

Johan Åkerberg and Mats Björkman.
Exploring security in profinet io.

Daniele Antonioli and Nils Ole Tippenhauer.
Minicps: A toolkit for security research on cps networks.

Basecamp Digital Bond.

M Bristow.
Modscan: a scada modbus network scanner.
In *DefCon-16 Conf., Las Vegas, NV*, 2008.

Peter Huitsing, Rodrigo Chandia, Mauricio Papa, and Sujeet Shenoi.
Attack taxonomies for the modbus protocols.

Andreas Paul, Franka Schuster, and Hartmut König.
Towards the protection of industrial control systems—conclusions of a vulnerability analysis of profinet io.

Theodoros Spyridopoulos, Ioanna-Aikaterini Tzep, Theo Tryfonas, and Maria Koruda.