Institut
Mines-Télécom

Cybersecurity for Industry 4.0

Security Issues and Mitigation in Ethernet POWERLINK

Hervé Debar
Institut Mines-Télécom - Télécom SudParis
herve.debar@telecom-sudparis.eu

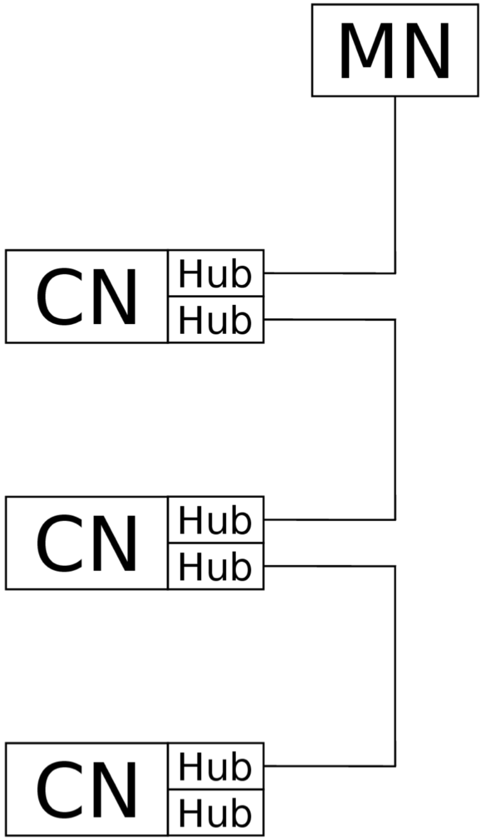
Joint work with Jonathan Yung & Louis Granboulan (AIRBUS Group Innovations)

Security of SCADA protocols

- **Focus of this talk: 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
- **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

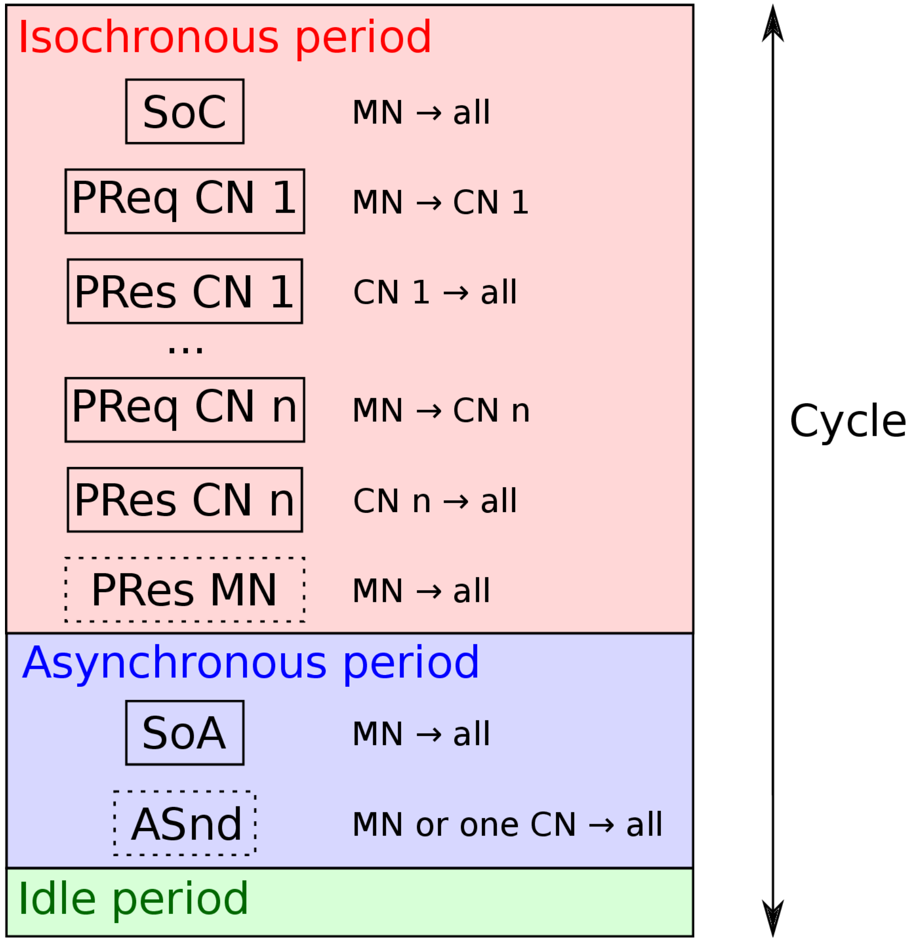
Ethernet POWERLINK Protocol Architecture

