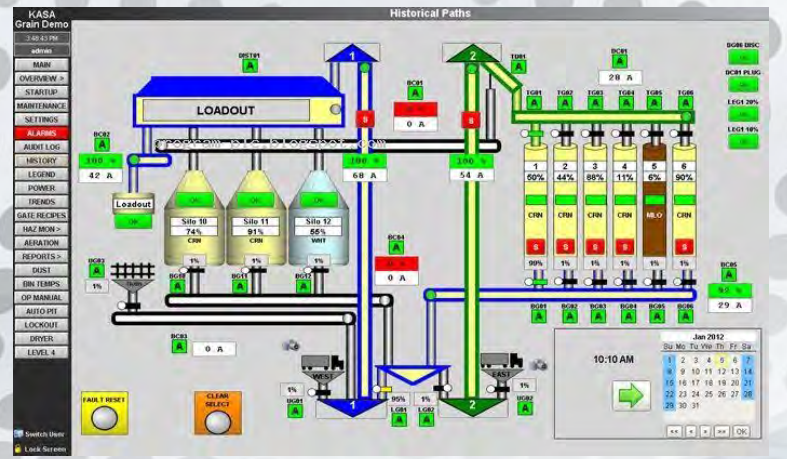
Industrial Control System (ICS) Cyber Threat Hunting Using Suricata

Leonard Jacobs, M.S., MBA, CISSP, CSSA

Presented at Suricon 2021

October 21, 2021

What are Industrial Control Systems?



Where are Industrial Control Systems?

- Industrial Control Systems are the devices and systems that “Run the World.”
- Industrial Control Systems are found everywhere.
- ICS run manufacturing processes or keep electricity flowing or keep the natural gas flowing through pipelines by controlling the pressure or systems that run oil refineries or the computer that runs an automobile.

Netsecuris LLC

Cybersecurity Driven by Human Intelligence ™

What makes ICS different from other kinds of systems?

- ICS control functions and processes.
  - ICS is not so much about data being generated than it is about controlling functions and facilitating processes.
  - There can be a subtle difference between whether an industrial function is failing due to some physical failure or a cyber attack.
  - Cyber Threat Hunting is so important in an industrial control environment.

What makes ICS different from other kinds of systems?

- Life Safety, Availability, Integrity, and Confidentiality.
  - Life safety is most important in industrial environments. People and the environment can be harmed if a cyber attack causes havoc.
  - Secondly, availability is foremost in industrial environments. If production comes to a halt due to cyber attacks, then a financial impact on a company can occur.
  - Secondly, availability is foremost in industrial environments. If production comes to a halt due to cyber attacks, then a financial impact on a company can occur.
  - Confidentiality mostly comes into play within an industrial environment if a process or formula or network diagrams is exposed. ICS may contain information about a process or formula in the coding of the system or device.

What makes ICS different from other kinds of systems?

- ICS have different protocols than IT.
  - ICS use specialized TCP/IP protocols that you don't find in IT Business Systems environments.
  - For example, Application Layer protocol Modbus TCP has special function codes that are used to control the functionality of an ICS and can even shutdown a device.
  - Imagine if a cyber attacker was able to force an industrial control device offline.

Netsecuris LLC

Cybersecurity Driven by Human Intelligence ™

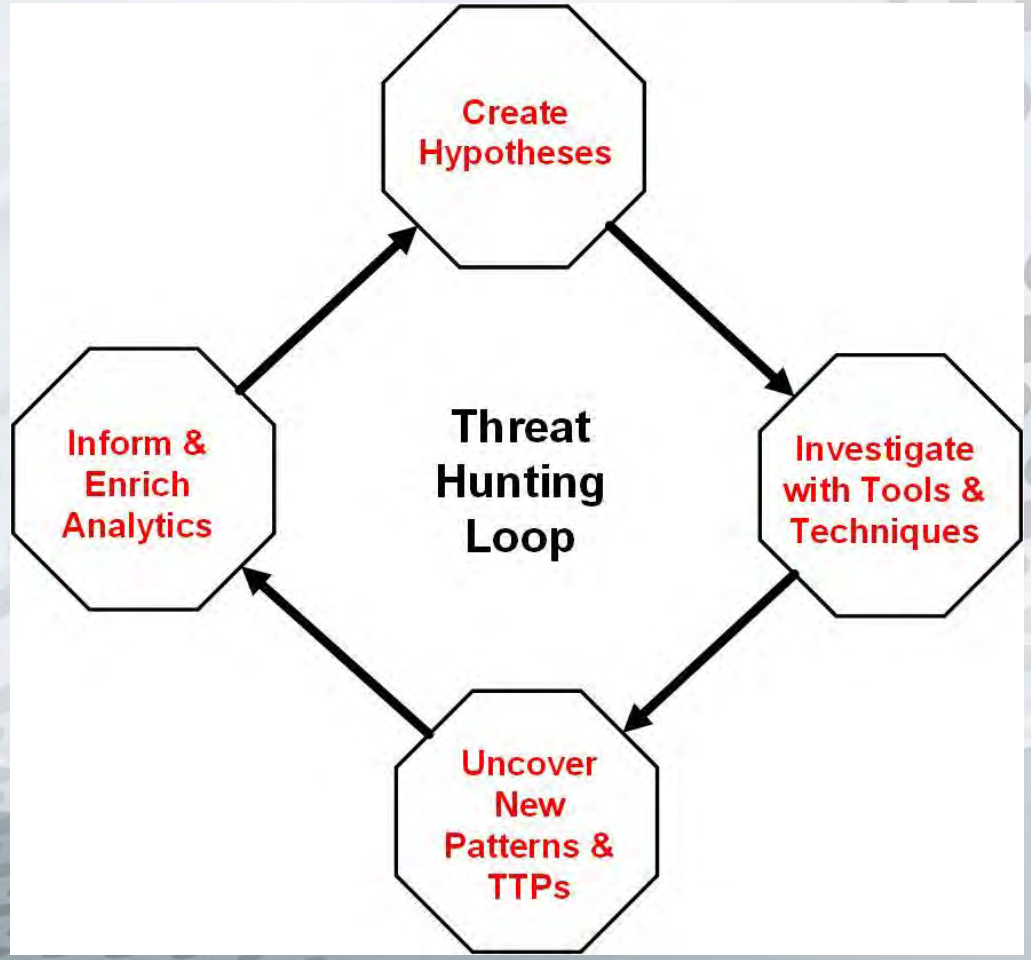
What is Cyber Threat Hunting?

