



Public and Private, Common Flaws in ICS Network Protocols

Mars Cheng and Selmon Yang

Cyber Threat Researcher and Staff Engineer, TXOne Networks

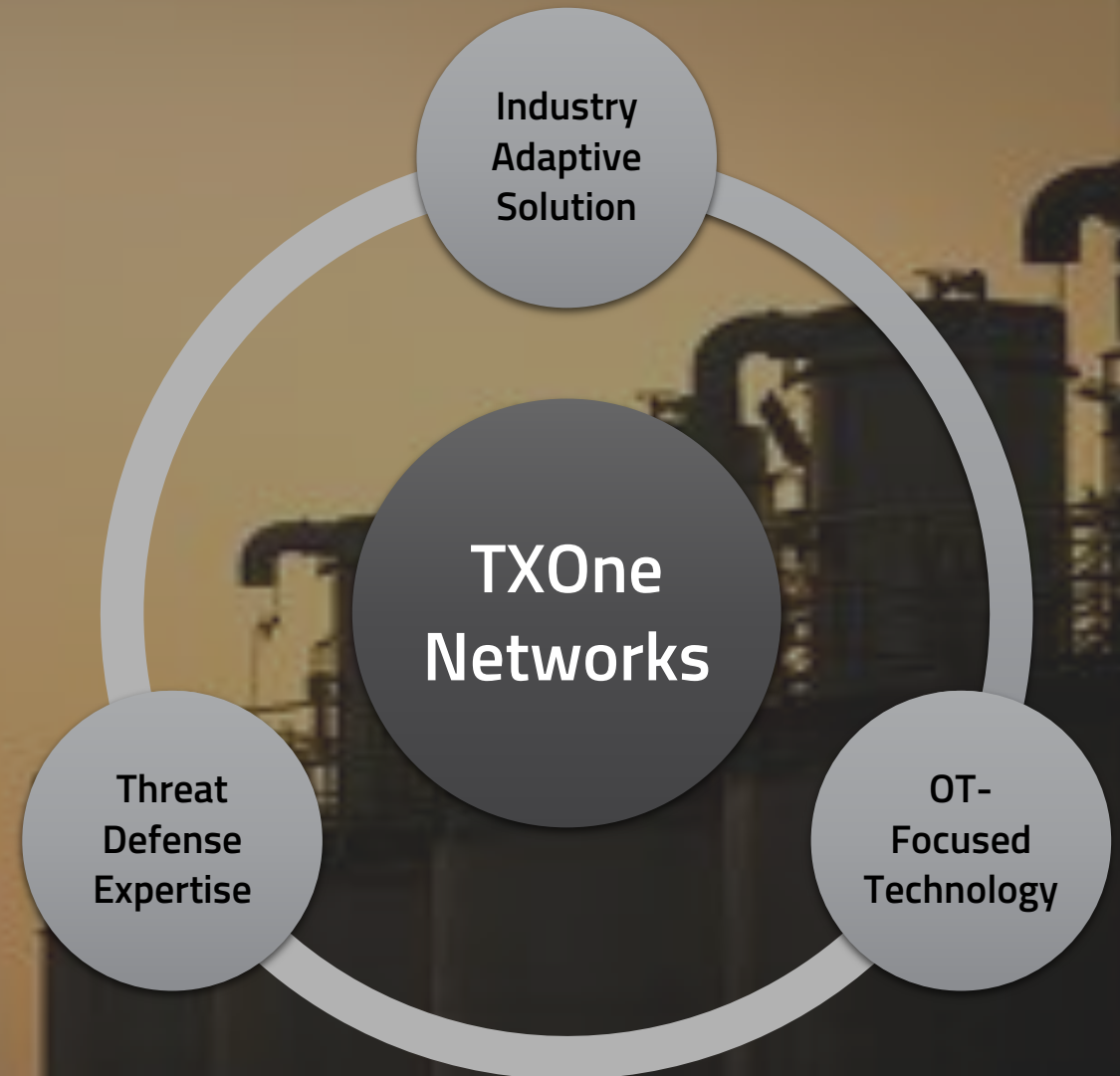
002
HITB LOCKDOWN
livestream



WHO WE ARE

A Joint venture company of
Trend Micro Inc. and **Moxa Inc.**

30 years+ Cybersecurity Threat Intelligent
30 years+ OT Network Expertise



To accelerate the industrial world to secure automation and data exchange

Who we are?



Mars Cheng

Cyber Threat Researcher

ICS/SCADA Security Research

Threat Hunting

Web, App, IoT, ICS/SCADA Penetration

Testing

Applied Cryptography



Selmon Yang

Staff Engineer

IT/SCADA Protocol Parsing

Linux Kernel Programming

Honeypot Deployment & Optimization



Outline

- ICS Architecture and Attack Vectors
- Public and Private: ICS Protocols
- Common Flaws in ICS Protocols
- How to Work Against ICS Network Protocol Attacks

The background of the slide is a photograph of an industrial facility, likely a refinery or chemical plant, silhouetted against a warm, orange-hued sky at sunset or sunrise. The facility consists of several tall, cylindrical distillation columns and large storage tanks, interconnected by a complex network of pipes and walkways. The lighting is dramatic, with the sun low on the horizon, creating a strong contrast between the dark structures and the bright sky.

ICS Architecture and Attack Vectors

PURDUE REFERENCE ARCHITECTURE

SITE
MANUFACTURING
OPERATION &
CONTROL

CELL/AREA
SUPERVISORY
CONTROLS

CONTROL

PROCESS

LEVEL 4/5

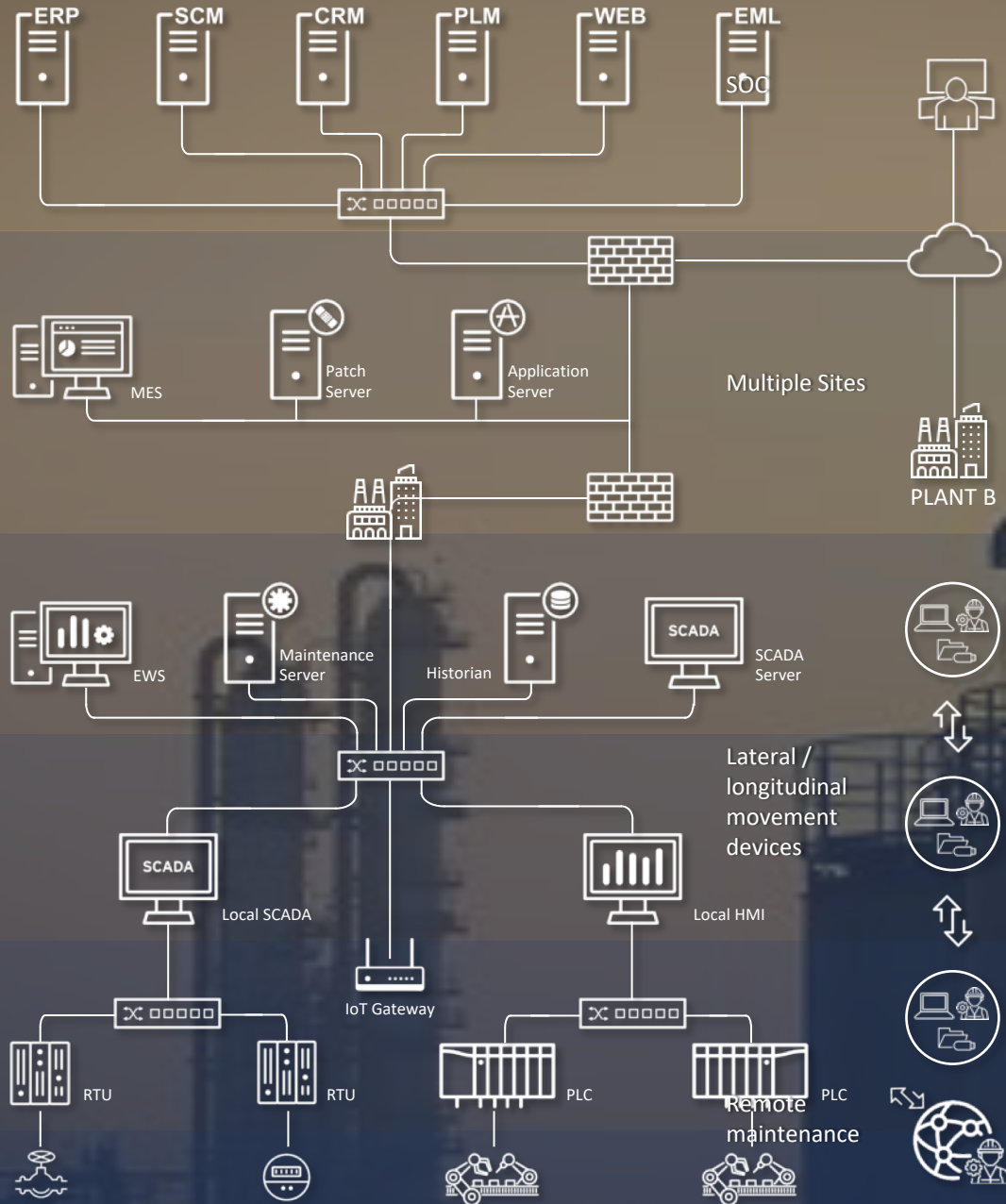
DMZ

LEVEL 3

LEVEL 2

LEVEL 1

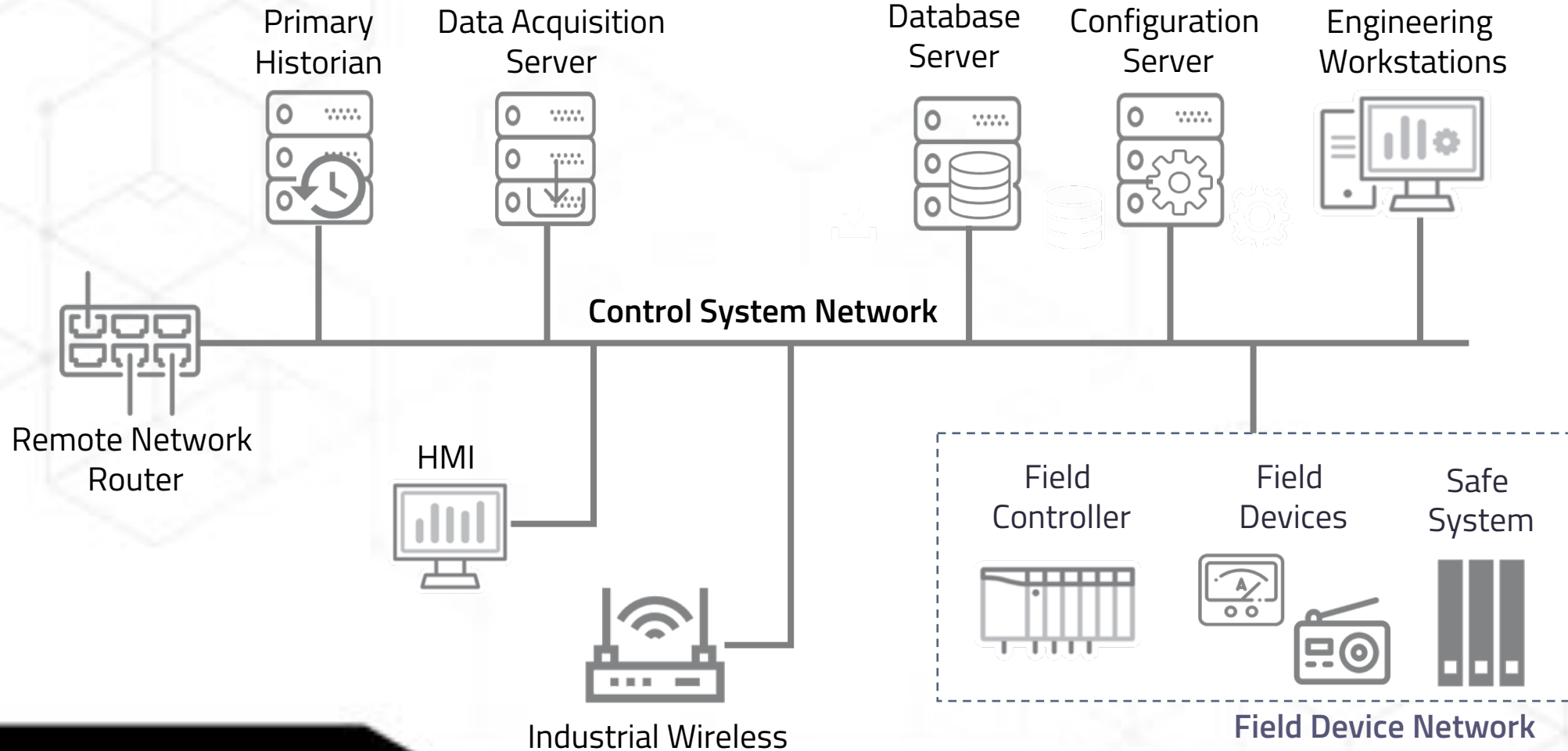
LEVEL 0



Information Technology (IT)

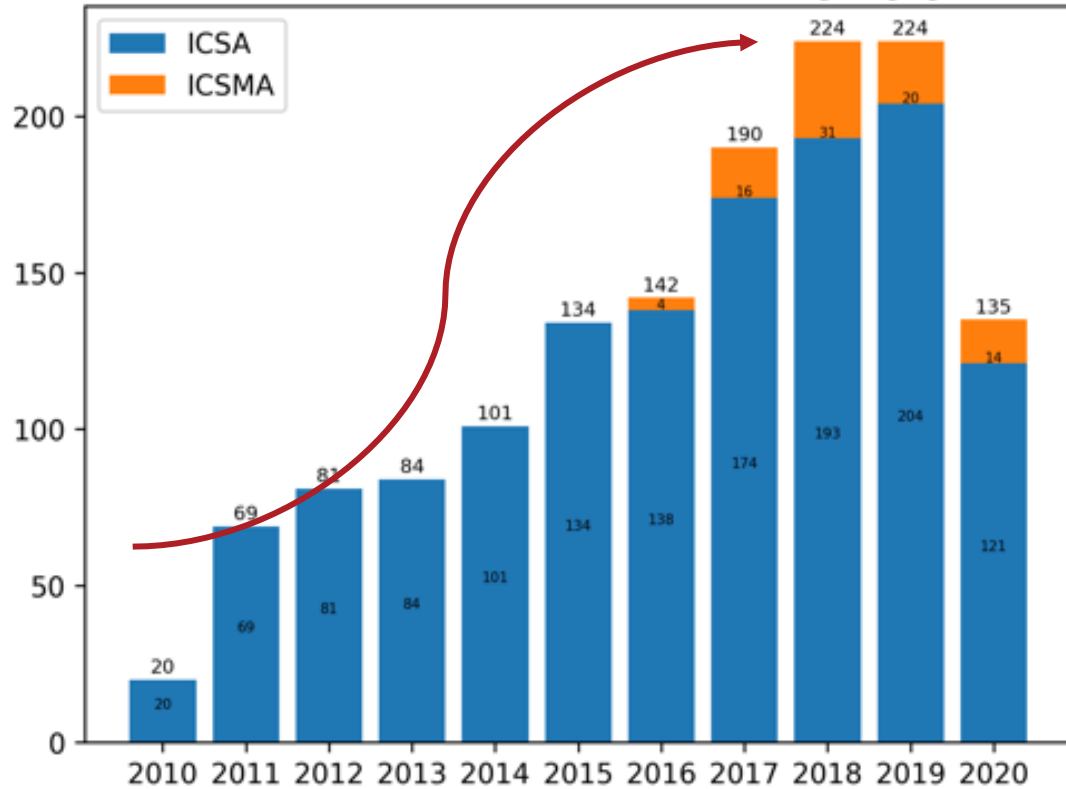
Operational Technology (OT)