- **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 MN and CNs are connected through Hubs.**
- **Attacks require physical access to a free port**



Ethernet POWERLINK Protocol structure

- **Composed of three periods:**
  - Isochronous period
  - Asynchronous period
  - Idle period
- **Operated by a Network Management (NMT) state machine**
  - The MN can change the NMT state of a CN through an ASnd command.
  - A CN can ask the MN to send an NMT command to change the NMT state of a CN or of the MN.
  - NMT State of a CN are (non exhaustive list):
    - Init, pre_operational_1, pre_operational_2, ready_to_operate, operational, stopped



Isochronous period

SoC MN → all

PReq CN 1 MN → CN 1

PRes CN 1 CN 1 → all

...

PReq CN n MN → CN n

PRes CN n CN n → all

PRes MN MN → all

Asynchronous period

SoA MN → all

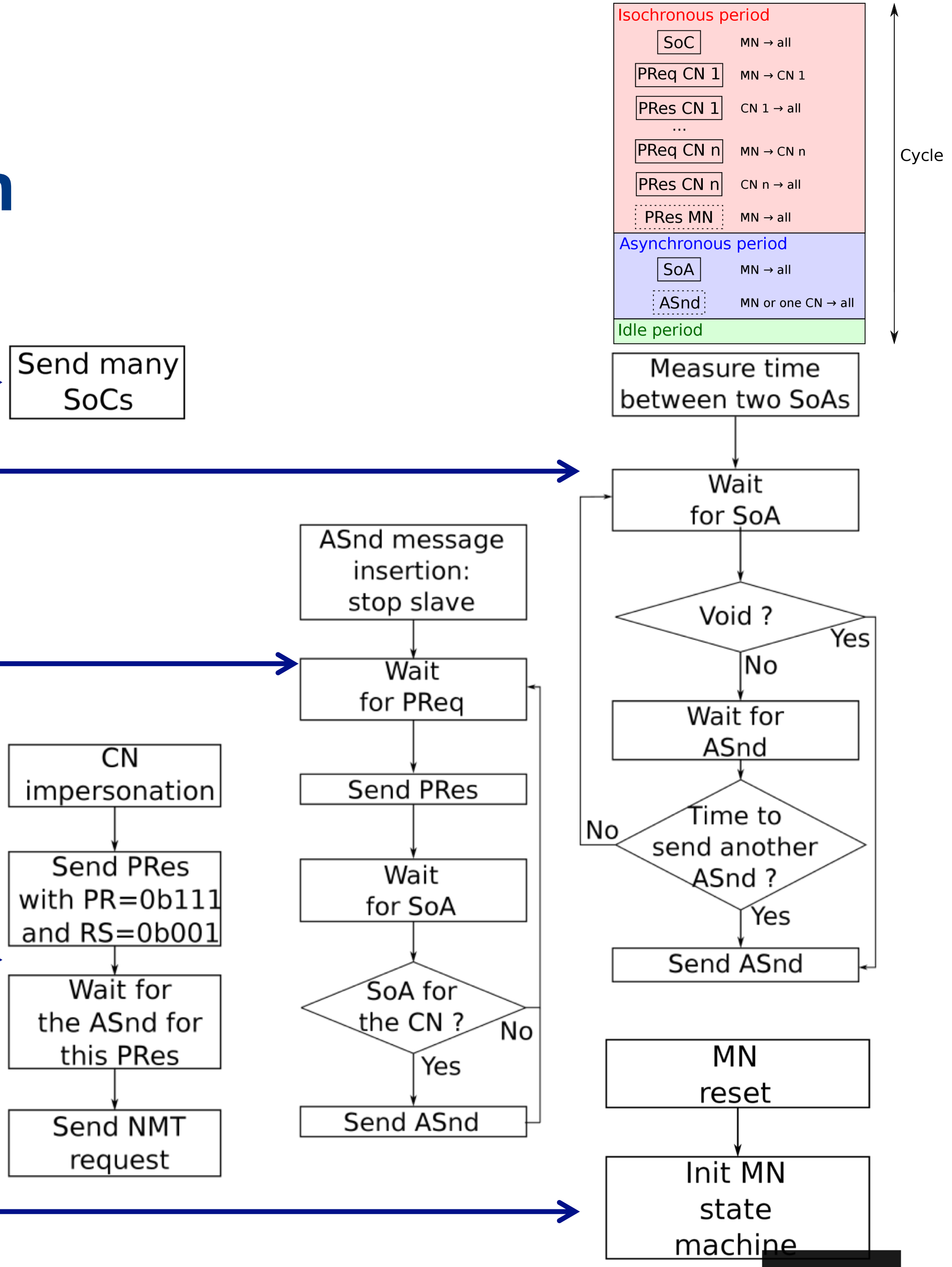
ASnd MN or one CN → all

Idle period

Cycle

Attacks description

- **Denial of service**
- **Acyclic command insertion**
- **CN impersonation**
- **MN reset**
- **MN impersonation**



Isochronous period	
SoC	MN → all
PReq CN 1	MN → CN 1
PRes CN 1	CN 1 → all
...	
PReq CN n	MN → CN n
