RSAConference2020
Asia Pacific & Japan

A Virtual Learning Experience | 15–17 July

HUMAN ELEMENT

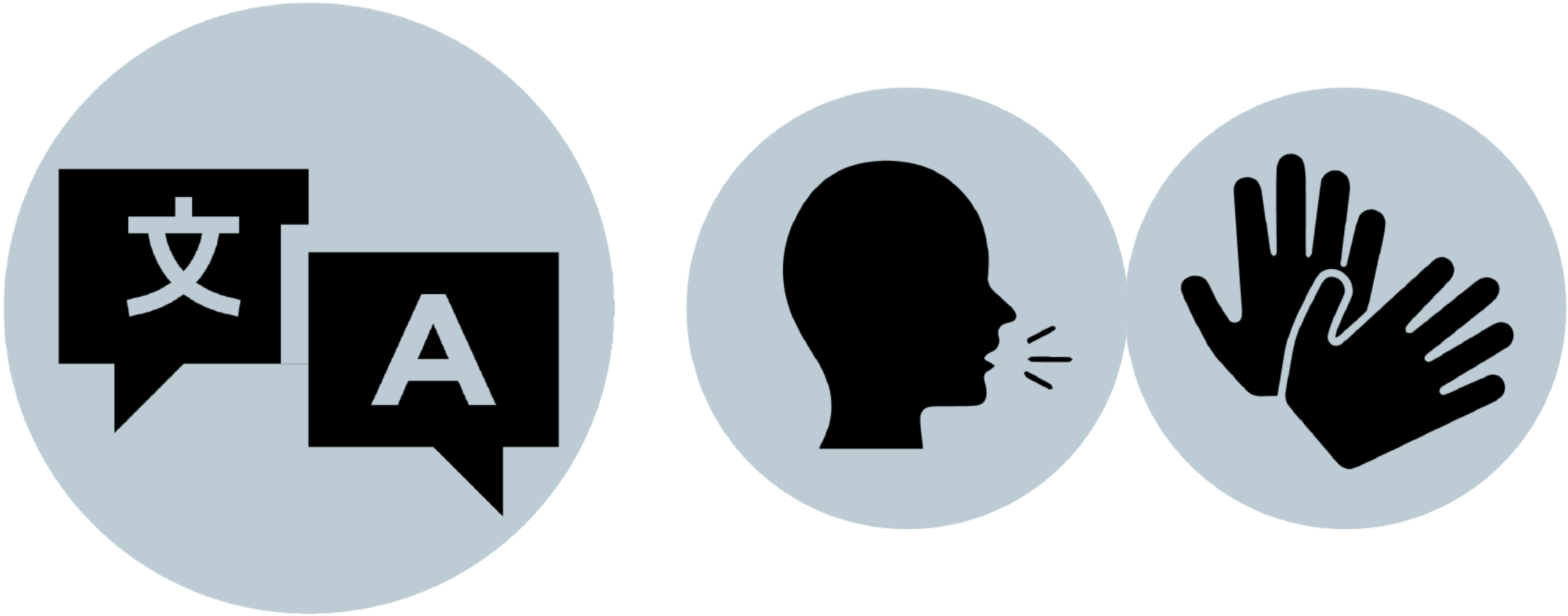
SESSION ID: **SAO-W08V**

How Protocol Gateways May Introduce More Problems than They Solve

Philippe Lin
With Charles Perine, Marco Balduzzi, Ryan Flores, Rainer Vosseler, Luca Bongiorni

Senior Threat Researcher
Trend Micro Forward Looking Research Team
@miaoski

Protocol Gateway = Translator



It Translate Between ...