Common ICS Architecture

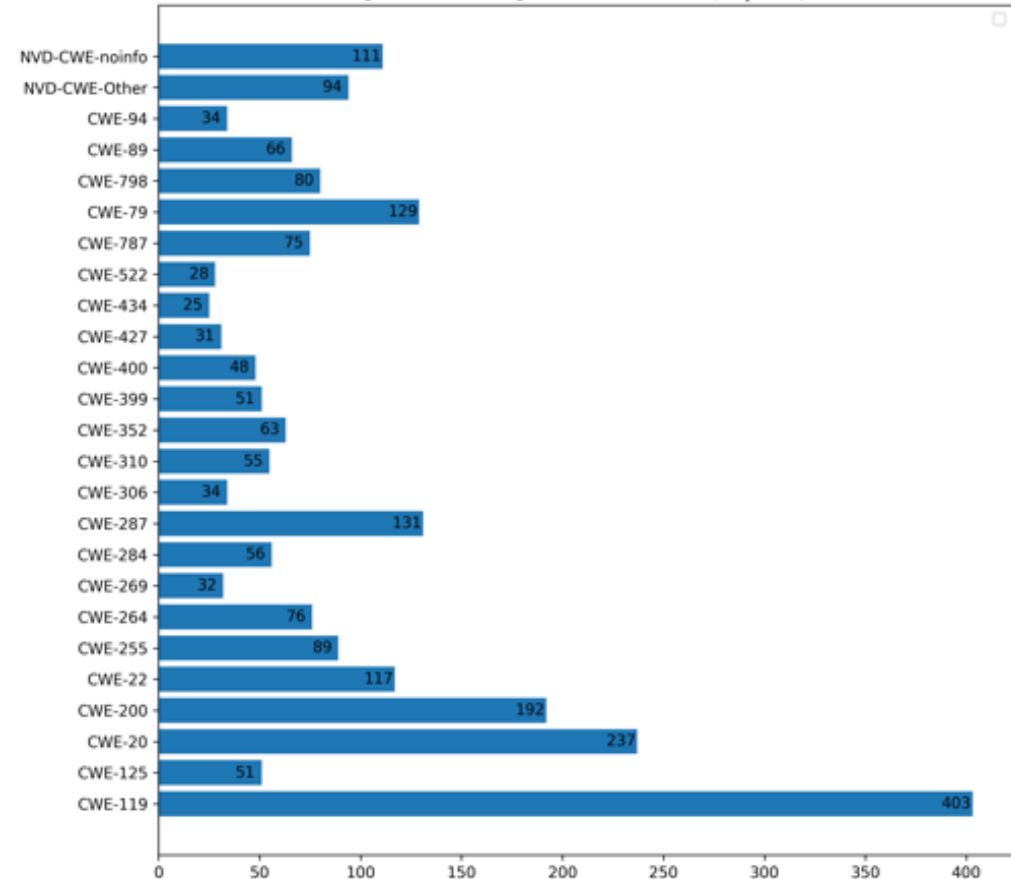


ICS-Related Vulnerabilities Information

[2020-07-13] ICS-CERT Advisory by year

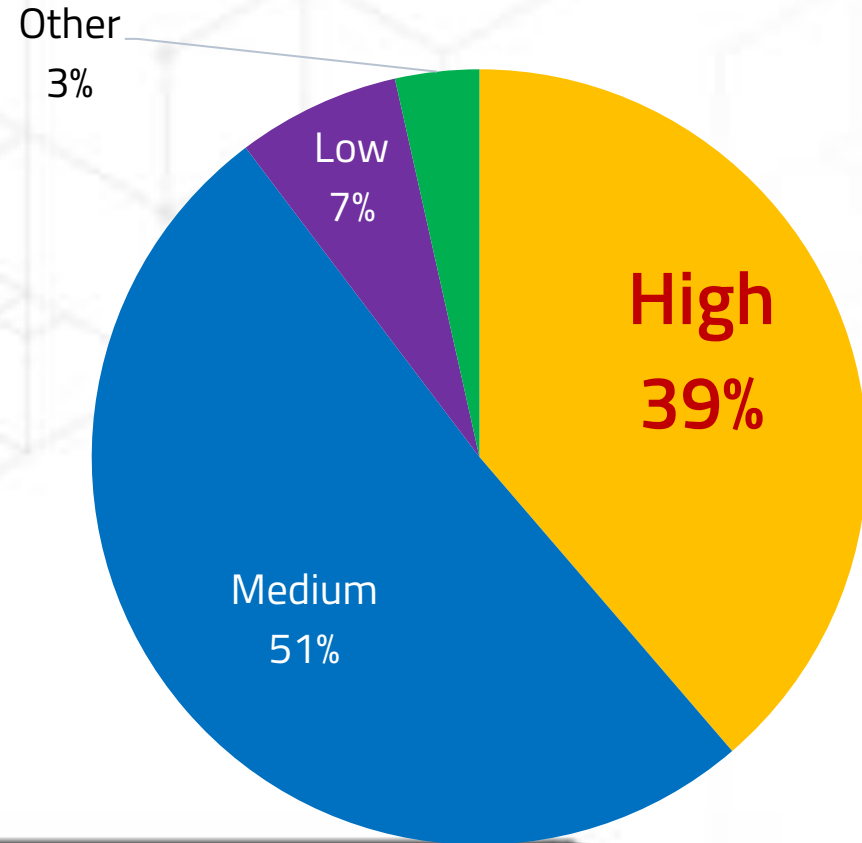


[2020-07-13] CWE Statistics (Top 25)

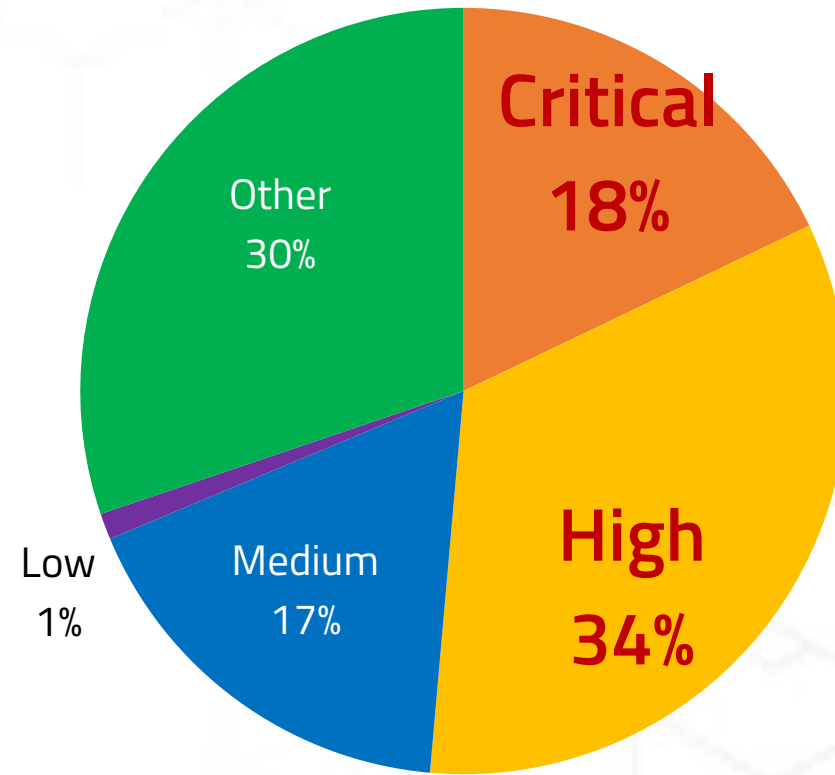


ICS-Related Vulnerabilities Information

[2020-07-13] CVSS2.0 Statistics



[2020-07-13] CVSS 3.0 Statistics





ICS ATT&CK Matrix-Stuxnet

Initial Access	Execution	Persistence	Evasion	Discovery	Lateral Movement	Collection	Command and Control	Inhibit Response Function	Impair Process Control	Impact
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Detect Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File Infection	Rogue Master Device	Network Service Scanning	Program Organization Units	Detect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image		Block Serial Comm Port	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File Infection		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
								Program Download		
								Rootkit		
							System Firmware			
							Utilize/Change Operating Mode			

11 Tactics
81 Techniques



ICS ATT&CK Matrix-Industroyer, CRASHOVERRIDE

Initial Access	Execution	Persistence	Evasion	Discovery	Lateral Movement	Collection	Command and Control	Inhibit Response Function	Impair Process Control	Impact
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Detect Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File Infection	Rogue Master Device	Network Service Scanning	Program Organization Units	Detect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image		Block Serial COM	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File Infection		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
								Program Download		
								Rootkit		
							System Firmware			
							Utilize/Change Operating Mode			

11 Tactics
81 Techniques



ICS ATT&CK Matrix-Triton, Trisis, Hatman

Initial Access	Execution	Persistence	Evasion	Discovery	Lateral Movement	Collection	Command and Control	Inhibit Response Function	Impair Process Control	Impact
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Detect Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File	Rogue Master Device	Network Service Scanning	Program Organization Units	Detect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image		Block Serial Comm Port	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
						Program Download				
						Rootkit				
						System Firmware				
						Utilize/Change Operating Mode				

11 Tactics
81 Techniques

ICS/SCADA Security Threat Situation



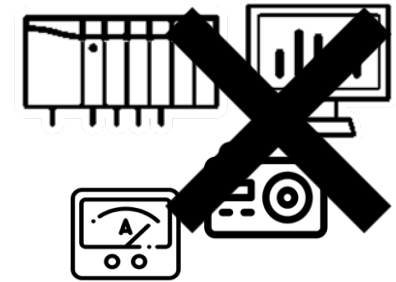
Vulnerabilities are mostly critical and high risk levels



The number of vulnerability is rising year by year



The security incidents have a huge impact

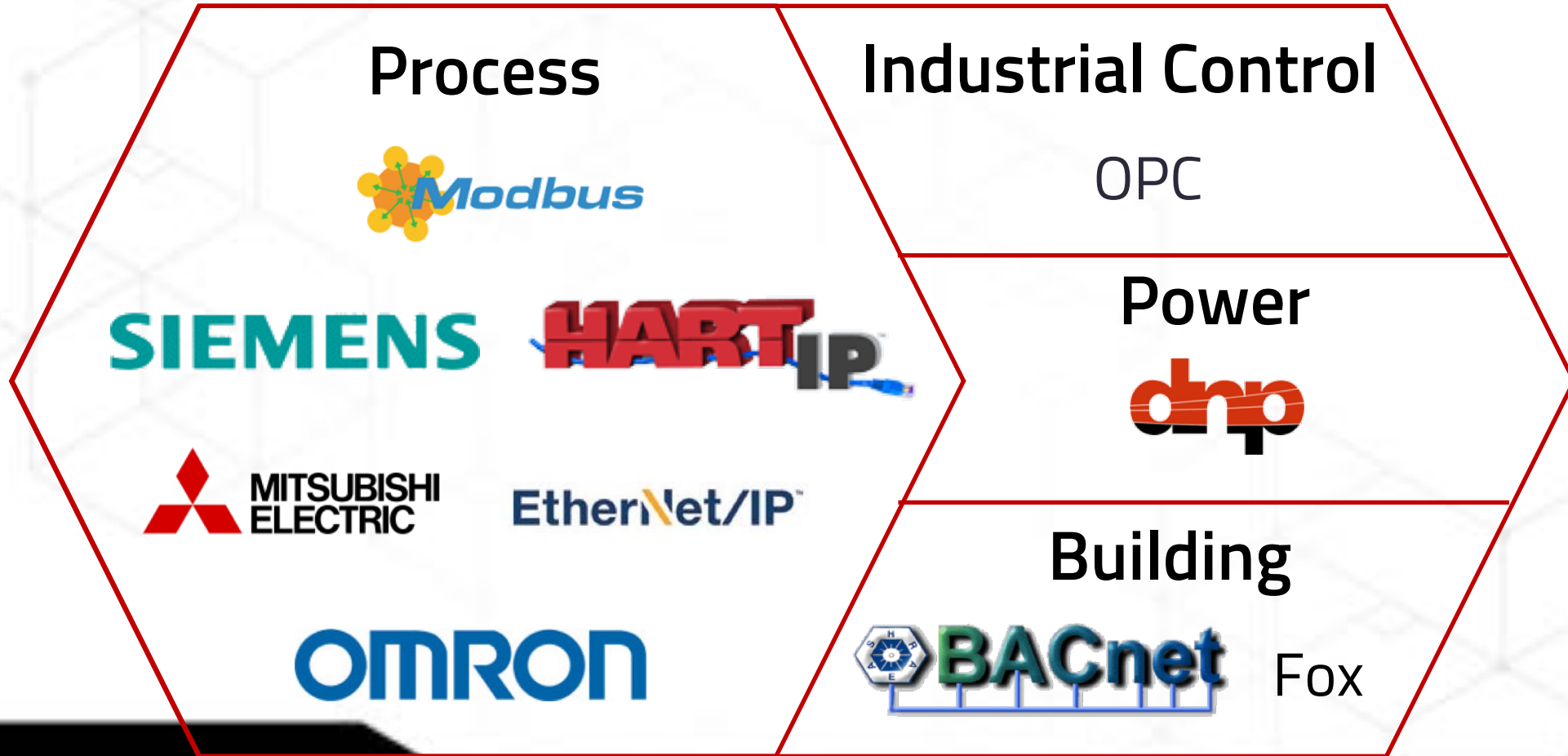


ICS/SCADA are not secure at all

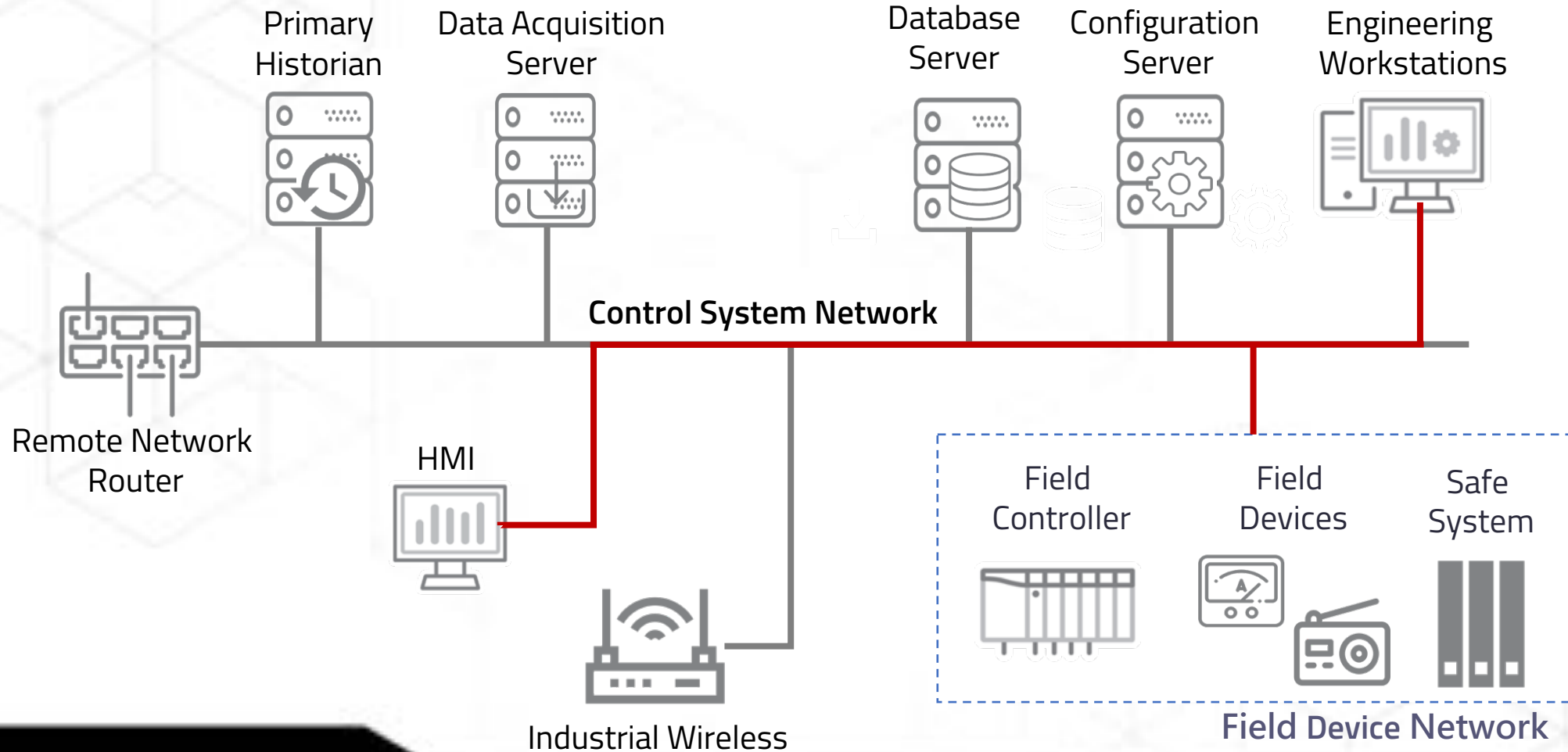
Critical

ICS Protocols

ICS Protocols



The Communication of ICS Protocols



Critical Infrastructure Sectors (Taiwan)



High-Tech Park



Energy



Traffic



Communications



Government



Water



Finance



Medical

ICS Protocols and Critical Infrastructure Sectors (Singapore)



Aviation



Maritime



Water



Transport



Healthcare



Energy



Government



Security and Emergency Services



Banking and Finance



Infocomm



Media

ICS Protocols and Critical Infrastructure Sectors (Japan)



Aviation



Financial



Airport



Gas



Water



Information and communication



Medical



Electric power supply



Railway



Chemical



Credit card



Government and administrative



Petroleum



Logistics

ICS Protocols and Critical Infrastructure Sectors (US)



Chemical



Commercial Facilities



Communications



Critical Manufacturing



Emergency Services



Energy



Financial Services



Healthcare and Public Health



Transportation Systems



Food and Agriculture



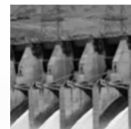
Defense Industrial Base



Government Facilities



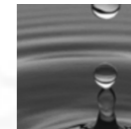
Nuclear Reactors, Materials,
and Waste



Dams



Information Technology



Water and Wastewater
Systems

ICS Protocols and Critical Infrastructure Sectors (US)



Chemical



Commercial Facilities



Communications



Critical Manufacturing



Emergency Services



Energy



Financial Services



Healthcare and Public Health



Transportation Systems



Food and Agriculture



Defense Industrial Base



Government Facilities



**Nuclear Reactors, Materials,
and Waste**



Dams



Information Technology



**Water and Wastewater
Systems**



Public and Private: ICS Network Protocols

Why Public vs. Private Protocols?