PRes CN n	CN n → all
PRes MN	MN → all
Asynchronous period	
SoA	MN → all
ASnd	MN or one CN → all
Idle period	

Send many SoCs

Measure time between two SoAs → Wait for SoA → Void ? (Yes → Send ASnd; No → Wait for ASnd) → Time to send another ASnd ? (No → Wait for SoA; Yes → Send ASnd)

ASnd message insertion: stop slave → Wait for PReq → Send PRes → Wait for SoA → SoA for the CN ? (No → Wait for PReq; Yes → Send ASnd → Wait for PReq)

CN impersonation → Send PRes with PR=0b111 and RS=0b001 → Wait for the ASnd for this PRes → Send NMT request

MN reset → Init MN state machine

Attack results

Initial experiments

Attacks	B&R components	openPOWERLINK
Denial of service	OK	OK
Acyclic command insertion	~OK	OK
CN impersonation	Not OK	OK
MN reset	Not OK	OK
MN impersonation	OK	OK

Current experiments

Attacks	B&R components	openPOWERLINK
Denial of service	OK	OK
Acyclic command insertion	OK	OK
CN impersonation	~OK	OK
MN reset	OK	OK
MN impersonation	OK	OK

Summing Up the Attacks

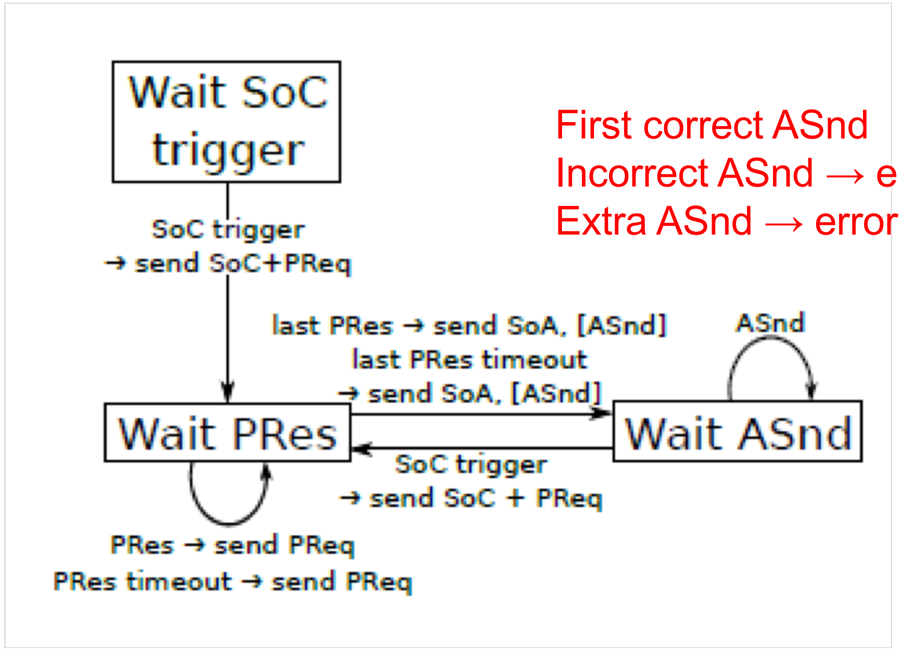
- **The Master/Slave paradigm simplifies any DoS attacks.**
  - we do not handle mitigation against DoS attacks here