Between different physical layers of the same protocol family

Modbus/TCP to Modbus/RTU

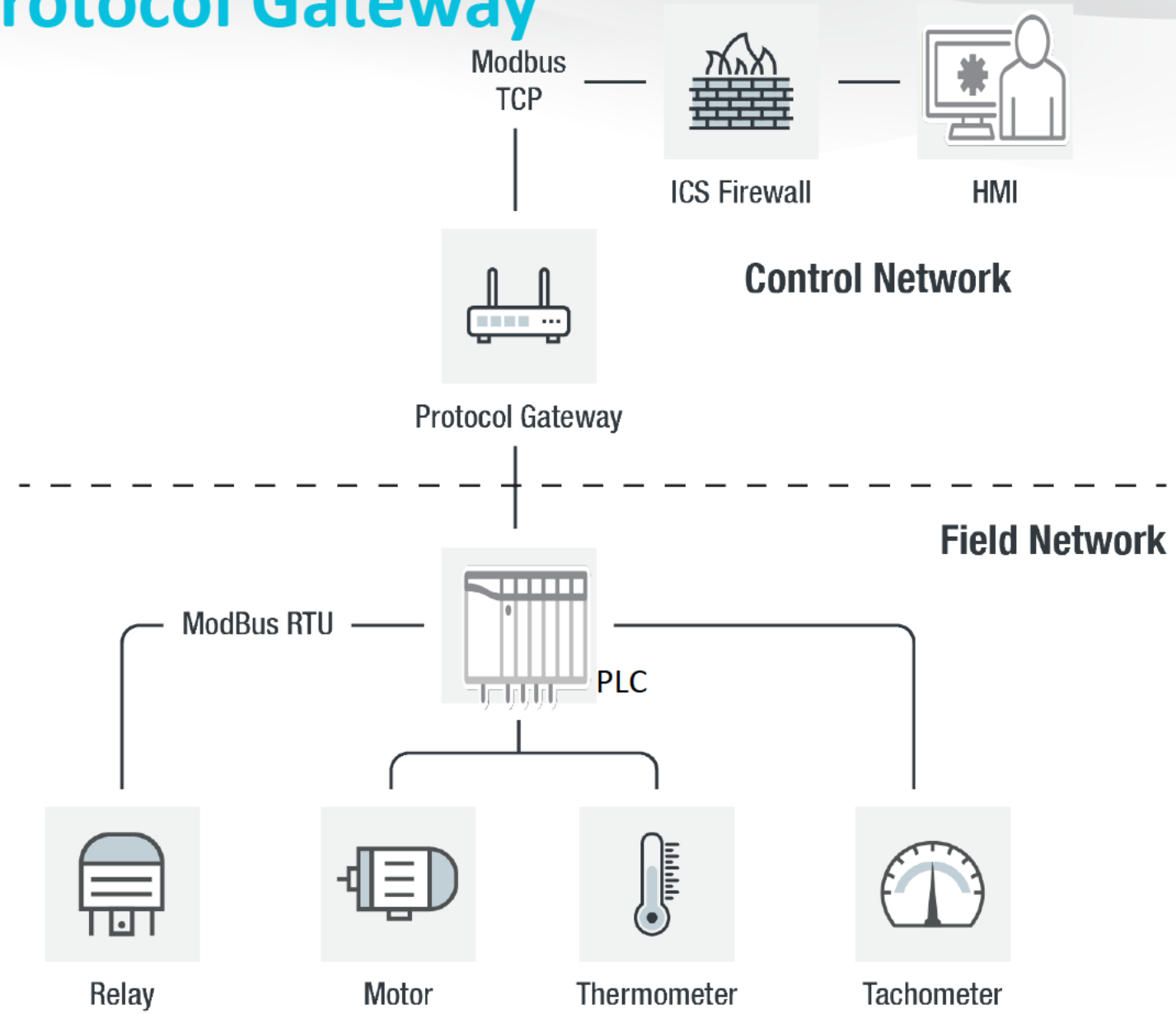
Between different protocols of the same physical layer

Modbus/TCP to Ethernet/IP

Combination of both

Modbus/RTU to Ethernet/IP

Where Is Protocol Gateway



Two Types of Protocol Gateway

Real-time gateways

Incoming packets ➔ evaluated ➔ translated ➔ forwarded

Data stations

Asynchronous translation, Use of mapping tables

We Were Looking for Vulnerabilities in …

- The translator
  - Management software / firmware
  - Heavy traffic → Resource exhaustion
- Protocol translation
  - Protocol parsing → Denial of service
  - Translation errors → Hack the PLC behind
- Cloud support
  - Encrypted and secure?