Public



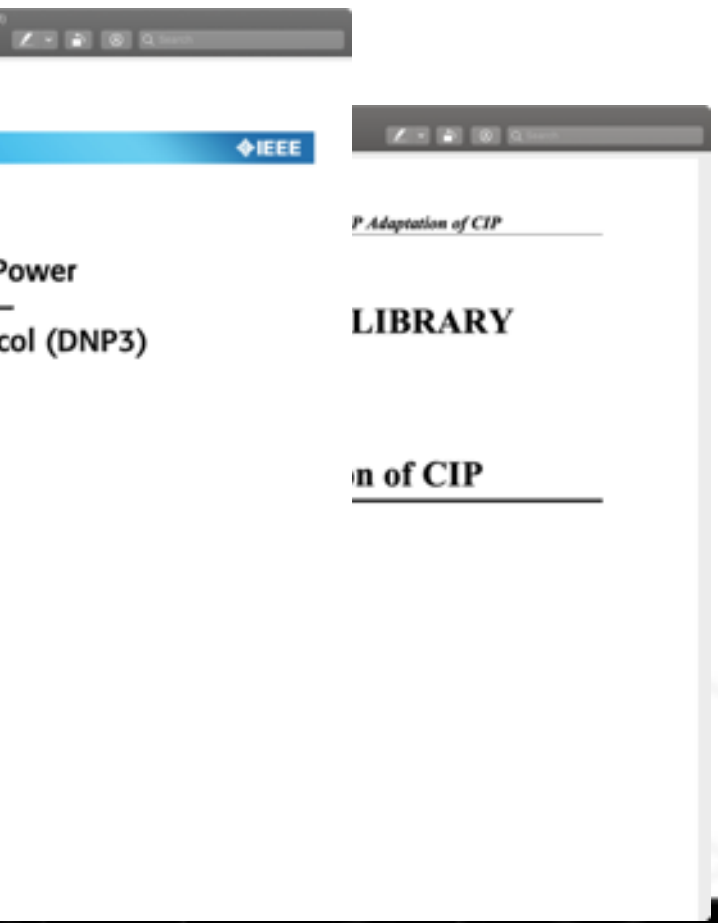
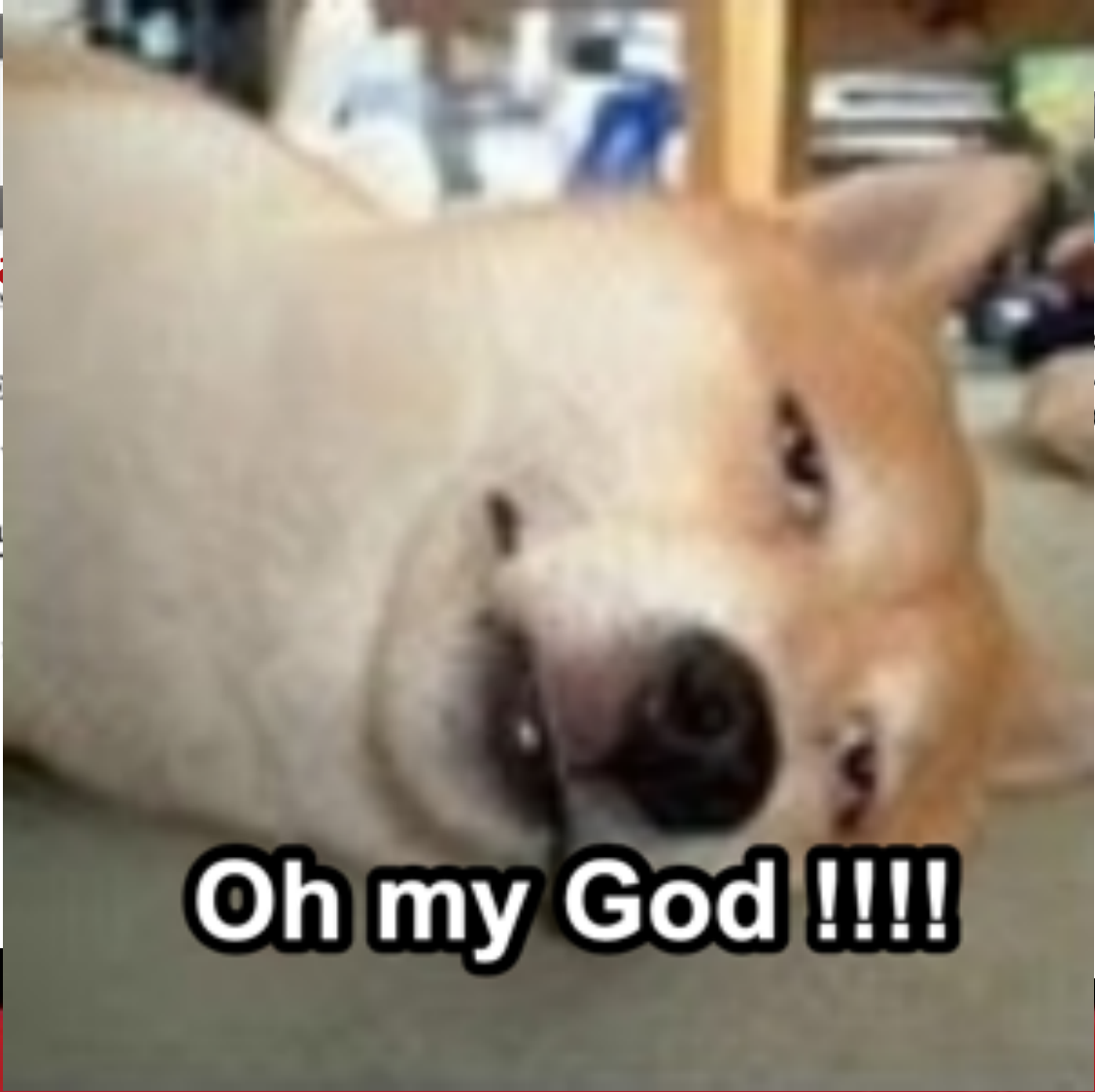
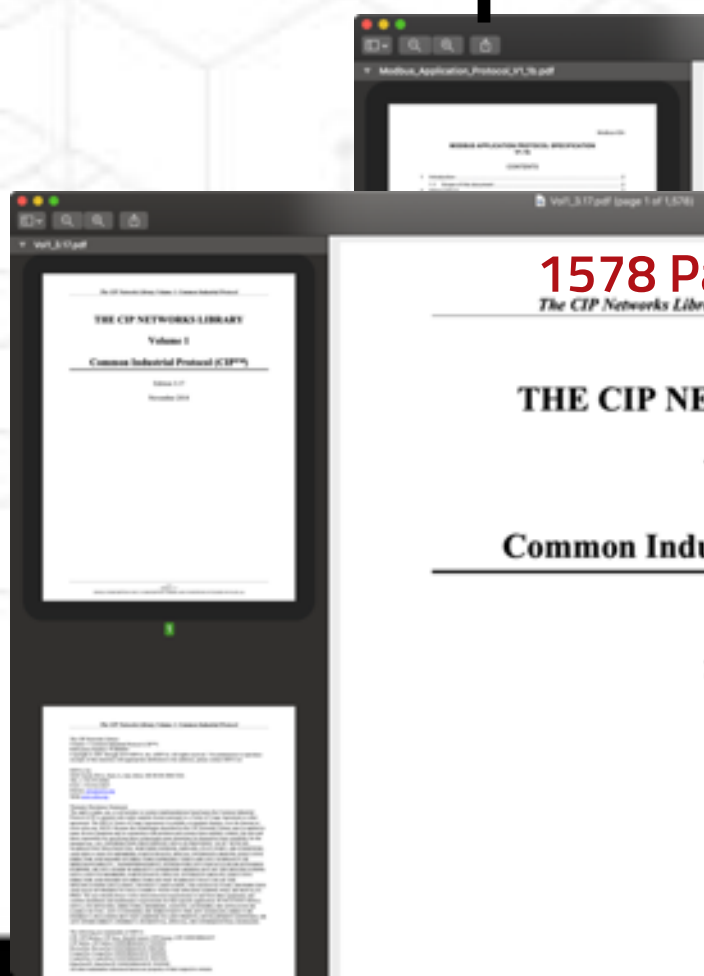
Private



EtherNet/IP

SIEMENS

The Specification of Public Protocols



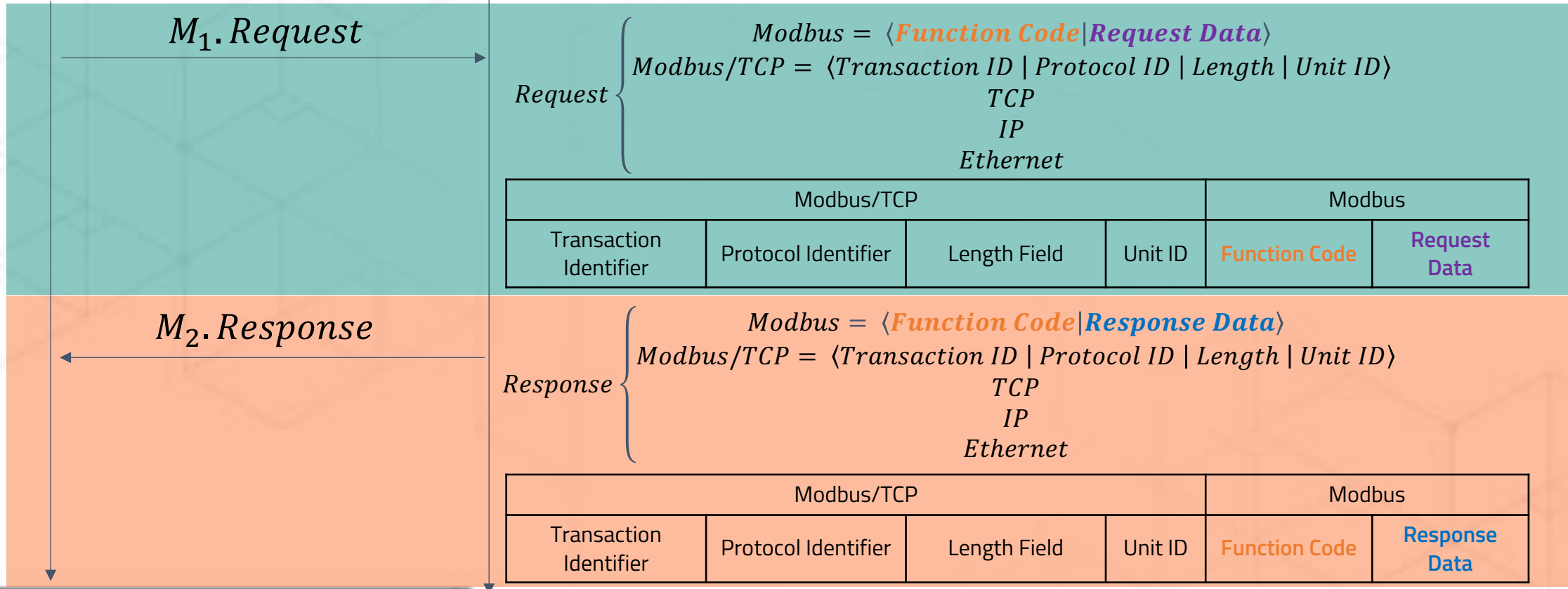
Modbus/TCP Handshake Process



HMI



PLC



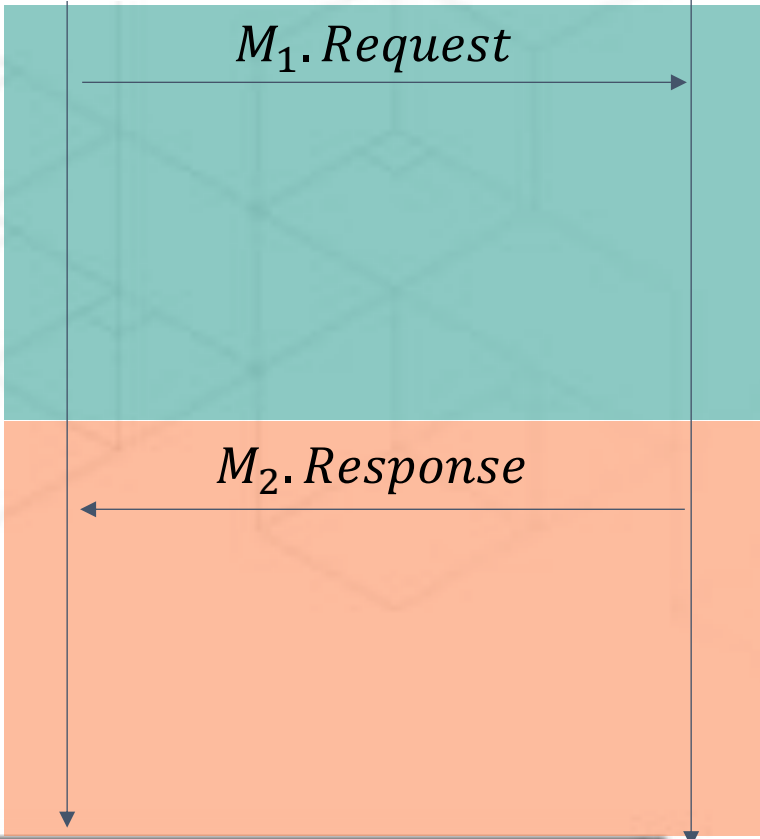
Modbus/TCP Handshake Process



HMI



PLC



M₂.Response

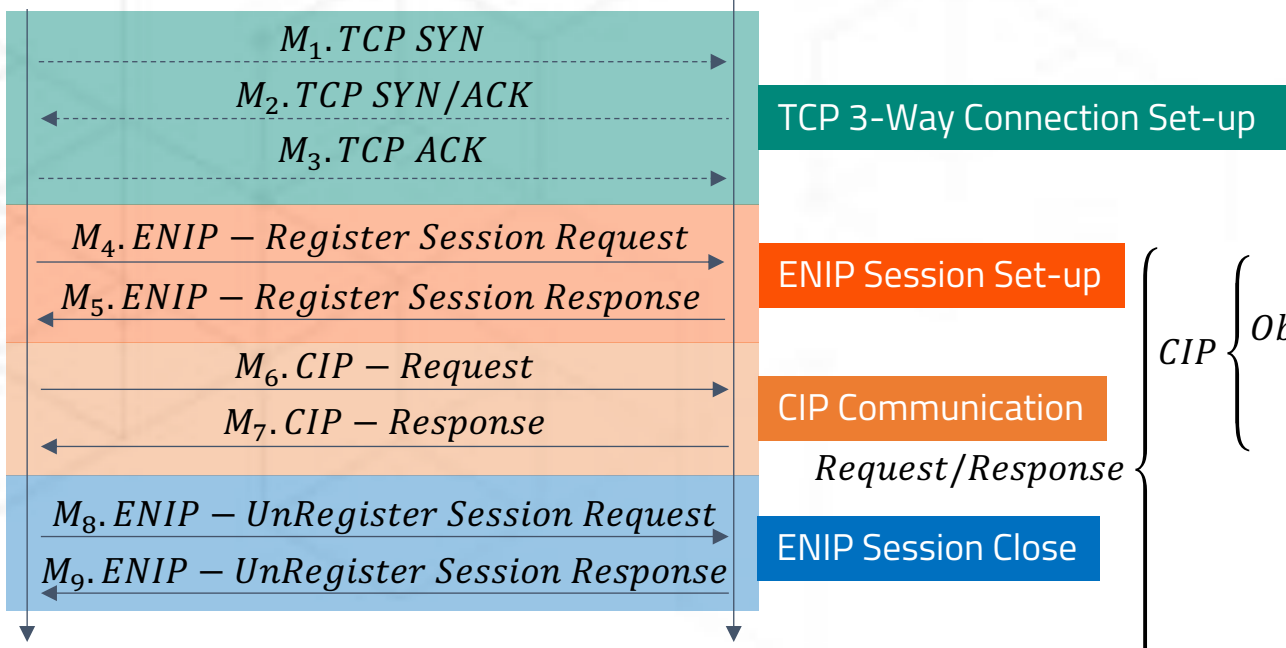
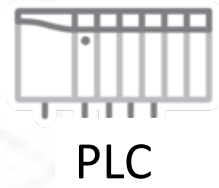
No.	Time	Source	Destination	Protocol	Length	Info
697	249252826.8...	18.1.1.234	18.1.1.234	Modbus/TCP	78	Query: Trans: 76; Unit: 255, Func: 3; Read Holding Registers
698	249252826.8...	18.1.1.234	18.1.1.234	Modbus/TCP	275	Response: Trans: 76; Unit: 255, Func: 3; Read Holding Registers

Transaction Identifier: 76
Protocol Identifier: 0
Length: 283
Unit Identifier: 255