- **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 CN

Attack Mitigation
State Machine Modification

Master Node

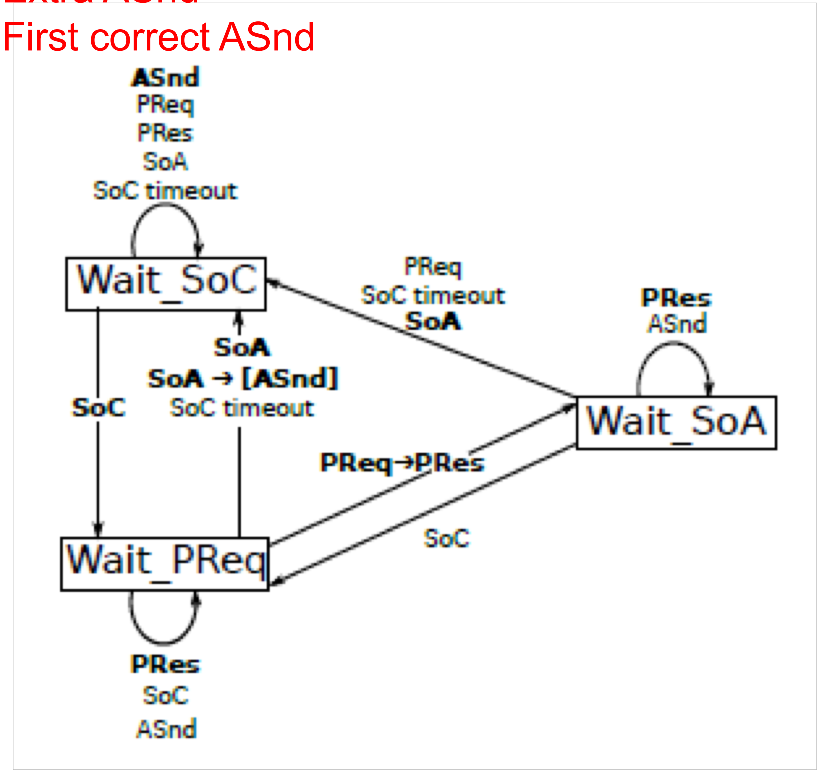
Controlled Node



Wait SoC trigger
SoC trigger → send SoC+PReq
last PRes → send SoA, [ASnd]
last PRes timeout → send SoA, [ASnd]
ASnd
Wait PRes
Wait ASnd
SoC trigger → send SoC + PReq
PRes → send PReq
PRes timeout → send PReq

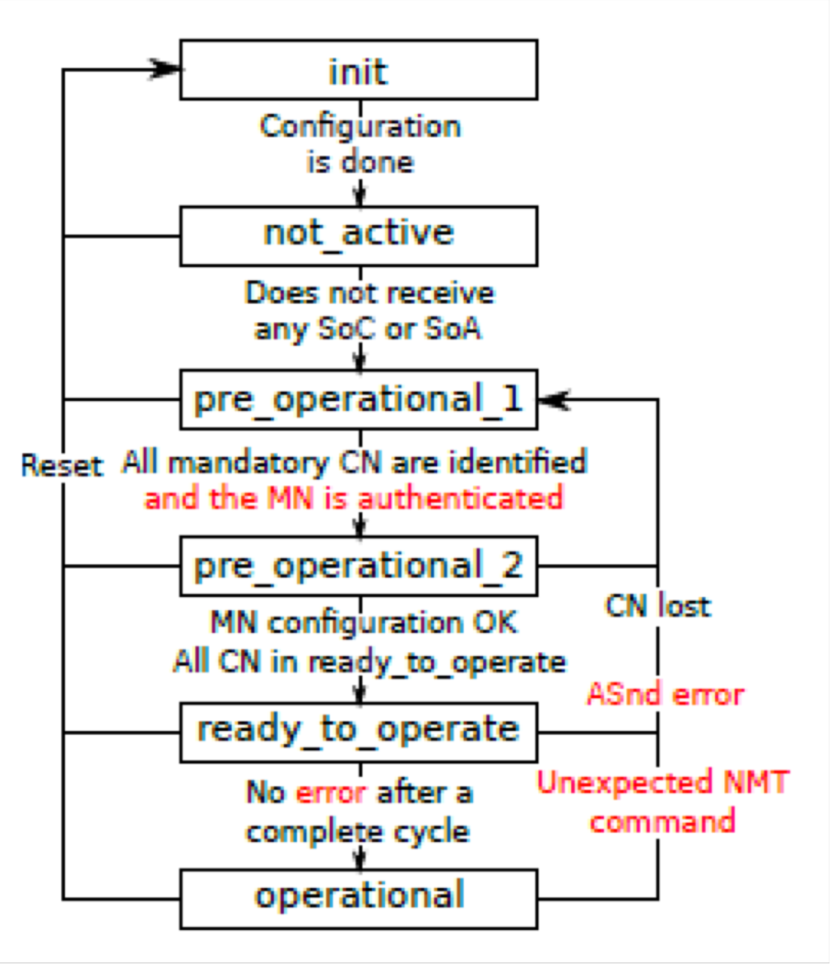
First correct ASnd
Incorrect ASnd → error
Extra ASnd → error