- Threat hunting is the practice of proactively searching for cyber threats that are lurking undetected in a network and in systems.
- Threat hunting is an essential component of any defense strategy.
- Monitoring alerts and logs is good for known alert patterns; Threat Hunting is designed to detect unknown traffic patterns.

Netsecuris LLC

Cybersecurity Driven by Human Intelligence ™

Cyber Threat Hunting Process



Source: Sqrrl circa 2012

Netsecuris LLC
Cybersecurity Driven by Human Intelligence ™

Types of Cyber Threat Hunting

- A **structured hunt** is based on the tactics, techniques and procedures (TTPs) of an attacker.
- An **unstructured hunt** is initiated based on a trigger.
- A **hybrid hunt** is based on a combination of structured and unstructured hunts.

Forming the Cyber Threat Hunt Hypothesis

- Hypothesis-driven investigations are often triggered by a new threat.
- Hypothesis-driven investigations can be triggered by a new behavior seen in systems and networks.
- Hypothesis-driven investigations can be triggered by a system result seeming out of the ordinary or unusual.

Collecting ICS Security Data with Suricata

Test Bench Setup Example

- RaspberryPI 4 8GB model, with either Raspbian or Ubuntu 20.04 Server OS and Suricata version 6.0.3 installed.
- Raspberry Pi's USB3 port connected to test network with USB3 Ethernet Adaptor for data collection and management.
- Netgear GS308E Gigabit Network Managed Switch with Port Mirror connected to Pi's Gigabit network port monitoring PLC connection port on switch.

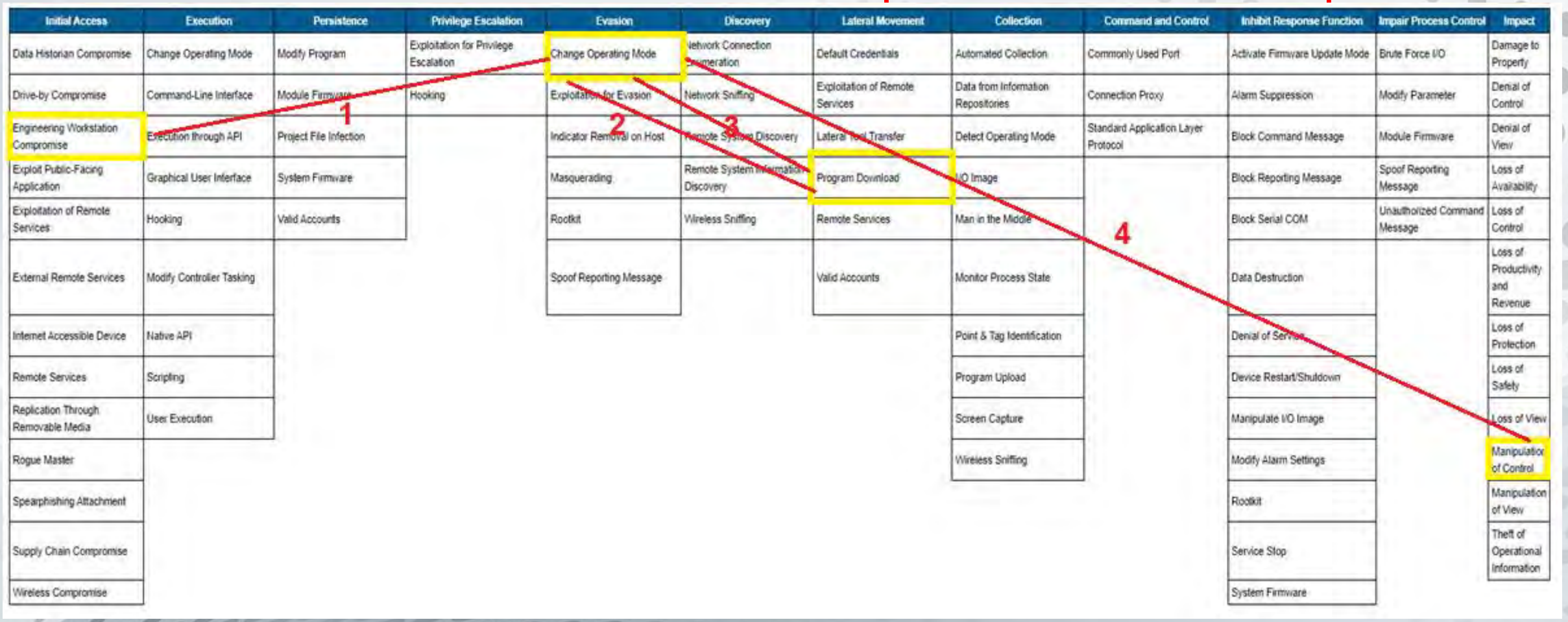
Netsecuris LLC
Cybersecurity Driven by Human Intelligence ™