Modbus
.000 0011 = Function Code: Read Holding Registers (3)
[Request Frame: 697]
[Time from request: 0.0164730000 seconds]
Byte Count: 200
Register 500 (UINT16): 00
Register 501 (UINT16): 0
Register 502 (UINT16): 60
Register 503 (UINT16): 0
Register 504 (UINT16): 0
Register 505 (UINT16): 0
Register 506 (UINT16): 0
Register 507 (UINT16): 0
Register 508 (UINT16): 0
Register 509 (UINT16): 0
Register 510 (UINT16): 0
Register 511 (UINT16): 0
Register 512 (UINT16): 0
Register 513 (UINT16): 0
Register 514 (UINT16): 0
Register 515 (UINT16): 0
Register 516 (UINT16): 0
Register 517 (UINT16): 0

.000 0011 = Function
Reference Number: 54
Word Count: 100

EtherNet/IP CIP Handshake Process



Request/Response {
EtherNet/IP {
Encapsulation Header
Command Specific Data
TCP/UDP
IP
Ethernet

CIP {
CIP Motion | Motion Control | Semiconductor Profiles
Object Library (Communication, Applications, Time Synchronization)
Data Management Services Explicit and I/O Messages
Connection Management, Routing

EtherNet/IP {
Encapsulation Header
Command Specific Data
TCP/UDP
IP
Ethernet

Command	Length	Session Handle	Status	Max Delay	Sender Context	Options	Command-specific Data
2 bytes	2 bytes	4 bytes	4 bytes	2 bytes	6 bytes	4 bytes	6 bytes

Function Code

- Get Attributes All 0x01
- Set Attributes All 0x02
- Get Attribute List 0x03
- Set Attribute List 0x04
- Start 0x06
- Stop 0x07

Apply a display filter: <Kt>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.24.254.49	10.60.60.60	TCP	74	38878 → 44818 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=1977658066 TSecr=0 WS
2	0.000285	10.60.60.60	10.24.254.49	TCP	74	44818 → 38878 [SYN, ACK] Seq=0 Ack=1 Min=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=2583868386
3	0.000383	10.24.254.49	10.60.60.60	TCP	66	38878 → 44818 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=1977658066 TSecr=2583868386
4	0.000693	10.24.254.49	10.60.60.60	DNTP	94	Register Session (Req), Session: 8x00000000
5	0.000860	10.60.60.60	10.24.254.49	TCP	66	44818 → 38878 [ACK] Seq=1 Ack=29 Win=29056 Len=0 TSval=2583868387 TSecr=1977658066
6	0.002458	10.60.60.60	10.24.254.49	DNTP	94	Register Session (Rsp), Session: 8x12345678
7	0.002458	10.24.254.49	10.60.60.60	TCP	66	38878 → 44818 [ACK] Seq=29 Ack=29 Win=29312 Len=0 TSval=1977658068 TSecr=2583868388
8	0.003934	10.24.254.49	10.60.60.60	CIP	136	Identity - Get Attribute List
9	0.009653	10.60.60.60	10.24.254.49	CIP	201	Success: Identity - Get Attribute List
10	0.052188	10.24.254.49	10.60.60.60	TCP	66	38878 → 44818 [ACK] Seq=99 Ack=164 Min=38336 Len=0 TSval=1977658118 TSecr=2583868395
11	3.329832	10.24.254.49	10.60.60.60	TCP	66	38878 → 44818 [FIN, ACK] Seq=99 Ack=164 Win=38336 Len=0 TSval=1977661396 TSecr=2583868395
12	3.330282	10.60.60.60	10.24.254.49	TCP	66	44818 → 38878 [FIN, ACK] Seq=164 Ack=100 Win=29056 Len=0 TSval=2583863716 TSecr=1977661396
13	3.330338	10.24.254.49	10.60.60.60	TCP	66	38878 → 44818 [ACK] Seq=100 Ack=165 Min=38336 Len=0 TSval=1977661396 TSecr=2583863716

Frame 9: 201 bytes on wire (1608 bits), 201 bytes captured (1608 bits)

- ▶ Ethernet II, Src: Dell_c0:00:a2 (64:00:6acc:00:a2), Dst: Vmware_f5:00:f1 (08:0c:29:f5:00:f1)
- ▶ Internet Protocol Version 4, Src: 10.60.60.60, Dst: 10.24.254.49
- ▶ Transmission Control Protocol, Src Port: 44818, Dst Port: 38878, Seq: 29, Ack: 99, Len: 135
- ▶ EtherNet/IP (Industrial Protocol), Session: 8x12345678, Send RR Data
- ▶ Common Industrial Protocol
- ▶ Service: Get Attribute List (Response)
 - 1... = Request/Response: Response (8x1)
 - .000 0011 = Service: Get Attribute List (0x03)
- ▶ status: success
- [Request Path Size: 2 words]
- ▶ [Request Path: Identity, Instance: 8x01]

```
0000 00 0c 29 f5 00 f1 64 00 6a cd 00 a2 00 00 45 00  --}...d:j.....E
0010 00 00 fe ef 40 00 3f 06 ed 00 0a 3c 3c 3c 0a 18  --...g-?....-ooc-
0020 fe 31 af 12 97 de cf ff 51 96 73 0e c7 4c 00 18  --1.....0 s:L...
0030 00 e3 1d 38 00 00 01 01 00 0a 9a 02 94 ab 75 e0  --...8.....-u...
0040 aa d5 0f 00 0f 00 78 56 34 12 00 00 00 00 00 00  --.a.o.v 4.....
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  --.....-.....
0060 82 00 00 00 00 00 b2 00 5f 00 83 00 00 00 0b 00  --....._.....
0070 01 00 00 00 01 00 82 00 00 00 0e 00 83 00 00 00  --.....T.....
0080 35 00 04 00 00 00 14 0b 05 00 00 00 60 31 06 00  --$.....-...1...
0090 00 00 1a 06 6c 00 07 00 00 00 14 31 37 35 36 2d  --...L.....-1756-
00a0 4c 35 31 2f 42 20 4c 4f 47 49 50 35 35 36 31 00  --L&L/B LO GIX5561-
00b0 00 00 00 ff 09 00 00 00 00 00 0a 00 00 00 00 0b  --.....-.....
00c0 00 00 00 05 54 58 4f 4e 45  --...TXON E
```

EtherNet/IP Traffic

The image shows a Wireshark packet capture window titled "enip_nw_attr_priv_violation.pcap". The main pane displays a list of 18 packets. Packet 8 is highlighted, showing a CIP Connection Manager Forward Open request. The details pane for this packet is expanded to show the "Service Forward Open (Request)" section, which includes fields for priority, tick time, time-out ticks, actual time out, network connection IDs, serial numbers, originator vendor ID (Rockwell Software, Inc.), and reserved fields.

No.	Time	Source	Destination	Protocol	Length	Info
5	0.007583	192.168.1.10	192.168.1.250	TCP	60	44818 → 34248 [ACK] Seq=1 Ack=29 Win=8164 Len=0
6	0.007632	192.168.1.10	192.168.1.250	ENDP	82	Register Session (Rsp), Session: 0x00730001
7	0.007690	192.168.1.250	192.168.1.10	TCP	54	34248 → 44818 [ACK] Seq=29 Ack=29 Win=29312 Len=0
8	0.010134	192.168.1.250	192.168.1.10	CIP CM	142	Connection Manager - Forward Open (Message Router)
9	0.013060	192.168.1.10	192.168.1.250	TCP	60	44818 → 34248 [ACK] Seq=29 Ack=117 Win=8184 Len=0
10	0.022088	192.168.1.10	192.168.1.250	CIP CM	124	Success: Connection Manager - Forward Open
11	0.025012	192.168.1.250	192.168.1.10	CIP	112	Class (0x6b) - Get Attribute List
12	0.031926	192.168.1.10	192.168.1.250	TCP	60	44818 → 34248 [ACK] Seq=99 Ack=175 Win=8134 Len=0
13	0.035021	192.168.1.10	192.168.1.250	CIP	111	Success: Class (0x6b) - Get Attribute List
14	0.038979	192.168.1.250	192.168.1.10	CIP	113	Class (0x6b) - Set Attribute List
15	0.042754	192.168.1.10	192.168.1.250	TCP	60	44818 → 34248 [ACK] Seq=156 Ack=234 Win=8133 Len=0
16	0.049067	192.168.1.10	192.168.1.250	CIP	104	Privilege violation: Class (0x6b) - Set Attribute List
17	0.052599	192.168.1.250	192.168.1.10	CIP	112	Class (0x6b) - Get Attribute List
18	0.056473	192.168.1.10	192.168.1.250	TCP	60	44818 → 34248 [ACK] Seq=206 Ack=292 Win=8134 Len=0

Frame 8: 142 bytes on wire (1136 bits), 142 bytes captured (1136 bits) on interface 0
Ethernet II, Src: IntelCor_Sa:7f:f8 (08:00:27:00:7f:f8), Dst: Rockwell_c7:b0:70 (08:00:27:00:c7:b0:70)
Internet Protocol Version 4, Src: 192.168.1.250, Dst: 192.168.1.10
Transmission Control Protocol, Src Port: 34248, Dst Port: 44818, Seq: 29, Ack: 29, Len: 88
EtherNet/IP (Industrial Protocol), Session: 0x00730001, Send RR Data
Common Industrial Protocol
CIP Connection Manager
Service Forward Open (Request)
0... .. = Request/Response: Request (0x0)
.100 0100 = Service Forward Open (0x54)
Command Specific Data
...0 = Priority: 0
.... 0000 = Tick time: 0
Time-out ticks: 249
Actual Time Out: 249ms
D→T Network Connection ID: 0x00000031
T→D Network Connection ID: 0x00fe0030
Connection Serial Number: 0x1337
Originator Vendor ID: Rockwell Software, Inc. (0x004d)
Originator Serial Number: 0xdeadbeef
Connection Timeout Multiplier: *4 (0)
Reserved: 0x000000
D→T RPI: 0000.0000s
D→T Network Connection Parameters: 0x0336

```
0050 00 00 00 00 02 00 00 00 00 00 02 00 30 00 54 02 ..... 0 T
0060 20 05 24 01 00 19 31 00 00 00 30 00 fe 00 37 13 ..$...I...0...7
0070 4d 00 ef be ad de 00 00 00 00 00 12 7a 00 14 43 M.....2...C
0080 00 12 7a 00 14 43 a3 03 01 00 20 02 24 01 ..2..C... ..$
```

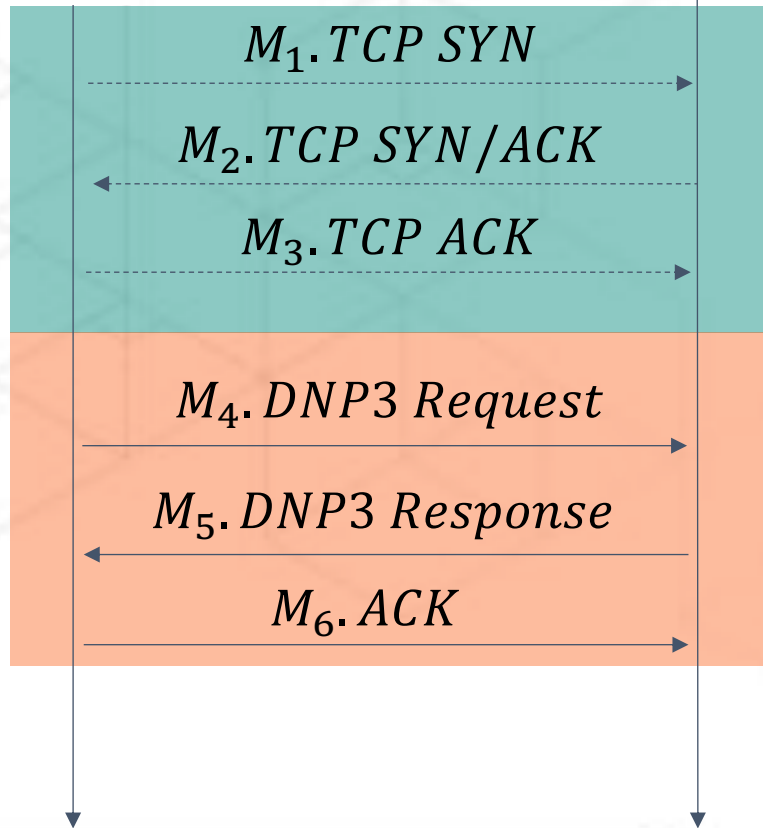
DNP3 Handshake Process



HMI

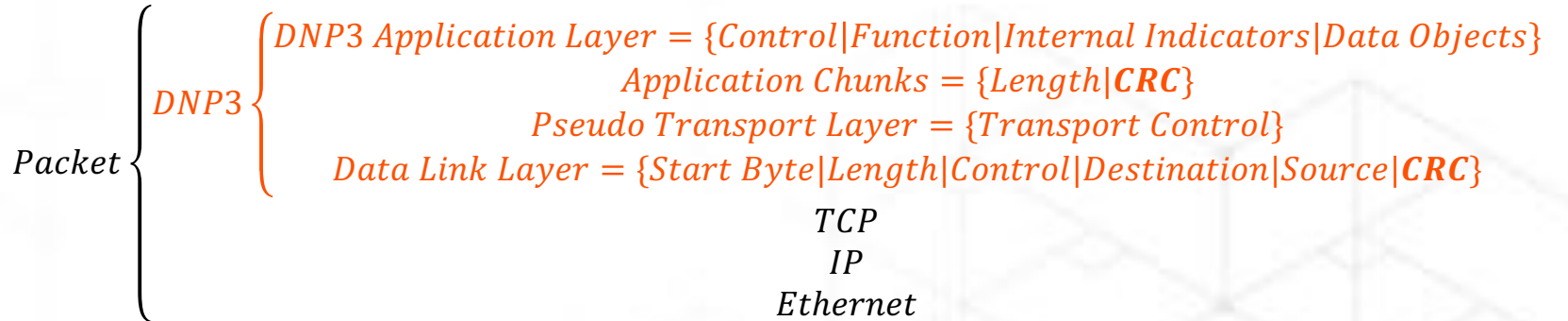


Outstation



TCP Connection Established

DNP3 Communication



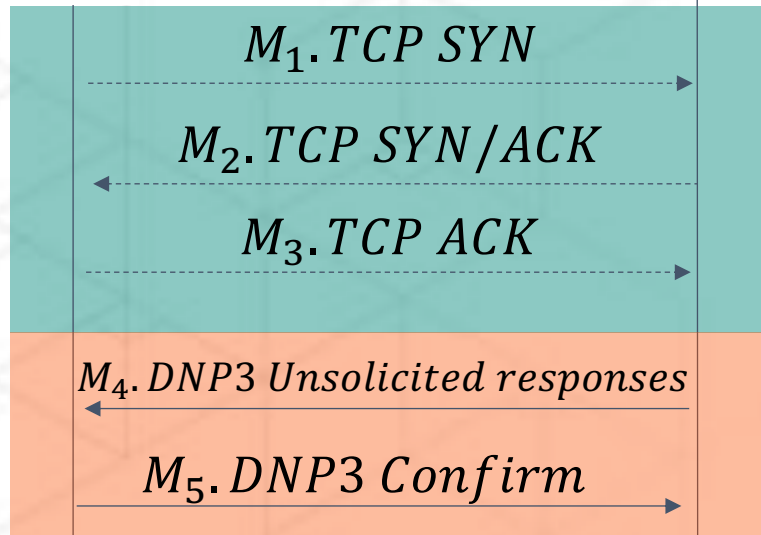
DNP3 Handshake Process (Unsolicited responses)



HMI



Outstation



M_4 .DNP3 Unsolicited responses

No.	Time	Source	Destination	Protocol	Length	Info
4	0.001247	192.168.60.130	192.168.60.1	DNP 3.0	71	Unsolicited Response
5	0.002143	192.168.60.1	192.168.60.130	DNP 3.0	69	Confirm


```
▶ Frame 5: 69 bytes on wire (552 bits), 69 bytes captured (552 bits)
▶ Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: VMware_24:3a:0a (00:0c:29:24:3a:0a)
▶ Internet Protocol Version 4, Src: 192.168.60.1, Dst: 192.168.60.130
▶ Transmission Control Protocol, Src Port: 50596, Dst Port: 20000, Seq: 1, Ack: 18, Len: 15
▼ Distributed Network Protocol 3.0
  ▶ Data Link Layer, Len: 8, From: 1, To: 10, DIR, PRM, Unconfirmed User Data
  ▶ Transport Control: 0xc0, Final, First(FIR, FIN, Sequence 0)
  ▶ Data Chunks
  ▶ [1 DNP 3.0 AL Fragment (2 bytes): #5(2)]
▼ Application Layer: (FIR, FIN, UNS, Sequence 0, Confirm)
  ▼ Application Control: 0xd0, First, Final, Unsolicited(FIR, FIN, UNS, Sequence 0)
    1... .... = First: Set
    .1.. .... = Final: Set
    ..0. .... = Confirm: Not set
    ...1 .... = Unsolicited: Set
    .... 0000 = Sequence: 0
  Function Code: Confirm (0x00)
```