Protocol Translation

Key to our work

Interesting attack vector for adversaries targeting ICS

Can be abused to conduct stealthy attacks

ICS firewalls bypass

Targets PLC behind protocol gateway

Devices and Problems

	Realtime 1	Realtime 2	Realtime 3	Station 1	Station 2
Translator Vulnerabilities	Yes	Yes	Yes	Yes	Yes
Protocol Translation	Yes	No	No	Yes	Yes
Cloud Support (MQTT)	Yes	N/A	N/A	No	No

Studied Devices

Vendor	Country of Vendor	Price range	Interfaces	Type	OS
Nexcom	Taiwan	200$	Ethernet, serial, wireless	Real-time	FreeRTOS
Schneider	France	600$	Ethernet, serial	Real-time	ThreadX
Digi	USA	350$	Ethernet, serial	Real-time	Embedded Linux
RedLion	USA	650$	Ethernet, serial	Data station	Embedded Linux
Moxa	Taiwan	500$	Ethernet, serial	Data station	Embedded Linux

RSAConference2020 APJ

A Virtual Learning Experience

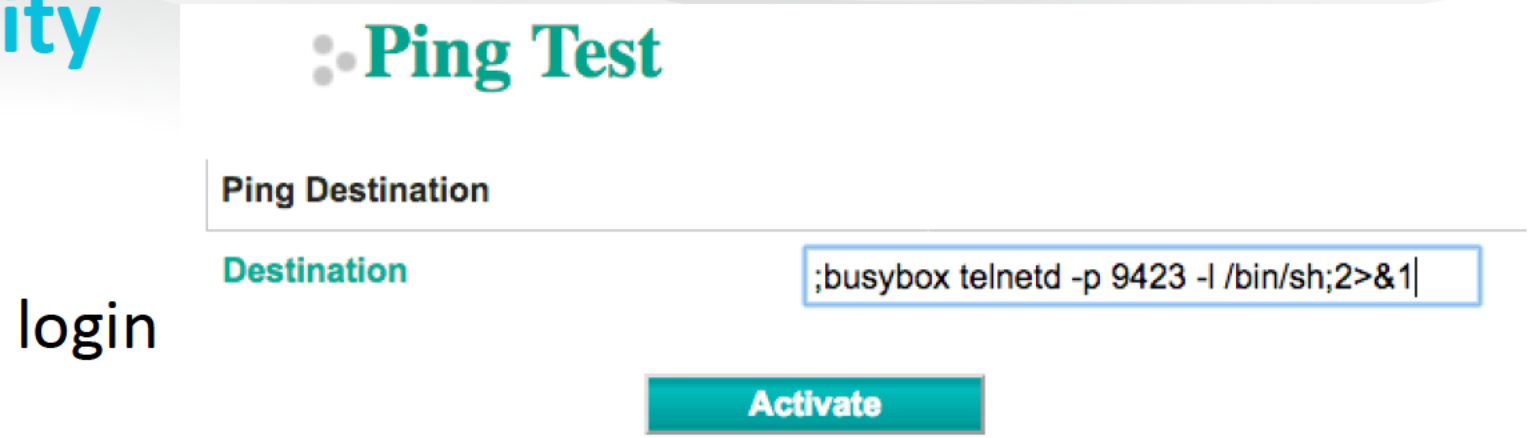
Vulnerabilities

Resource Exhaustion / DoS

Realtime 1	Protocol translation stopped
Realtime 2	Protocol translation stopped
Realtime 3	Reboot
Data Station 1	Crafted packets
Data Station 2	Special packets (documented in user manual)