Incorrect ASnd
Extra ASnd
First correct ASnd



Attack Mitigation
MNT MN State Machine Modification

- **Better error checking**
- **Include authentication**



Mitigation evaluation

- **Denial of Service:**
  - not handled here
- **Acyclic command insertion:**
  - The CN 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 MN.
  - However, even in this case, it will be detected.
- **CN impersonation:**
  - Change of the MN state machine: the MN checks the ASnd sent on the wire
  - The attacker can’t send an NMT command without being spotted by the MN.
- **MN reset:**
  - This attack requires the impersonation of a CN.
- **MN impersonation:**
  - The authentication phase during start-up blocks this attacks.

Residual risk analysis

- **Ethernet POWERLINK communications are not totally secured:**
  - We can stop the communications.
  - We can inject commands.
  - We can inject input data.
  - We can impersonate an MN.
- **We proposed some modifications:**
  - reinforcing the asynchronous period
  - improving the start-up period

Design of a security master for Powerlink

- **Analogous to the OpenSafety context**
- **Adding one specific CN slave node to the network**
  - Safety related configuration
  - Check communications
  - Handle safety nodes
- **Modifying other nodes, CNs, as safety nodes to act on safety commands**
- **Modifications to the implementation of the protocol to protect against data corruption**
  - Limit to message size
  - CRC
  - Timestamps

Attacker model

- **Protection of the cyclic part of the cycle**
- **Acyclic part attacked through cyclic commands**
- **Integrity and authenticity attacks**
  - DoS not handled, extremely hard due to timing constraints
  - Confidentiality not handled, considered irrelevant
- **The attacker must be able to connect to a free RJ45 port**
  - Easy at the end of the chain
  - Possible with interruption in other places

SecurityMaster features

- **AES-CMAC on all data transported by the powerlink messages**
- **New secure messages/sub-protocols**
  - Network management for control messages sent by the SecurityMaster
  - Error reporting to securely report errors to the HMI through the MN
  - Key management: initialization, key change
- **Several configurations possible**
  - Isolated security master: reporting to HMI
  - Secure CN/monitored MN: detection of malicious commands
  - Secure MN/monitored CN: check for malicious responses
  - Full security

Security evaluation

	CN Impersonation	MN Impersonation	PRes modification	PReq modification
Isolated SecurityMaster	-	-	-	-
Monitored MN	Detected	Detected	Detected	
Monitored CN	-	Detected	-	Detected
Full security	Blocked	Blocked	Blocked	Blocked

Performance evaluation

Nb CN	Data size	Mon. MN	Mon. CN	Full sec.	Open-Safety
1	1	+0,64%	+1,94%	+4,66%	+0,01%
1	200	+4,08%	+5,18%	+13,5%	+22%
1	1490	+9,08%	+9,57%	+27,8%	N/A
20	1	+1,16%	+1,37%	+6,15%	+0,37%
20	200	+9,29%	+9,43%	+27,1%	+53,4%
20	1490	+12,1%	+12,2%	+36,2%	N/A
238	1	+1,26%	+1,28%	+6,40%	+0,49%
238	200	+9,93%	+9,94%	+28,7%	+57,2%
238	1490	+12,3%	+12,3%	+36,7%	N/A

Theoretical impact of the security protocol in terms of additional fields and cryptographic operations, using CMAC benchmark data

Conclusion

- **Securing isosynchronous protocols is feasible**
  - Requires adding a new node
  - Requires new protocol messages
  - Complies with Ethernet Powerlink specifications
  - Similar to accepted technical practices (OpenSafety)
- **Implementation needs to validate the proposal**
  - B&R automation testbed under way
  - Difficulties
    - Use cases
    - Programming

Thank you for your attention

Questions ?