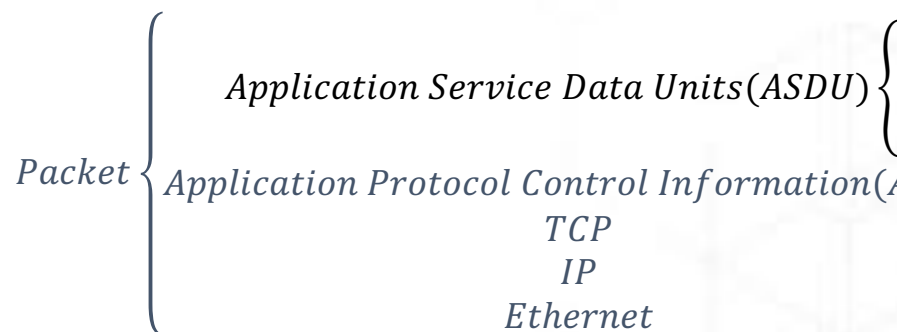
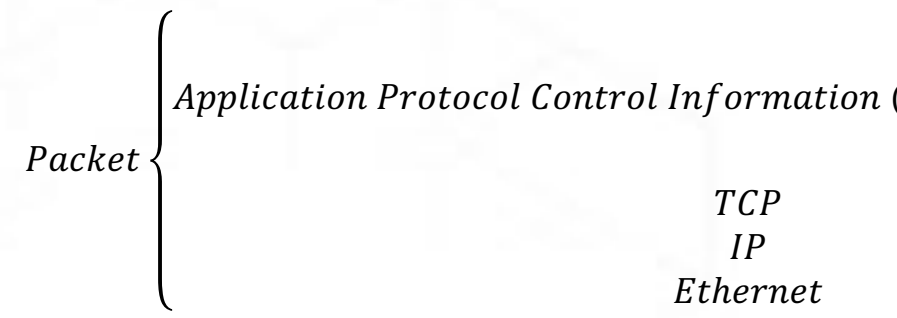
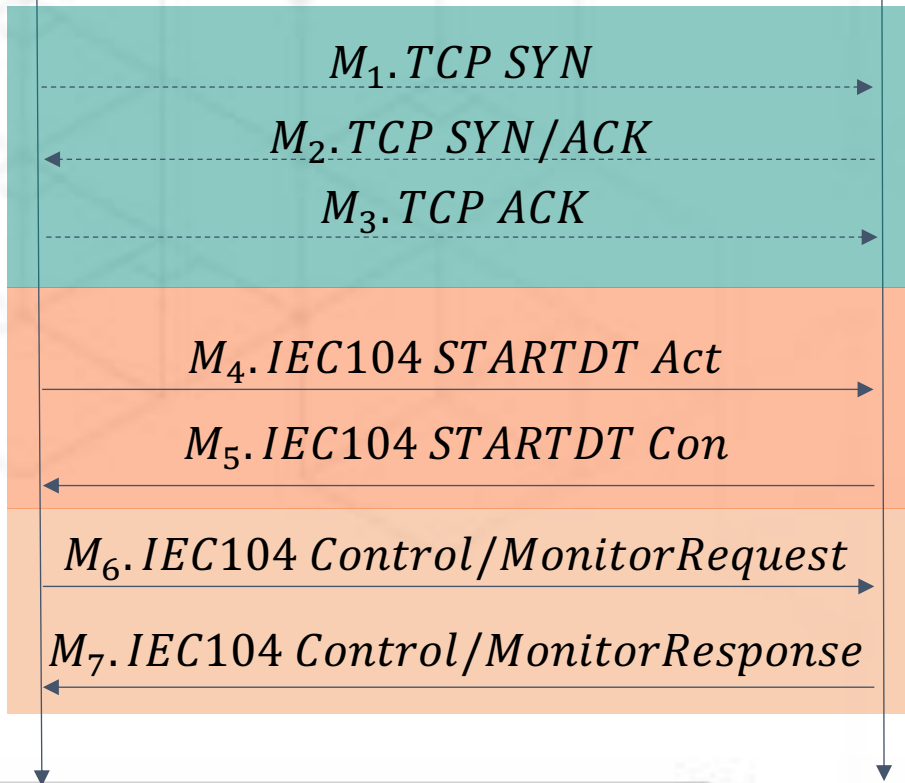
IEC 104 Handshake Process



HMI



RTU



8 bits		
Type Identification		0
S	Number of objects	
Q	Number of objects	
T	P/N	Cause of transmission (COT)
Start Byte (0x68)		Originator address (ORG)
Length of APDU		ASDU address fields (3 bytes)
Information object address (IOA) fields (3 bytes)		Information Elements
Information Elements		Time Tag
Information Object 2		Information Object N
Information Object N		

Application Protocol Data Unit = APCI + ASDU

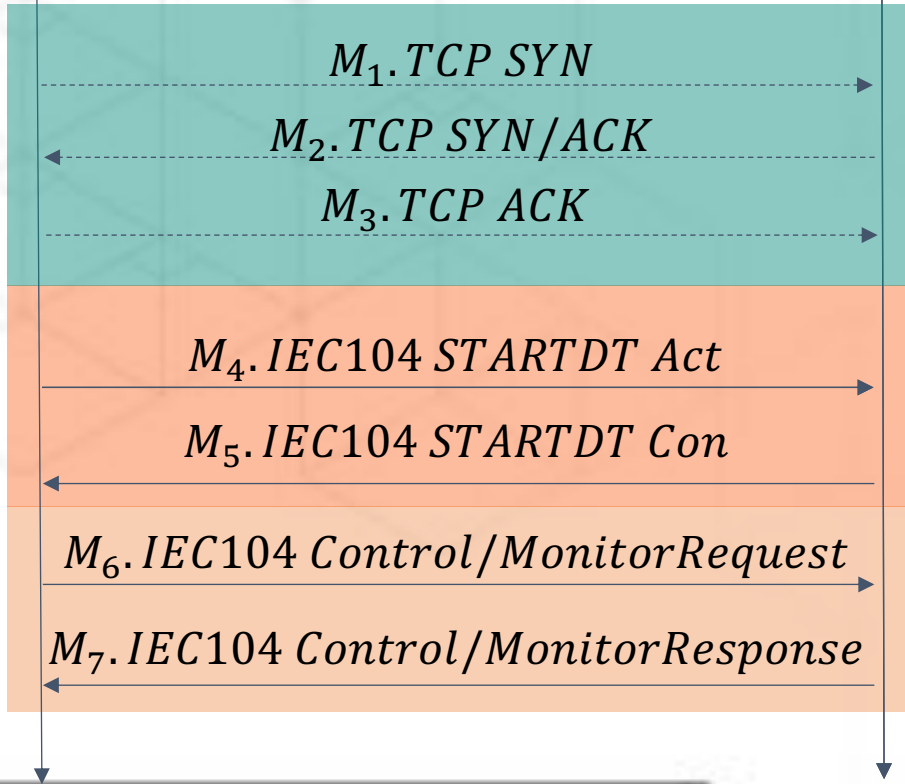
IEC 104 Handshake Process



HMI



RTU



M_4 . IEC104 Control/MonitorRequest
 M_7 . IEC104 Control/MonitorResponse

The image shows a Wireshark capture of the IEC 104 handshake process. The capture is filtered for the IP address 192.168.1.44. The table below shows the captured packets:

No.	Time	Source	Destination	Protocol	Length	Info
57	141.981..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (9,2) ASDU=37133 C_IC_NA_1 ActTerm IOA=0
59	142.522..	192.168.1.44	10.209.13.145	IEC 60870-5-104	66	66 ← S (8) ← S (10)
60	143.178..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	72	72 → I (10,2) ASDU=37133 M_ME_NB_1 Spont IOA=39999
62	150.236..	192.168.1.44	10.209.13.145	IEC 60870-5 ASDU	70	70 ← I (2,11) ASDU=37133 C_IC_NA_1 Act IOA=0
63	151.168..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (11,3) ASDU=37133 C_IC_NA_1 ActCon IOA=0
65	151.442..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	79	79 → I (12,3) ASDU=37133 M_SP_NA_1 Inrogen IOA[10]=10010-10019
67	151.613..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (13,3) ASDU=37133 M_DP_NA_1 Inrogen IOA=15000
68	151.669..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (14,3) ASDU=37133 C_IC_NA_1 ActTerm IOA=0
70	152.036..	192.168.1.44	10.209.13.145	IEC 60870-5-104	66	66 ← S (13) ← S (15)
71	152.575..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	72	72 → I (15,3) ASDU=37133 M_ME_NB_1 Spont IOA=39999
73	159.735..	192.168.1.44	10.209.13.145	IEC 60870-5 ASDU	70	70 ← I (3,16) ASDU=37133 C_IC_NA_1 Act IOA=0
74	160.395..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (16,4) ASDU=37133 C_IC_NA_1 ActCon IOA=0
76	160.620..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	79	79 → I (17,4) ASDU=37133 M_SP_NA_1 Inrogen IOA[10]=10010-10019
78	160.791..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (18,4) ASDU=37133 M_DP_NA_1 Inrogen IOA=15000
80	161.017..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	70	70 → I (19,4) ASDU=37133 C_IC_NA_1 ActTerm IOA=0
82	161.520..	192.168.1.44	10.209.13.145	IEC 60870-5-104	66	66 ← S (18) ← S (20)
83	162.150..	10.209.13.145	192.168.1.44	IEC 60870-5 ASDU	72	72 → I (20,4) ASDU=37133 M_ME_NB_1 Spont IOA=39999
85	169.233..	192.168.1.44	10.209.13.145	IEC 60870-5 ASDU	70	70 ← I (4,21) ASDU=37133 C_IC_NA_1 Act IOA=0

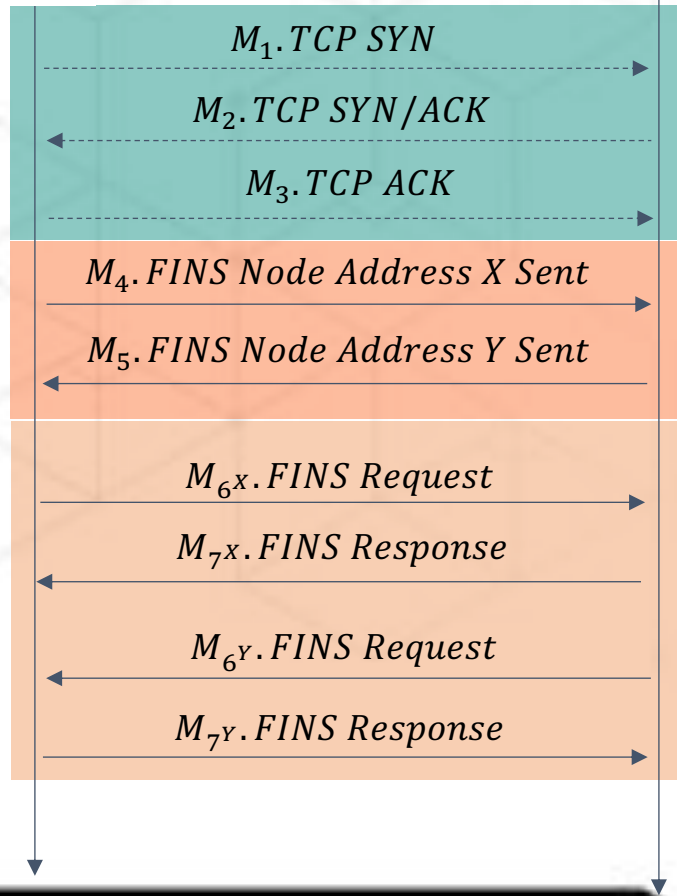
The packet details pane shows the following information for the selected packet (No. 74):

- Frame 74: 70 bytes on wire (560 bits), 70 bytes captured (560 bits)
- Ethernet II, Src: ARRISGro_48:5d:8c (08:1c:11:40:5d:8c), Dst: Dell_10:d4:1cc (08:15:c5:10:d4:1cc)
- Internet Protocol Version 4, Src: 10.209.13.145, Dst: 192.168.1.44
- Transmission Control Protocol, Src Port: 2404, Dst Port: 1099, Seq: 296, Ack: 113, Len: 16
- IEC 60870-5-104: → I (16,4)
- START
- AppLen: 14
- 0 = Type: I (0x00)
- Tx: 16
- Rx: 4
- IEC 60870-5-101/104 ASDU: ASDU=37133 C_IC_NA_1 ActCon IOA=0 'Interrogation command'
- TypeId: C_IC_NA_1 (100)
- R... .. = SQ: False
- ..00 0001 = NumIx: 1
- ..00 0111 = CauseTx: ActCon (7)
- ..0... .. = Negative: False
- R... .. = Test: False
- QA: 9
- Addr: 37133
- IOA: 0
- IOA: 0
- 000: Station interrogation (global) (20)

Function List

- Process information in monitor direction (21)
 - 0x0a (10) M_ME_TA_1: Measured value, normalized value with time tag
- Process telegrams with long time tag IO (11)
- Process information in control direction (7)
 - 0x2d (45) C_SC_NA_1 : Single command
- Command telegrams with long time tag (7)
- System information in monitor direction (1)
- System information in control direction (8)
 - 0x65 (101) C_CI_NA_1 : Counter interrogation command
- Parameter in control direction (4)
- File transfer (8)
 - 0x78 (120) F_FR_NA_1 : File ready

ORMON FINS Handshake Process



Connection Established

FINS Node Address Exchanged

Data Communication

Request

$$\begin{aligned}
 \text{FINS Frame} &= \langle \text{Request Command} | \text{Data} \rangle \\
 \text{FINS Header} &= \langle \text{ICF} | \text{RSV} | \text{GCT} | \text{DNA} | \text{DA1} | \text{DA2} | \text{SNA} | \text{SA1} | \text{SA2} | \text{SID} \rangle \\
 \text{FINS/TCP Header} &= \langle \text{Magic} | \text{Length} | \text{Command} | \text{Error Code} \rangle \\
 &\text{TCP/UDP} \\
 &\text{IP} \\
 &\text{Ethernet}
 \end{aligned}$$

FINS Header										FINS Frame	
ICF	RSV	GCT	DNA	DA1	DA2	SNA	SA1	SA2	SID	Request Command	Data

Response

$$\begin{aligned}
 \text{FINS Frame} &= \langle \text{Request Command} | \text{Response Code} | \text{Data} \rangle \\
 \text{FINS Header} &= \langle \text{ICF} | \text{RSV} | \text{GCT} | \text{DNA} | \text{DA1} | \text{DA2} | \text{SNA} | \text{SA1} | \text{SA2} | \text{SID} \rangle \\
 \text{FINS/TCP Header} &= \langle \text{Magic} | \text{Length} | \text{Command} | \text{Error Code} \rangle \\
 &\text{TCP/UDP} \\
 &\text{IP} \\
 &\text{Ethernet}
 \end{aligned}$$

FINS Header										FINS Frame		
ICF	RSV	GCT	DNA	DA1	DA2	SNA	SA1	SA2	SID	Request Command	Response Code	Data

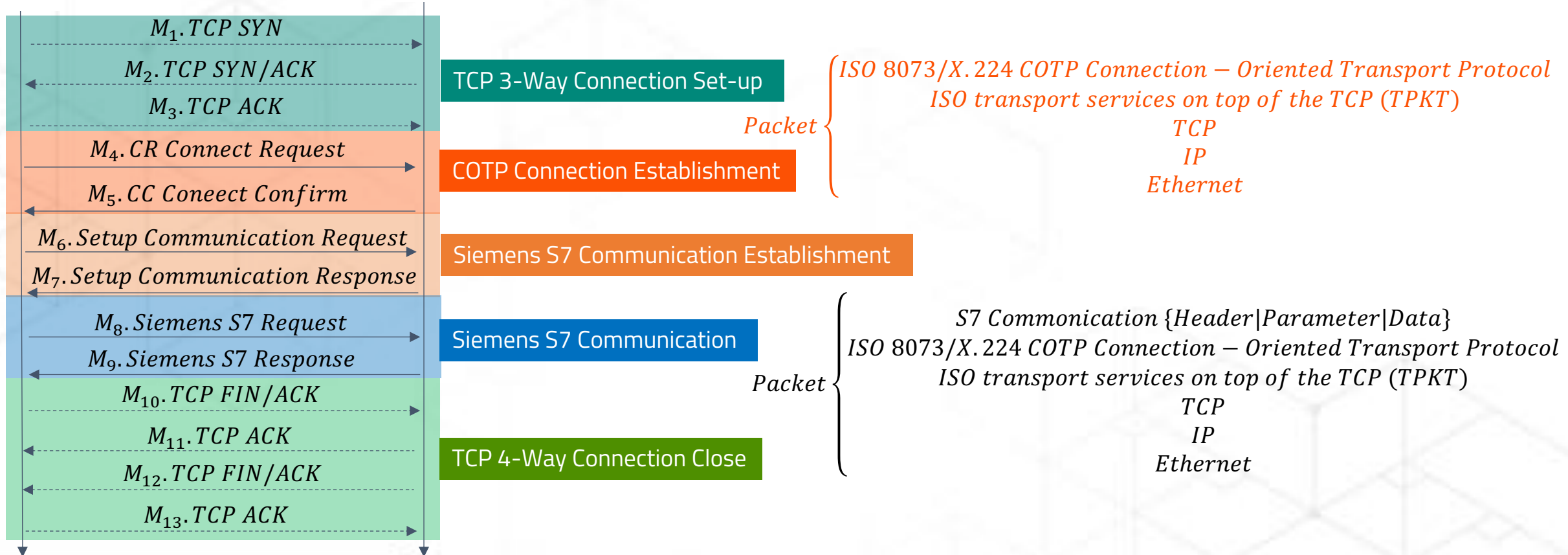
Siemens S7 Handshake Process



HMI



PLC



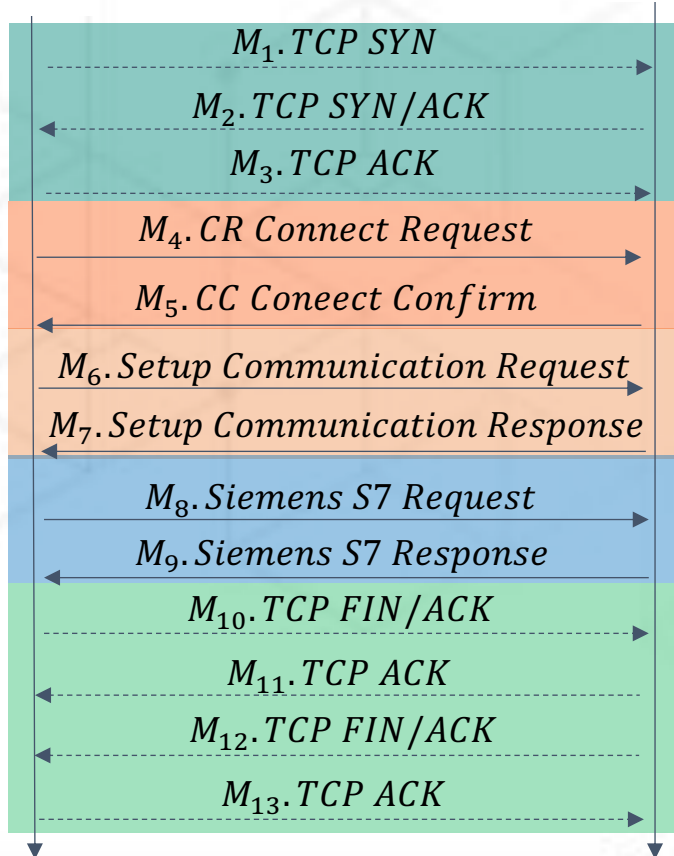
Siemens S7 Handshake Process



HMI



PLC



M₈. Siemens S7 Request

The image shows a Wireshark network traffic capture. The main pane displays a list of packets, with packet 13 highlighted in green. The packet list pane shows the following details for packet 13:

No.	Time	Source	Destination	Protocol	Length	Info
4	3.464907	192.168.1.10	192.168.1.40	TCP	66	4173 → 182 [SYN] Seq=0 Win=0 Len=0 MSS=1460 WS=256 SACK_PERM=1
5	3.466905	192.168.1.40	192.168.1.10	TCP	68	182 → 4173 [SYN, ACK] Seq=0 Ack=1 Win=0 Len=0 MSS=1460
6	3.466954	192.168.1.10	192.168.1.40	TCP	54	4173 → 182 [ACK] Seq=1 Win=64240 Len=0
7	3.466281	192.168.1.10	192.168.1.40	COTP	76	CR TPOU src-ref: 0x0000 dst-ref: 0x0000 Len=0
8	3.470052	192.168.1.40	192.168.1.10	COTP	76	CC TPOU src-ref: 0x0000 dst-ref: 0x0000
9	3.470268	192.168.1.10	192.168.1.40	S7COMM	79	R0SCTR: [Job] Function:[Setup communication]
10	3.473995	192.168.1.40	192.168.1.10	S7COMM	81	R0SCTR: [Ack_Data] Function:[Setup communication]
11	3.474004	192.168.1.10	192.168.1.40	COTP	61	DT TPOU (0) [COTP fragment, 0 bytes]
12	3.474160	192.168.1.10	192.168.1.40	S7COMM	87	R0SCTR: [Userdata] Function:[Request] → [CPU functions] → [Read SZL] ID=0
13	3.480118	192.168.1.40	192.168.1.10	S7COMM	135	R0SCTR: [Userdata] Function:[Response] → [CPU functions] → [Read SZL] ID=0
14	3.480207	192.168.1.10	192.168.1.40	COTP	61	DT TPOU (0) [COTP fragment, 0 bytes]
15	3.480577	192.168.1.10	192.168.1.40	COPYMM	87	RMSCTR: [DataTransfer] Function:[Response] → [CPU functions] → [Read CR] ID=0

The packet details pane for packet 13 shows the following structure:

- Frame 13: 135 bytes on wire (10800 bits), 135 bytes captured (10800 bits)
- Ethernet II, Src: Siemens_23:eb:3b (00:1b:1b:23:eb:3b), Dst: AsustekC_04:5e:41 (90:e6:ba:04:5e:41)
- Internet Protocol Version 4, Src: 192.168.1.40, Dst: 192.168.1.10
- Transmission Control Protocol, Src Port: 182, Dst Port: 4173, Seq: 50, Ack: 80, Len: 81
- TPKT, Version: 3, Length: 81
- ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
- S7 Communication
 - Header: (Userdata)
 - Protocol ID: 0x32
 - R0SCTR: Userdata (7)
 - Redundancy Identification (Reserved): 0x0000
 - Protocol Data Unit Reference: 768
 - Parameter length: 12
 - Data length: 52
 - Parameter: (Response) → [CPU functions] → [Read SZL]
 - Parameter head: 0x000132
 - Parameter length: 8
 - Method (Request/Response): Res (0x12)
 - 1000 = Type: Response (8)
 - 0200 = Function group: CPU functions (4)
 - Subfunction: Read SZL (1)
 - Sequence number: 1
 - Data unit reference number: 0
 - Last data unit: Yes (0x00)
 - Error code: No error (0x0000)
 - Data (SZL-ID: 0x0132, Index: 0x0004)
 - Return code: Success (0x00)
 - Transport size: OCTET STRING (0x09)
 - Length: 48
 - SZL-ID: 0x0132, Diagnostic type: CPU, Number of the partial list extract: Status data for one communication section of the CPU, section specified by index, No
 - SZL-Index: 0x0004 [Object management system status]
 - SZL partial list length in bytes: 48
 - SZL partial list count: 1
 - SZL data tree (list count no. 1)

Siemens S7 Plus Handshake Process

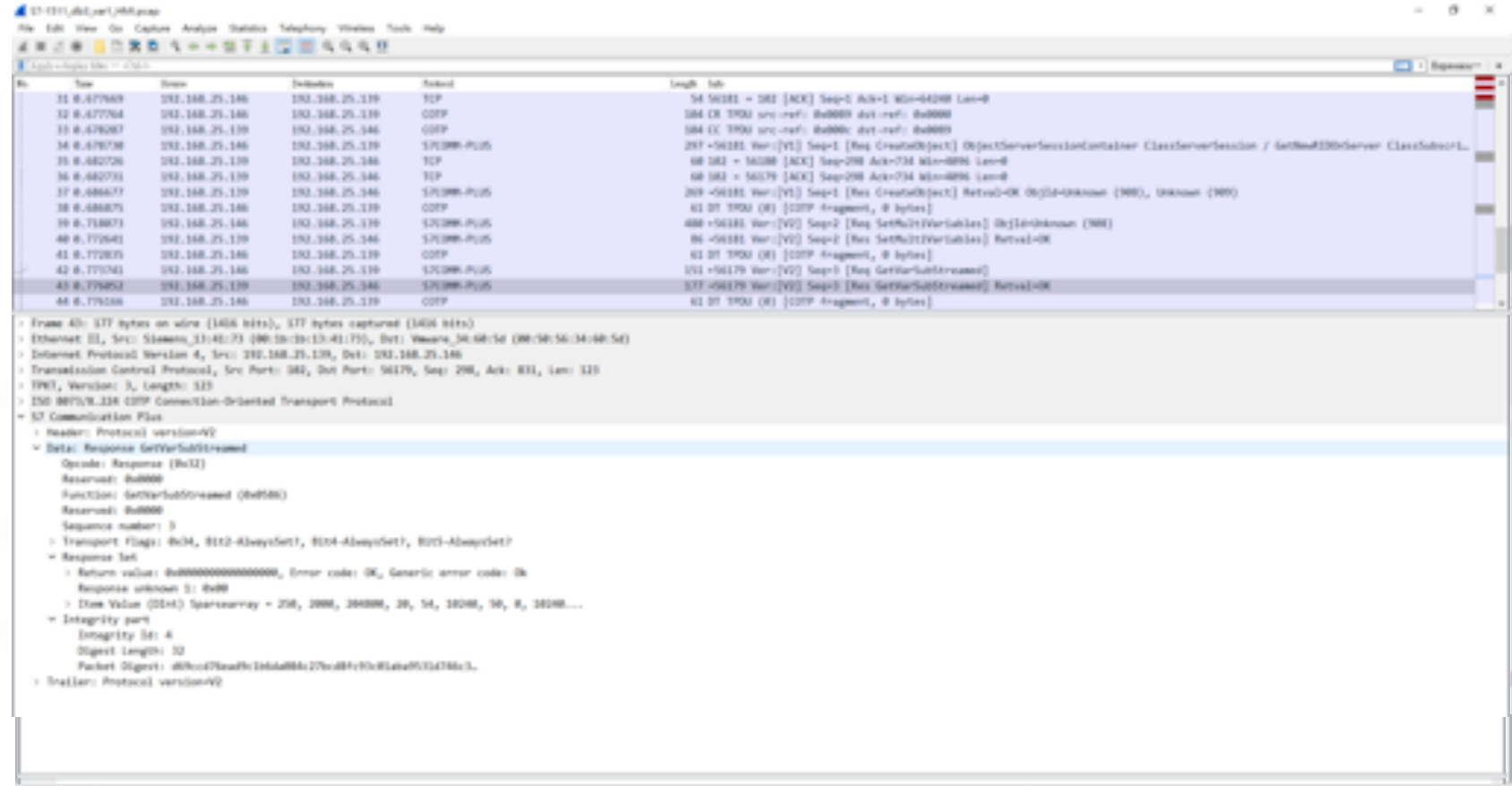
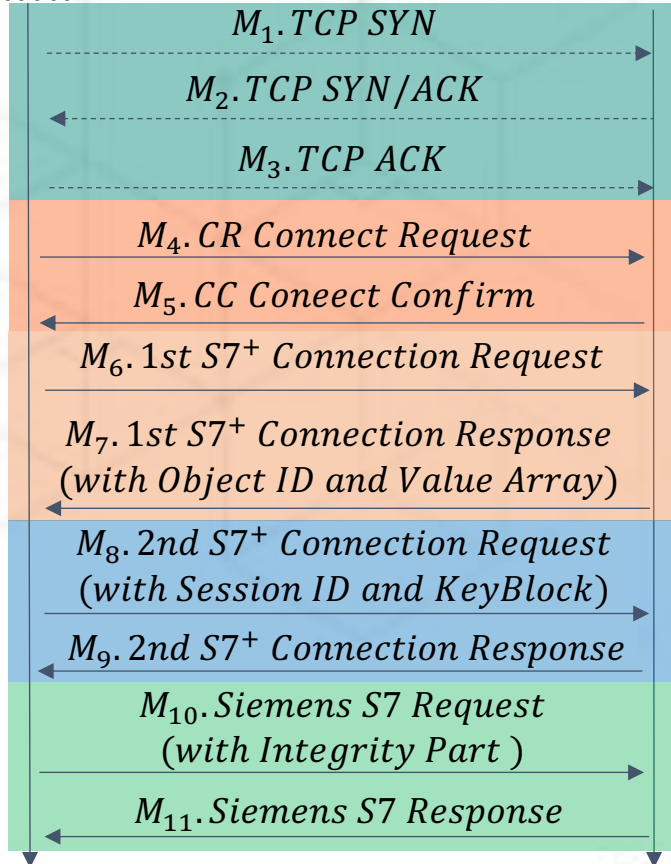


HMI



PLC

~~M₁. 1st S7+ Connection Request~~
~~M₂. 1st S7+ Connection Response~~
~~M₃. 2nd S7+ Connection Request~~
~~M₄. 2nd S7+ Connection Response~~



Siemens S7 Plus Version

V1

```
21 0.150399 192.168.1.191 192.168.1.35 COTP
22 0.151095 192.168.1.35 192.168.1.191 S7COMM-PLUS
23 0.207101 192.168.1.191 192.168.1.35 S7COMM-PLUS
24 0.207326 192.168.1.35 192.168.1.191 COTP
25 0.207608 192.168.1.35 192.168.1.191 S7COMM-PLUS
```

> Frame 22: 305 bytes on wire (2448 bits), 305 bytes captured (2448 bits)
> Ethernet II, Src: Vmware_44:2d:17 (00:0c:29:44:2d:17), Dst: SiemensN_08:e7:db (00:1c:06:08:e7:db)
> Internet Protocol Version 4, Src: 192.168.1.35, Dst: 192.168.1.191
> Transmission Control Protocol, Src Port: 49179, Dst Port: 102, Seq: 37, Ack: 37, Len: 251
> TPKT, Version: 3, Length: 251
> ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
S7 Communication Plus
 Header: Protocol version=V1
 Protocol Id: 0x72
 Protocol version: V1 (0x01)
 Data length: 236
 > Data: Request CreateObject
 > Trailer: Protocol version=V1

V3

```
1890 29.536238 10.24.103.251 10.24.103.200 S7COMM-PLUS
1891 29.536389 10.24.103.251 10.24.103.200 S7COMM-PLUS
1892 29.536413 10.24.103.200 10.24.103.251 TCP
1893 29.536512 10.24.103.251 10.24.103.200 TCP
```

Frame 1890: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits)
Ethernet II, Src: Vmware_a4:ca:98 (00:0c:29:a4:ca:98), Dst: LcfcHefe_d6:ee:43 (50:7b:9d:d6:ee:43)
Internet Protocol Version 4, Src: 10.24.103.251, Dst: 10.24.103.200
Transmission Control Protocol, Src Port: 46818, Dst Port: 102, Seq: 415738, Ack: 1, Len: 1448
[2 Reassembled TCP Segments (1882 bytes): #1888(1172), #1890(710)]
TPKT, Version: 3, Length: 1882
ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
S7 Communication Plus
 Header: Protocol version=V3
 Protocol Id: 0x72
 Protocol version: V3 (0x03)
 Data length: 1867
 Integrity part
 Digest Length: 32
 Packet Digest: 2e99d6b10d0581984adb5a684a2cb226771c0d173d03928d...
 > Data: Request GetMultiVariables
 > Trailer: Protocol version=V3

V2

```
42 53.710232 192.168.25.146 192.168.25.139 TCP
43 53.712034 192.168.25.139 192.168.25.146 S7COMM-PLUS
44 53.715816 192.168.25.146 192.168.25.139 COTP
45 53.715827 192.168.25.146 192.168.25.139 TCP
46 53.877113 192.168.25.139 192.168.25.146 TCP
```

Frame 43: 120 bytes on wire (960 bits), 120 bytes captured (960 bits)
Ethernet II, Src: Siemens_13:41:73 (00:1b:1b:13:41:73), Dst: Vmware_34:60:5d (00:50:56:34:60:5d)
Internet Protocol Version 4, Src: 192.168.25.139, Dst: 192.168.25.146
Transmission Control Protocol, Src Port: 102, Dst Port: 55863, Seq: 564, Ack: 1169, Len: 66
TPKT, Version: 3, Length: 66
ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
S7 Communication Plus
 Header: Protocol version=V2
 > Data: Response GetMultiVariables
 Opcode: Response (0x32)
 Reserved: 0x0000
 Function: GetMultiVariables (0x054c)
 Reserved: 0x0000
 Sequence number: 6
 > Transport flags: 0x34, Bit2-AlwaysSet?, Bit4-AlwaysSet?, Bit5-AlwaysSet?
 Response Set
 Integrity part
 Integrity Id: 10
 Digest Length: 32
 Packet Digest: c6bf255aaec1f182c3ee8fe37ca48ac577a080ae3a520112...
 > Trailer: Protocol version=V2

Wireshark can't analyze MELSEC Traffic

Apply a display filter: <X/ >

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.3.101	192.168.3.30	TCP	74	37602 → 5007 [SYN, Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=941583768 TSecr=0 ...
2	0.001294	192.168.3.30	192.168.3.101	TCP	60	5007 → 37602 [SYN, ACK] Seq=0 Ack=1 Min=11600 Len=0 MSS=1460
3	0.001498	192.168.3.101	192.168.3.30	TCP	60	37602 → 5007 [ACK] Seq=1 Ack=1 Win=29200 Len=0
4	0.001993	192.168.3.101	192.168.3.30	TCP	87	37602 → 5007 [PSH, ACK] Seq=1 Ack=1 Min=29200 Len=33
5	0.009271	192.168.3.30	192.168.3.101	TCP	77	5007 → 37602 [PSH, ACK] Seq=1 Ack=34 Win=11600 Len=23
6	0.009479	192.168.3.101	192.168.3.30	TCP	60	37602 → 5007 [ACK] Seq=34 Ack=24 Win=29200 Len=0
7	0.050138	192.168.3.101	192.168.3.30	TCP	60	37602 → 5007 [FIN, ACK] Seq=34 Ack=24 Win=29200 Len=0
8	0.061649	192.168.3.30	192.168.3.101	TCP	60	5007 → 37602 [ACK] Seq=24 Ack=35 Win=11600 Len=0
9	0.061688	192.168.3.30	192.168.3.101	TCP	60	5007 → 37602 [FIN, ACK] Seq=24 Ack=35 Win=11600 Len=0
10	0.062163	192.168.3.101	192.168.3.30	TCP	60	37602 → 5007 [ACK] Seq=35 Ack=25 Win=29200 Len=0