Firmware Vulnerability

- Privilege escalation
  - Root shell after normal login
  - CVSS 8.8




```
pi@ftr-moxa-modbus:~/modbus_fuzzers/boofuzz-modbus $ telnet 192.168.127.254 9423
Trying 192.168.127.254...
Connected to 192.168.127.254.
Escape character is '^]'.

/ # whoami
root
```

- Credential reuse
  - Random generator w/o seeds
  - Leaks I/O mapping functions

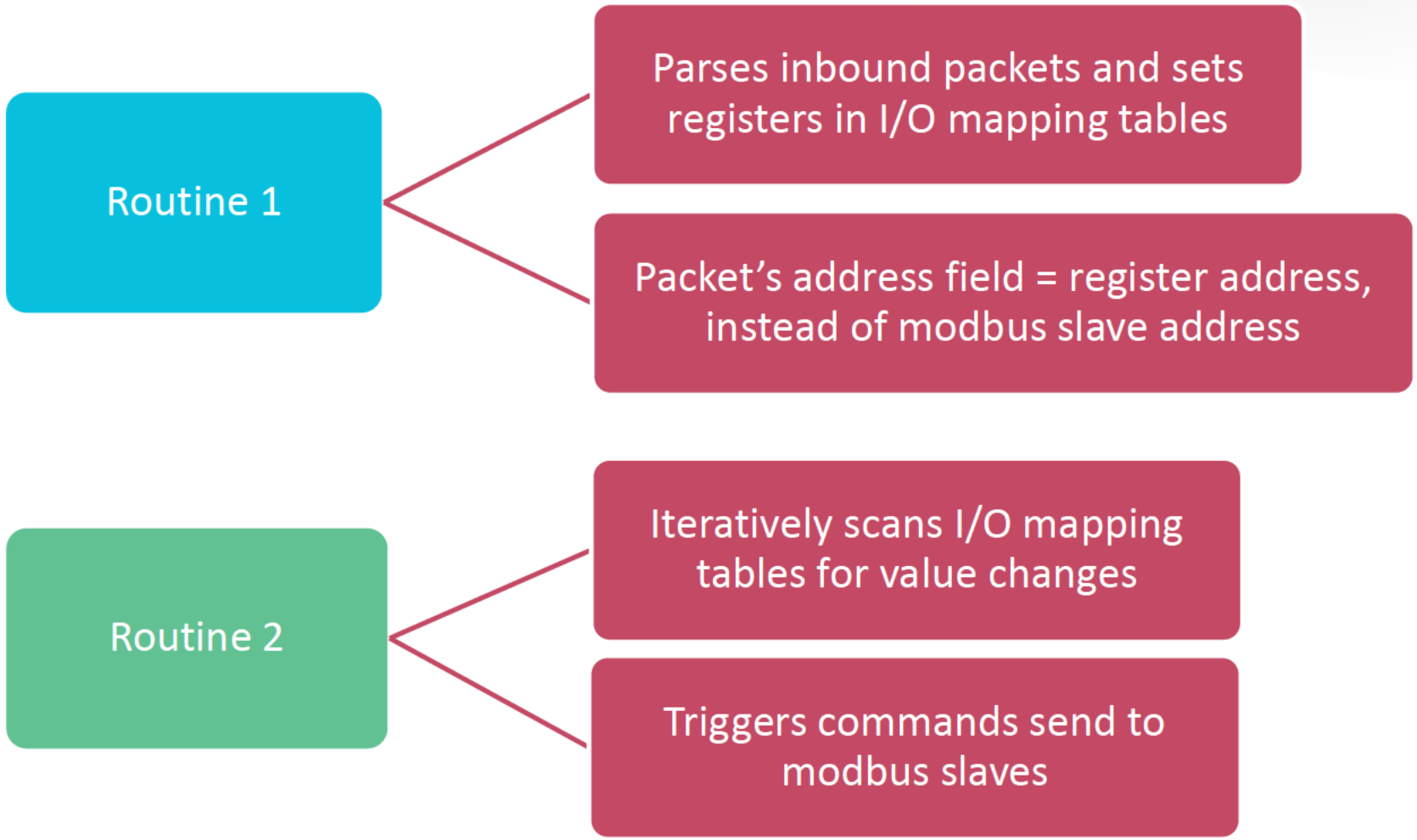
Error in Protocol Translation (Realtime)

Transaction ID	Protocol ID	Message Length	Unit ID	Function Code	Starting Address	Quantity of Regs	Byte Counts	Registers Value		RTU CRC
0001	0000	000B	01	10	0000	0002	04	0001	0005	
			01	10	0000	0002	04	0001	0005	626C

Error in Protocol Translation (Realtime)

Transaction ID	Protocol ID	Message Length	Unit ID	Function Code	Starting Address	Quantity of Regs	Byte Counts	Registers Value		RTU CRC
0001	0000	000B	01	**90**	0000	0002	04	0001	0005	
			01	10	0000	0002	04	0001	0005	626C

Protocol Translation (Data Station)



Normal Operation

#RSAC

1.

Name	Slave ID	Function	Address / Quantity	Trigger
Command1	1	5	Write address 1, Quantity 1	Data Change
Command2	1	6	Write address 11, Quantity 1	Data Change
Command3	1	15	Write address 21, Quantity 50	Data Change
Command4	1	16	Write address 101, Quantity 50	Data Change

1. Fun 5 addr 1, turn ON a coil
2. Maps to internal addr 00
3. Internal addr 00 → 01
4. 00 → 01 !!
5. Send modbus command Fun 5 addr 1 ON

I/O Data Mapping

2.

Data flow direction: Modbus TCP Client <-- Modbus RTU/ASCII Slave

Mapping address arrangement: Automatic

Your device : Modbus TCP Client

Role 1 of MGate 5105-MB-EIP : Modbus TCP Server

Name	Unit ID	TCP Port	Modbus Address
Command1	1	502	4x0001~4x0001
Command2	1	502	4x0001~4x0002
Command3	1	502	4x0002~4x0005
Command4	1	502	4x0006~4x0055

Role 2 of MGate 5105-MB-EIP : Modbus RTU/ASCII Master

Name	Function	Internal Address		Quantity
Command1	5	0	0	1 bytes
Command2	6	1	2	2 bytes
Command3	15	3	9	7 bytes
Command4	16	10	109	100 bytes

Your device : Modbus RTU/ASCII Slave

Submit

3.

Data flow direction: Modbus TCP Client --> Modbus RTU/ASCII Slave

Internal Address	00		01	02	03	04	05	06	07	08	09	0A	0B
0000h	00	→ 01	01	00	02	00	03	00	04	00	05	00	06
0010h	00		09	00	0A	00	00	00	00	00	00	00	00
0020h	00		00	00	00	00	00	00	00	00	00	00	00
0030h	00		00	00	00	00	00	00	00	00	00	00	00
0040h	00		00	00	00	00	00	00	00	00	00	00	00
0050h	00		00	00	00	00	00	00	00	00	00	00	00
0060h	00		00	00	00	00	00	00	00	00	00	00	00
0070h	00		00	00	00	00	00	00	00	00	00	00	00

1. Hacker abuses it

Name	Slave ID	Function	Address / Quantity	Trigger
Command1	1	5	Write address 1, Quantity 1	Data Change
Command2	1	6	Write address 11, Quantity 1	Data Change
Command3	1	15	Write address 21, Quantity 50	Data Change
Command4	1	16	Write address 101, Quantity 50	Data Change

1. Fun 5 addr **25**, turn ON a coil
2. Maps to internal addr **03**
3. Internal addr 02 → 03
4. 02 → 03 !!
5. Send modbus command **Fun 15 addr 21 length 50.**

I/O Data Mapping

2.