> Frame 4: 87 bytes on wire (696 bits), 87 bytes captured (696 bits)
> Ethernet II, Src: Vmware_eb:fb:fa (00:0c:29:ec:fb:fa), Dst: Mitsubis_99:23:c0 (38:e0:8e:99:23:c0)
> Internet Protocol Version 4, Src: 192.168.3.101, Dst: 192.168.3.30
> Transmission Control Protocol, Src Port: 37602, Dst Port: 5007, Seq: 1, Ack: 1, Len: 33
> Data (33 bytes)

```
0000 38 e0 8e 99 23 c0 00 0c 29 ec fb fa 00 00 45 00  8---#---}-----E-
0010 00 49 36 83 40 00 40 06 7c 58 c0 a0 03 65 c0 a0  -D-q@-|X---e-
0020 83 1e 92 e2 13 8f 47 a9 55 21 00 0c 1e 3b 50 18  -----G-Ut---P-
0030 72 10 ce 66 00 00 54 00 34 12 00 00 00 ff ff 83  r--f--T-4-----
0040 00 14 00 0a 00 03 04 00 00 02 01 00 00 00 a8 00  -----
```

Again, we built a Lua Plugin for MELSEC

```
m_protocol.lua
Applications > Wireshark.app > Contents > Plugins > wireshark > m_protocol.lua
1
2 mc_proto = Proto("mc_proto", "MELSEC", "MELSEC")
3
4
5 BIN_3E_REQ_KEY = 0x5000
6 BIN_3E_REQ_KEY = 0x5000
7 BIN_4E_REQ_KEY = 0x5000
8 BIN_4E_REQ_KEY = 0x5000
9
10 ASCII_3E_REQ_KEY = 0x35303030 ---"0000"
11 ASCII_3E_REQ_KEY = 0x04030303 ---"0000"
12 ASCII_4E_REQ_KEY = 0x35343030 ---"0000"
13 ASCII_4E_REQ_KEY = 0x04043030 ---"0000"
14
15 -- Binary
16 local mc_magic_bin = ProtoField.uint16("mc_proto_magic_bin", "Magic No.", base_HEX)
17 local mc_seq_bin = ProtoField.uint16("mc_proto_seq_bin", "Sequence No.", base_HEX_DEC)
18 local mc_rsvl_bin = ProtoField.uint16("mc_proto_rsvl_bin", "Reserved", base_HEX)
19 local mc_net_no_bin = ProtoField.uint8("mc_proto_net_bin", "Net No.", base_HEX)
20 local mc_node_no_bin = ProtoField.uint8("mc_proto_node_bin", "Node No.", base_HEX)
21 local mc_dst_proc_no_bin = ProtoField.uint16("mc_proto_dst_proc_bin", "Dst Proc No.", base_HEX)
22 local mc_dst_mod_sta_no_bin = ProtoField.uint8("mc_proto_dst_mod_sta_bin", "Dst Sta No.", base_HEX)
23 local mc_len_bin = ProtoField.uint16("mc_proto_len_bin", "Data Len", base_HEX_DEC)
24 local mc_timer_bin = ProtoField.uint16("mc_proto_timer_bin", "Timer", base_HEX_DEC)
25 local mc_encode_bin = ProtoField.uint16("mc_proto_encode_bin", "End Code", base_HEX)
26 local mc_cmd_bin = ProtoField.uint16("mc_proto_cmd_bin", "Command", base_HEX)
27 local mc_subcmd_bin = ProtoField.uint16("mc_proto_subcmd_bin", "Sub-Command", base_HEX)
28
29 -- Request data for command 0403, 0402, 0500
30 local mc_num_word_bin = ProtoField.uint8("mc_proto_word_bin", "Number of word access points", base_HEX)
31 local mc_num_double_word_bin = ProtoField.uint8("mc_proto_double_word_bin", "Number of double word access points", base_HEX)
32 local mc_num_bit_bin = ProtoField.uint8("mc_proto_bit_bin", "Number of bit access points", base_HEX)
33 local mc_device_num_bin = ProtoField.uint24("mc_proto_device_num_bin", "Device number", base_HEX)
34 local mc_set_reset_bin = ProtoField.uint8("mc_proto_set_reset_bin", "Set/Reset", base_HEX)
35
36 -- Request data for command 3402
37
38 local mc_write_word_data_bin = ProtoField.uint16("mc_proto_write_word_data_bin", "Write word data", base_HEX)
39 local mc_write_double_word_data_bin = ProtoField.uint32("mc_proto_write_double_word_data_bin", "Write double word data", base_HEX)
40
41 -- Request data for command 3402, 3401
42 local mc_device_code_bin = ProtoField.uint8("mc_proto_device_code_bin", "Device code", base_HEX)
43
44
45 -- Request data for command 3530
46
47
48 local mc_remote_pass_len_bin = ProtoField.uint16("mc_proto_remote_pass_len_bin", "Remote password length", base_HEX)
49 local mc_remote_pass_bin = ProtoField.string("mc_proto_remote_pass_bin", "Remote password", base_NONE)
50
51 -- Request data for command 0x5000
52 local mc_head_dev_num_bin = ProtoField.uint24("mc_proto_head_dev_num_bin", "Head device number", base_HEX_DEC)
53 local mc_num_dev_points_bin = ProtoField.uint16("mc_proto_num_dev_points_bin", "Number of device points", base_HEX_DEC)
54 local mc_write_data_bin = ProtoField.bytes("mc_proto_write_data_bin", "Write data", base_NONE)
```

The image shows a Wireshark network traffic capture. The top pane displays a list of captured packets. Packet 4 is highlighted, showing a MELSEC frame. The bottom pane shows the detailed view of this frame, including the sub-header and request data.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.3.101	192.168.3.30	TCP	74	37600 → 5007 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=941581
2	0.001052	192.168.3.30	192.168.3.101	TCP	60	5007 → 37600 [SYN, ACK] Seq=0 Ack=1 Win=11688 Len=0 MSS=1460
3	0.001546	192.168.3.101	192.168.3.30	TCP	60	37600 → 5007 [ACK] Seq=1 Ack=1 Win=29200 Len=0
4	0.001968	192.168.3.101	192.168.3.30	MELSEC	83	Melsec 3E Binary Request
5	0.008123	192.168.3.30	192.168.3.101	MELSEC	73	Melsec 3E Binary Response
6	0.008270	192.168.3.101	192.168.3.30	TCP	60	37600 → 5007 [ACK] Seq=30 Ack=20 Win=29200 Len=0

Frame 4: 83 bytes on wire (664 bits), 83 bytes captured (664 bits)
Ethernet II, Src: Vmware_ec:fb:fa (00:0c:29:ec:fb:fa), Dst: Mitsubisi_99:23:c0 (38:e0:0e:99:23:c0)
Internet Protocol Version 4, Src: 192.168.3.101, Dst: 192.168.3.30
Transmission Control Protocol, Src Port: 37600, Dst Port: 5007, Seq: 1, Ack: 1, Len: 29
3E Binary Request
Sub Header
Data Len: 0x0014 (20)
Timer: 0x000a (10)
Command: 0x0403 (Random Read Device)
Sub-Command: 0x0000
Request Data: 0201000000000000000000000000000000
Number of word access points: 0x02
Number of double word access points: 0x01
Device number: 0x000000
Device code: 0xa8
Device number: 0x000000
Device code: 0xa8
Device number: 0x000000
Device code: 0xa8

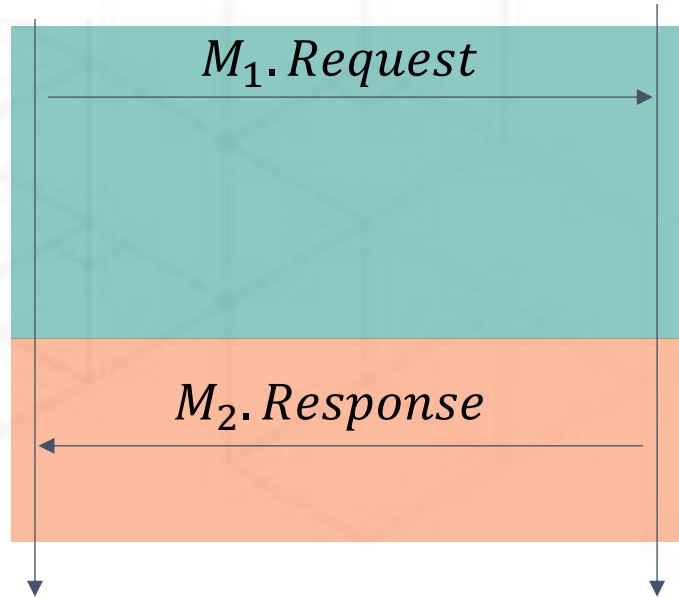
Mitsubishi Melsec/TCP Handshake Process



HMI



PLC



Request {
 $MELSEC = \langle Sub - Header | Access Route | Request Data Length | Timer | Request Data \rangle$
TCP
IP
Ethernet

Response {
 $MELSEC = \langle Sub - Header | Access Route | Response Data Length | End Code | Response Data \rangle$
TCP
IP
Ethernet

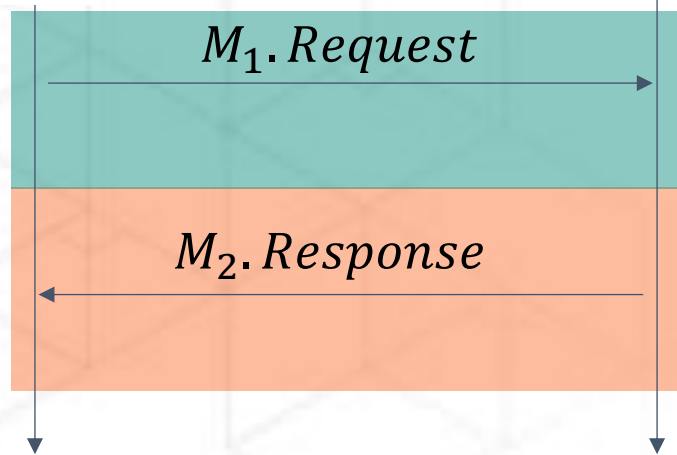
Mitsubishi Melsec/TCP Handshake Process



HMI



PLC



M₁.Request

M₂.Response

Packet 4: MELSEC 87 Melsec 4E Binary Request

Packet 5: MELSEC 77 Melsec 4E Binary Response

```
>> Frame 5: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface 0
>> Ethernet II, Src: Mitsubis_99:23:c0 (38:e0:0e:99:23:c0), Dst: Vmware_ec:fc:ba (08:0c:29:ec:fc:ba)
>> Internet Protocol Version 4, Src: 192.168.3.30, Dst: 192.168.3.101
>> Transmission Control Protocol, Src Port: 5007, Dst Port: 37682, Seq: 1, Ack: 34, Len: 23
>> 4E Binary Response
  >> Sub Header
    Magic No.: 0x0400
    Sequence No.: 0x1234 (4660)
    Reserved: 0x0000
    Net No.: 0x00
    Node No.: 0xff
    Dst Proc No.: 0x03ff
    Dst Sta No.: 0x00
  >> Data Len: 0x000a (10)
  End Code: 0x0000
  >> Response Data: 05000100000000e0
    Data read in word units: 0x0005
    Data read in word units: 0x0001
    Data read in double word units: 0x000e000d
```

0030 2d a0 8f cc 00 00 d4 00 34 12 00 00 00 ff ff 03 4
0040 00 05 00 00 05 00 01 00 0d 00 0e 00



Common Flaws in ICS Network Protocols

Insecure by Design

Type	Protocols	Handshake	Authentication	Message Encryption	
Public	Modbus/TCP	TCP Connection	×	×	
	DNP3/TCP	TCP Connection	×	×	
	EtherNetIP/CIP	ENIP Connection based	×	×	
	IEC104	TCP Connection + STARTDT	×	×	
Private	Melsec/TCP	TCP Connection	×	×	
	OMRON FINS/TCP	TCP Connection + FINS/TCP session based	×	×	
	S7COMM	TCP Connection + COTP + S7COMM Session	×	△(when EWS compile PLC program)	
	S7COMM Plus	TCP Connection + COTP + S7COMM+ Session	V1	×	×
			V2	×	√(HMAC-SHA256)
V3			√ (EWS <-> PLC)	√(HMAC-SHA256)	

Attacks on ICS Protocols

? Unknown

Type	Protocols	T814 Denial-of-Service	T836 Modify Parameter	T856 Spoof Reporting Message	T843 Program Download	T855 Unauthorized Command Message	
Public	Modbus/TCP	✓	✓	✓	?	✓	
	DNP3/TCP	✓	✓	✓	?	✓	
	EtherNetIP/CIP	?	✓	✓	✓	✓	
	IEC104	✓	✓	✓	?	✓	
Private	Melsec/TCP	?	✓	✓	✓	✓	
	OMRON FINS/TCP	?	✓	✓	✓	✓	
	S7COMM	✓	✓	✓	✓	✓	
	S7COMM Plus	V1	?	✓	?	✓	✓
		V2	?	✓	?	✓	✓
		V3	?	✓	?	✓	✓



ICS ATT&CK Matrix map to ICS Protocols Attack

Initial Access	Execution	Persistence	Evasion	Discovery	Lateral Movement	Collection	Command and Control	Inhibit Response Function	Impair Process Control	Impact
Data Historian Compromise	Change Program State	Hooking	Exploitation for Evasion	Control Device Identification	Default Credentials	Automated Collection	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Drive-by Compromise	Command-Line Interface	Module Firmware	Indicator Removal on Host	I/O Module Discovery	Exploitation of Remote Services	Data from Information Repositories	Connection Proxy	Alarm Suppression	Change Program State	Denial of Control
Engineering Workstation Compromise	Execution through API	Program Download	Masquerading	Network Connection Enumeration	External Remote Services	Detect Operating Mode	Standard Application Layer Protocol	Block Command Message	Masquerading	Denial of View
Exploit Public-Facing Application	Graphical User Interface	Project File	Rogue Master Device	Network Service Scanning	Program Organization Units	Detect Program State		Block Reporting Message	Modify Control Logic	Loss of Availability
External Remote Services	Man in the Middle	System Firmware	Rootkit	Network Sniffing	Remote File Copy	I/O Image	Standard Application Layer Protocol	Block Serial Comm Port	Modify Parameter	Loss of Control
Internet Accessible Device	Program Organization Units	Valid Accounts	Spoof Reporting Message	Remote System Discovery	Valid Accounts	Location Identification		Data Destruction	Module Firmware	Loss of Productivity and Revenue
Replication Through Removable Media	Project File		Utilize/Change Operating Mode	Serial Connection Enumeration		Monitor Process State		Denial of Service	Program Download	Loss of Safety
Spearphishing Attachment	Scripting					Point & Tag Identification		Device Restart/Shutdown	Rogue Master Device	Loss of View
Supply Chain Compromise	User Execution					Program Upload		Manipulate I/O Image	Service Stop	Manipulation of Control
Wireless Compromise						Role Identification		Modify Alarm Settings	Spoof Reporting Message	Manipulation of View
						Screen Capture		Modify Control Logic	Unauthorized Command Message	Theft of Operational Information
								Program Download		
								Rootkit		
								System Firmware		
							Utilize/Change Operating Mode			

11 Tactics
81 Techniques

The image shows a large industrial complex, possibly a refinery or chemical plant, silhouetted against a warm, orange-hued sky at sunset or sunrise. The facility consists of several tall, cylindrical distillation columns or towers, interconnected by a network of pipes and walkways. The lighting is dramatic, with the structures appearing as dark shapes against the bright, glowing background. The overall atmosphere is industrial and serene.