Data flow direction: Modbus TCP Client <-- Modbus RTU/ASCII Slave

Mapping address arrangement: Automatic

write write

Your device : Modbus TCP Client

Role 1 of MGate 5105-MB-EIP : Modbus TCP Server

Role 2 of MGate 5105-MB-EIP : Modbus RTU/ASCII Master

Your device : Modbus RTU/ASCII Slave

Name	Unit ID	TCP Port	Modbus Address
Command1	1	502	4x0001~4x0001
Command2	1	502	4x0001~4x0002
Command3	1	502	4x0002~4x0005
Command4	1	502	4x0006~4x0055

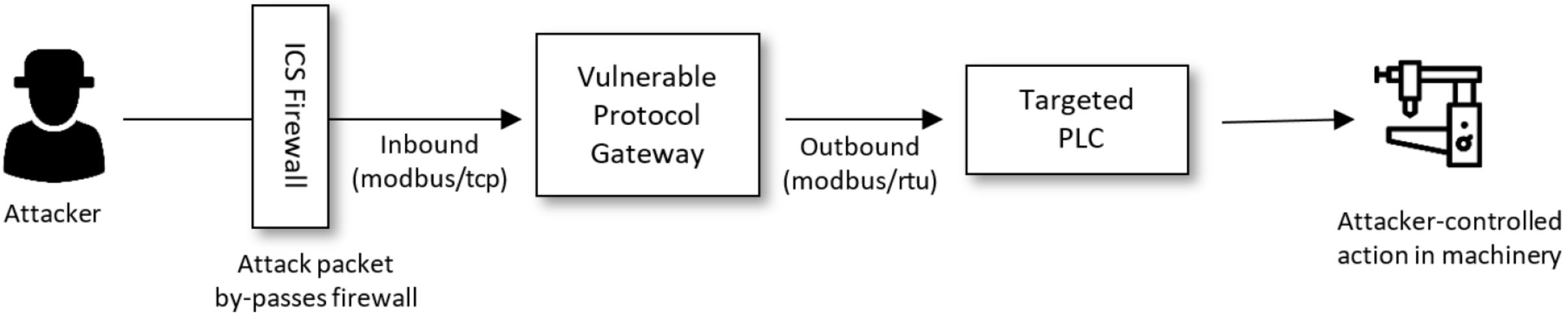
Name	Function	Internal Address		Quantity
Command1	5	0	0	1 bytes
Command2	6	1	2	2 bytes
Command3	15	3	9	7 bytes
Command4	16	10	109	100 bytes

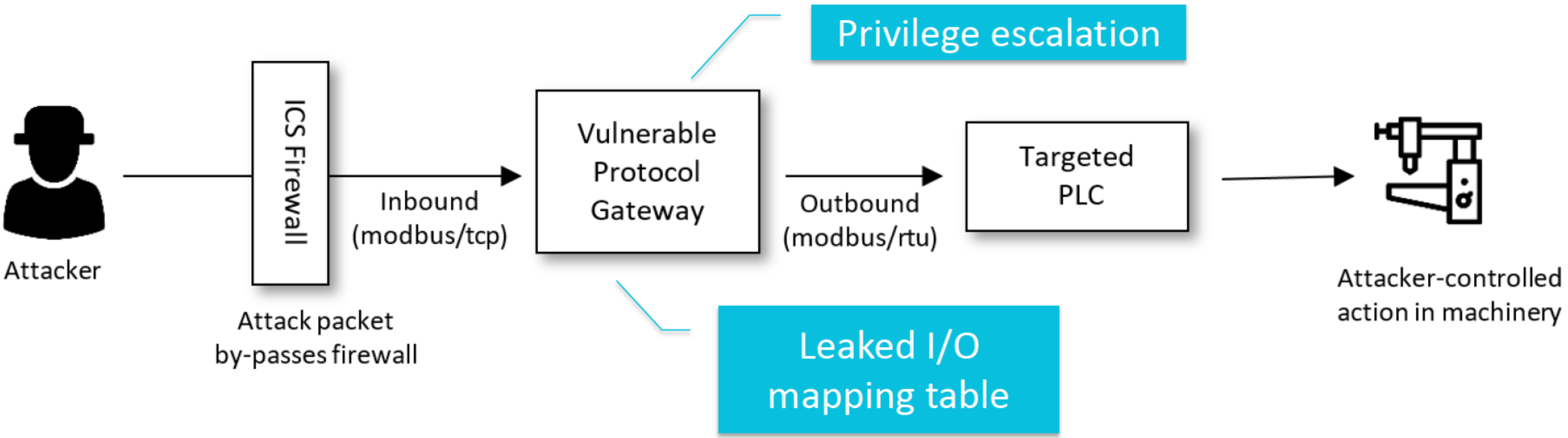
Submit

3.

Data flow direction: Modbus TCP Client --> Modbus RTU/ASCII Slave

Internal Address	00	01	02	03	04	05	06	07	08	09	0A	0B
0000h	00	01	00	02 → 03	00	03	00	04	00	05	00	06
0010h	00	09	00	0A	00	00	00	00	00	00	00	00
0020h	00	00	00	00	00	00	00	00	00	00	00	00
0030h	00	00	00	00	00	00	00	00	00	00	00	00
0040h	00	00	00	00	00	00	00	00	00	00	00	00
0050h	00	00	00	00	00	00	00	00	00	00	00	00
0060h	00	00	00	00	00	00	00	00	00	00	00	00
0070h	00	00	00	00	00	00	00	00	00	00	00	00


Attacker
ICS Firewall
Attack packet
by-passes firewall
Inbound
(modbus/tcp)
Vulnerable
Protocol
Gateway
Outbound
(modbus/rtu)
Targeted
PLC
Attacker-controlled
action in machinery


Privilege escalation
ICS Firewall
Vulnerable Protocol Gateway
Targeted PLC
Attacker
Inbound (modbus/tcp)
Outbound (modbus/rtu)
Attacker-controlled action in machinery
Attack packet by-passes firewall
Leaked I/O mapping table

Cloud (MQTT) Vulnerabilities

- Broken confidentiality
  - Does not support encryption
  - NULL username enables rouge MQTT brokers
- No input validation
  - Arbitrary byte blobs can be translated to MQTT
  - SQL injection to exploit the backend

RSAConference2020 APJ

A Virtual Learning Experience

Summary

Protocol gateways is fundamental in IT-OT integration

Prone to crafted packets and damages PLC and/or production

Valid packets == stealthy attacks

ICS firewall does not always work

Apply What You Have Learned Today (1)

Next Week

- Identify protocol gateways in your ICS environment
- Are they real-time gateways or data stations?
- Are they sending anything to the cloud?

Apply What You Have Learned Today (2)

In 3 Months	Configure your network to protect the gateways
	Whitelist the IP / MAC that can access the gateways
	Contact vendor for firmware upgrade and test the upgrade in lab environment

Apply What You Have Learned Today (3)

Within 6 Months	Add firmware upgrade to scheduled downtime
	Identify an ICS firewall that can filter invalid packets

Disclosed vulnerabilities

Gateway	Name	ID	Reporting Date	Status
Realtime gateway 1	Protocol Translation Bypass	ZDI-CAN-10485	Feb 10, 2020	Open
	Unencrypted MQTT	ZDI-CAN-10486	Feb 10, 2020	Open
	Authentication Bypass	ZDI-CAN-10487	Feb 10, 2020	Open
	Unsanitized MQTT Upstream	ZDI-CAN-10488	Feb 10, 2020	Open
Data Station 1	Modbus Read Denial-of-Service	ZDI-CAN-10804	Mar 23, 2020	Open
	Arbitrary Memory Leakage	ZDI-CAN-10897	Apr 5, 2020	Open
Data station 2	Information disclosure through Proprietary Commands	ZDI-CAN-10792	Mar 18, 2020	Open
	Credential reuse through Proprietary Commands	ZDI-CAN-10791	Mar 18, 2020	Open
	Post-auth root shell and persistence	CVE-2020-8858	Oct 14, 2019	**Fixed**

Full disclosure in mid August.