Demo Time



T836-Modify Parameter with Mitsubishi Melsec

T836-Modify Parameter

```
Internet Protocol Version 4, Src: 192.168.3.87, Dst: 192.168.3.39
Transmission Control Protocol, Src Port: 56486, Dst Port: 7788, Seq: 1, Ack: 1, Len: 23
3E Binary Request
  Sub Header
    Data Len: 0x000e (14)
    Timer: 0x000a (10)
    Command: 0x1401 (Batch Write Device)
    Sub-Command: 0x0000
  Request Data: 640000a801000a00
    Head device number: 0x000064 (100)
    Device code: 0xa8
    Number of device points: 0x0001 (1)
  Write data: 0a00
```

10s

```
Internet Protocol Version 4, Src: 192.168.3.87, Dst: 192.168.3.39
Transmission Control Protocol, Src Port: 56497, Dst Port: 7788, Seq: 1, Ack: 1, Len: 23
3E Binary Request
  Sub Header
    Data Len: 0x000e (14)
    Timer: 0x000a (10)
    Command: 0x1401 (Batch Write Device)
    Sub-Command: 0x0000
  Request Data: 640000a801000200
    Head device number: 0x000064 (100)
    Device code: 0xa8
    Number of device points: 0x0001 (1)
  Write data: 0200
```

2s

```
Internet Protocol Version 4, Src: 192.168.3.87, Dst: 192.168.3.39
Transmission Control Protocol, Src Port: 56510, Dst Port: 7788, Seq: 1, Ack: 1, Len: 23
3E Binary Request
  Sub Header
    Data Len: 0x000e (14)
    Timer: 0x000a (10)
    Command: 0x1401 (Batch Write Device)
    Sub-Command: 0x0000
  Request Data: 640000a801000800
    Head device number: 0x000064 (100)
    Device code: 0xa8
    Number of device points: 0x0001 (1)
  Write data: 0800
```

8s

```
Internet Protocol Version 4, Src: 192.168.3.87, Dst: 192.168.3.39
Transmission Control Protocol, Src Port: 56521, Dst Port: 7788, Seq: 1, Ack: 1, Len: 23
3E Binary Request
  Sub Header
    Data Len: 0x000e (14)
    Timer: 0x000a (10)
    Command: 0x1401 (Batch Write Device)
    Sub-Command: 0x0000
  Request Data: 640000a801001e00
    Head device number: 0x000064 (100)
    Device code: 0xa8
    Number of device points: 0x0001 (1)
  Write data: 1e00
```

30s

lab_pic_command_injection.pcap

Source	Destination	Protocol	Length	Info
192.168.3.87	192.168.3.39	M_Protocol	77	M_Protocol [0] 3E Binary Request
192.168.3.39	192.168.3.87	M_Protocol	65	M_Protocol [0] 3E Binary Request
192.168.3.87	192.168.3.39	M_Protocol	77	M_Protocol [0] 3E Binary Request
192.168.3.39	192.168.3.87	M_Protocol	65	M_Protocol [0] 3E Binary Request
192.168.3.87	192.168.3.39	M_Protocol	77	M_Protocol [0] 3E Binary Request
192.168.3.39	192.168.3.87	M_Protocol	65	M_Protocol [0] 3E Binary Request
192.168.3.87	192.168.3.39	M_Protocol	77	M_Protocol [0] 3E Binary Request
192.168.3.39	192.168.3.87	M_Protocol	65	M_Protocol [0] 3E Binary Request
192.168.3.87	192.168.3.39	M_Protocol	77	M_Protocol [0] 3E Binary Request
192.168.3.39	192.168.3.87	M_Protocol	65	M_Protocol [0] 3E Binary Request

mc_proto

```
Sub Header
Data Len: 0x000e (14)
Timer: 0x000a (10)
Command: 0x1401 (Batch Write Device)
Sub-Command: 0x0000
Request Data: 640000a801000a00
  Head device number: 0x000064 (100)
  Device code: 0xa8
  Number of device points: 0x0001 (1)
Write data: 0a00
```

```
0020 03 27 dc a6 1e 6c 73 f8 d6 09 00 34 42 e8 50 18  .'. .ls. .4B-P.
0030 ff ff 93 29 00 00 50 00 00 ff ff 03 00 0e 00 0a  ..)..P. ....
0040 00 01 14 00 00 64 00 00 a8 01 00 0a 00  .d.....
```

Write data (mc_proto.write_data_bin), 2 bytes

Packets: 40 - Displayed: 8 (20.0%)

3E Binary Request

3E Binary Request

3E Binary Request

3E Binary Request

3E Binary Request



T843-Program Download with Mitsubishi Melsoft



T856-Spoof Reporting Message with Modbus/TCP



T855-Unauthorized Command Message with Omron FINS

Common Flaws in ICS Protocols

No Authentication



No Authorization



No Encryption



Stack Overflow



T880-Loss of Safety

T829-Loss of View

T827-Loss of Control

T826-Loss of Availability

T815-Denial of View

Impact

T813-Denial of Control

T832-Manipulation of View

T831-Manipulation of Control

The background of the image shows a large industrial facility, possibly a refinery or chemical plant, silhouetted against a warm, orange-hued sky at sunset or sunrise. The facility consists of several tall, cylindrical distillation columns or towers, interconnected by a complex network of pipes, walkways, and structural steel. The lighting is dramatic, with the sky being a uniform, bright orange, and the industrial structures appearing as dark, intricate shapes. The overall mood is industrial and somewhat somber due to the low light.

How to Defend Against ICS Network Protocol Attacks

Vulnerable OT Environment

1 Shadow OT

- Unknown devices and unknown connections

2 Insecure Authentication

- Flaws come from design or implementation

3 Insecure Protocols

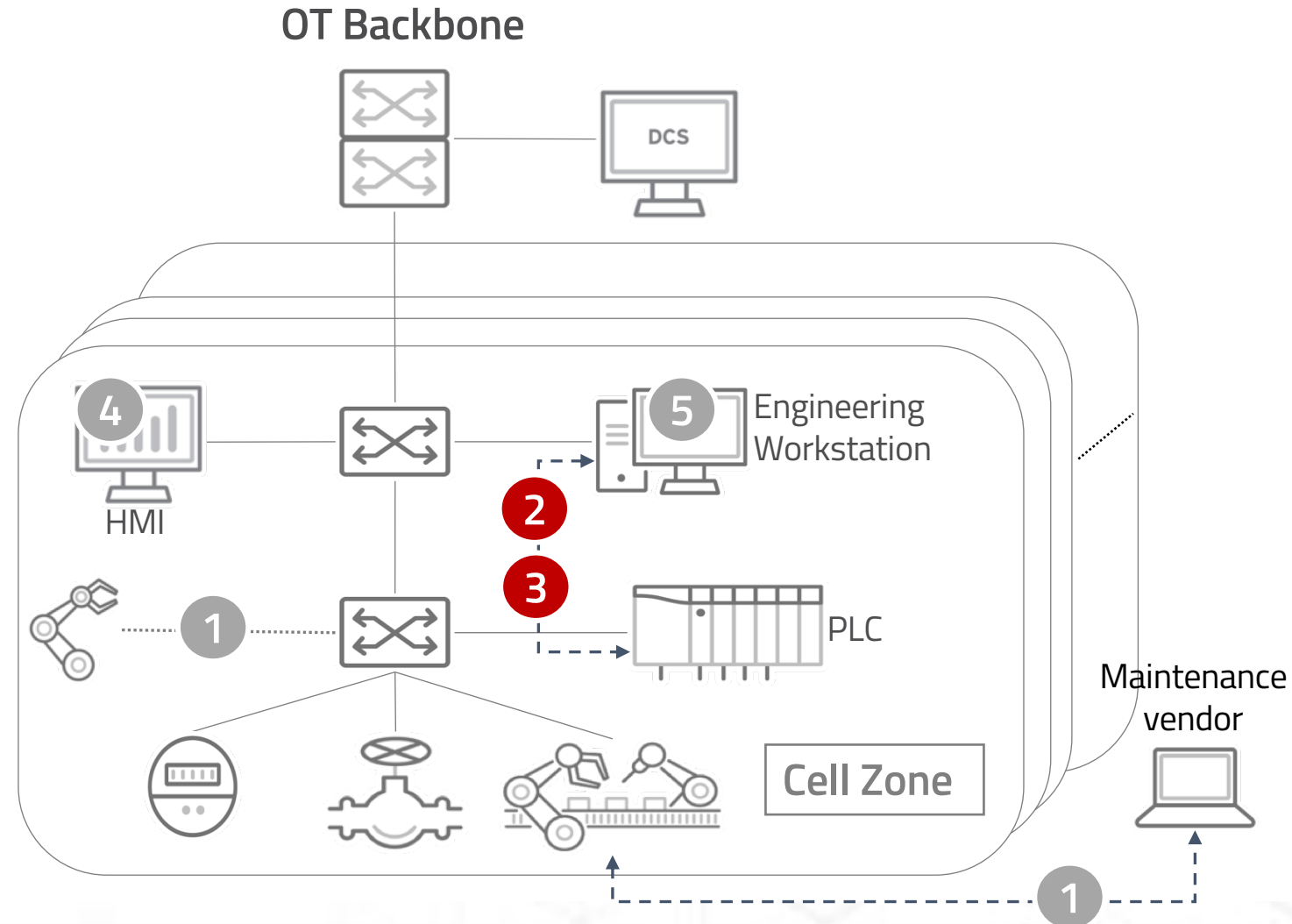
- Simply unencrypted

4 Unpatched Devices

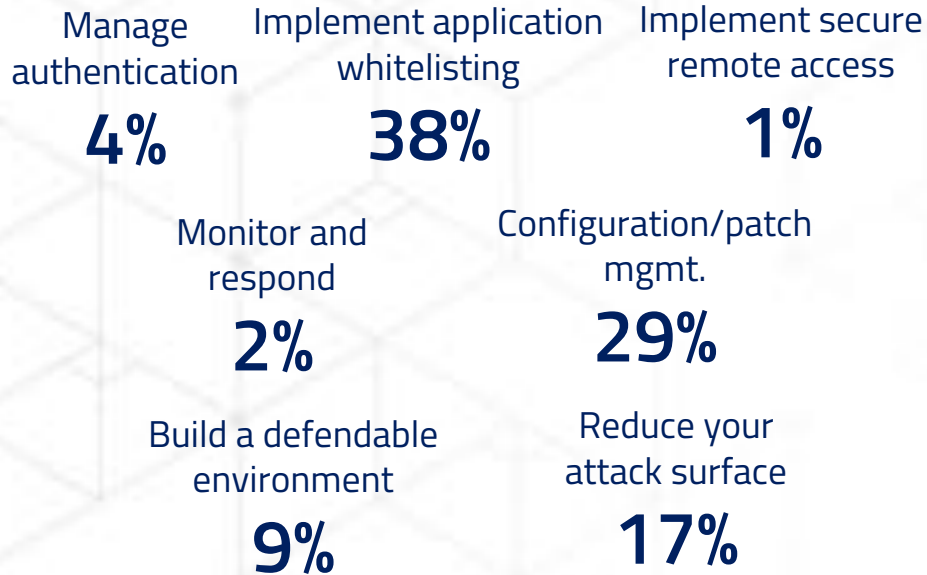
- Patching is not feasible or available

5 Insecure 3rd-Party Software

- Vulnerable, and might be compromised from supply chain in the beginning



Suggested Strategies from ICS CERT

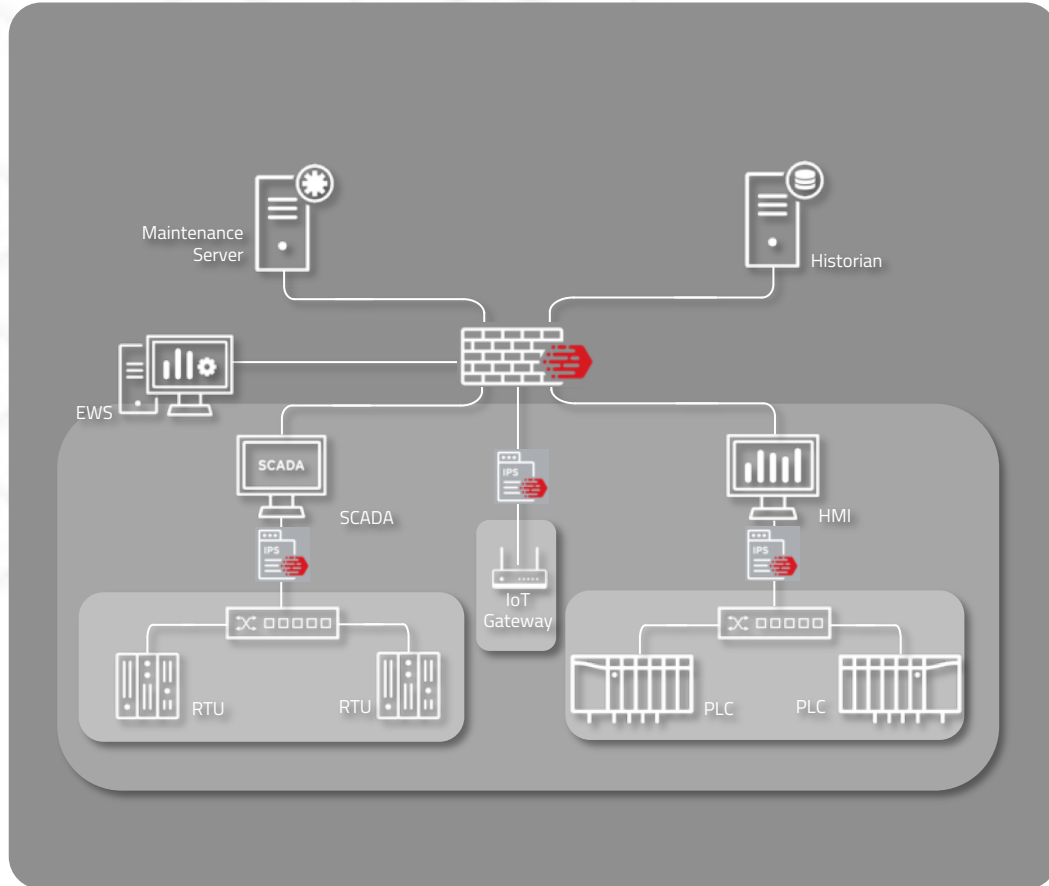


Incidents responded by ICS-CERT: https://ics-cert.us-cert.gov/sites/default/files/documents/Seven%20Steps%20to%20Effectively%20Defend%20Industrial%20Control%20Systems_S508C.pdf



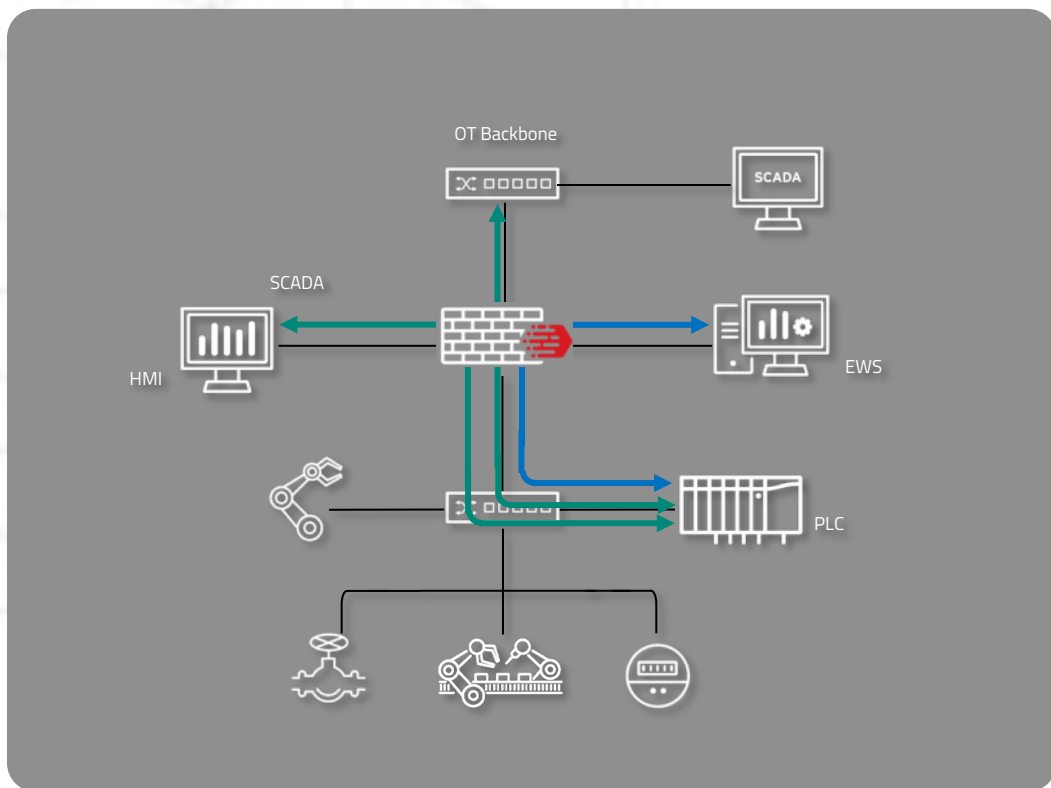
Implementing **FIVE** Tactics to prevent 98% incidents

Effective Segmentation, Virtual Patch, Containment



- Divide a big flat L2 network into secured segments
- Virtual Patch (IPS)
 - Containment of malware and worms
 - Shield device vulnerabilities
 - Deeply inspect IT protocols: SMB, RDP, ...
- Industrial-Grade Hardware

Granular Control Over Popular OT Protocols



↔ Read Only ↔ Full Access

- Asset and protocol visibility
- Fine-grained access control in different levels
 - Devices
 - Protocols (Modbus, Melsec/SLMP, CC-Link IE, Ethernet/IP, Profinet, S7COMM, HSMS/SECS-II, ...)
 - Control Commands (read, configure, shutdown, ...)
- Greatly lower the possibility of Denial-of-Service attacks by OT trojans



Network Whitelisting Control against Siemens S7 attack

AtCP



Thank You!

HITBLOCKDOWN⁰⁰²
livestream

Mars Cheng and Selmon Yang, mars_cheng@txone-networks.com