MITRE ATT&CK for ICS Knowledgebase

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

Source: https://collaborate.mitre.org/attackics/index.php/Technique_Matrix

MITRE ATT&CK for ICS Simple Hunt Example



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

MITRE ATT&CK For ICS Attack Scenario

Run a Cyber Threat Hunt looking for the following activity. Based on the hypothesis that a malicious actor gains access to an Engineering Workstation. What all could be accomplished?

1. Malicious actor changes the operating mode of the target ICS controller to program mode.
2. Malicious actor downloads a modified program to target ICS controller.
3. Malicious actor changes the operating mode of the target ICS controller to run mode.
4. Malicious actor has manipulated the controls on the target ICS controller.

Suricata signatures and other functions can be utilized to look for evidence of this activity.

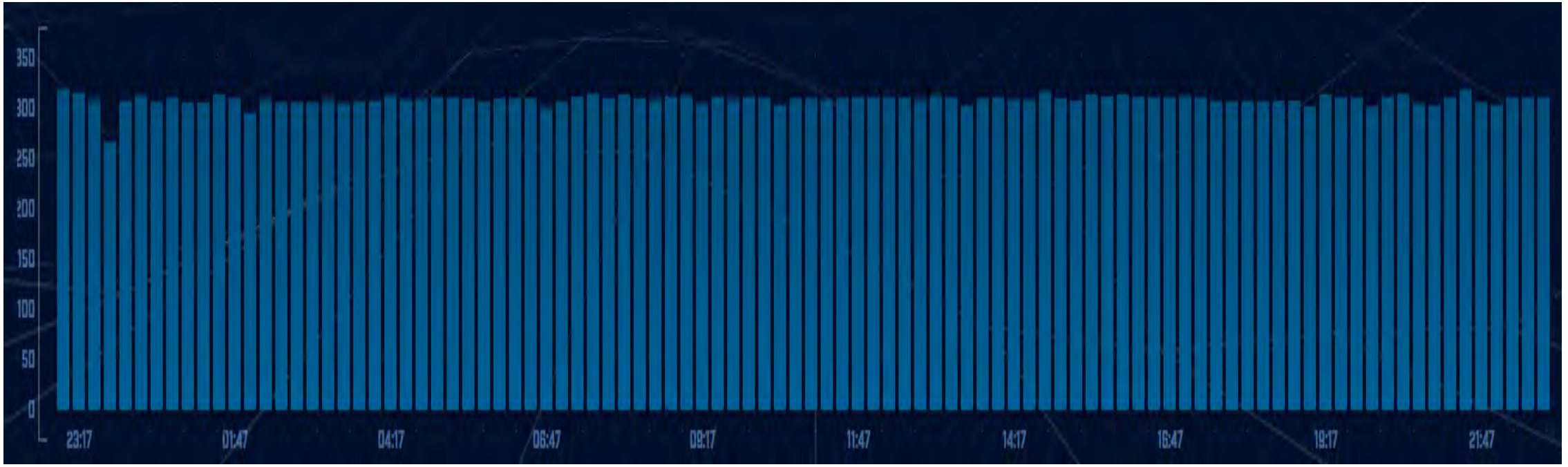
Cyber Threat Hunting on ICS Networks

- Check for persistent connections.
- Analyze the amount of data being communicated.
- Dig Deeper into the network.

Netsecuris LLC

Cybersecurity Driven by Human Intelligence ™

Cyber Threat Hunting Evidence Example



This example shows the connection frequency that has been graphed over a 24-hour period. Every bar represents the number of connections that took place every 15 minutes. This consistency most likely represents Command and Control (C2) connection activity.

Netsecuris LLC
Cybersecurity Driven by Human Intelligence ™

Cyber Threat Hunting Evidence Example

This example shows the number of connections transferring amounts of data. In this case approximately 42,000 connections transferred 52 bytes of data over a 24-hour period.

Netsecuris LLC
Cybersecurity Driven by Human Intelligence ™

ICS Protocols

- Modbus TCP
  - The device programming values can be controlled with these function codes.

Commonly used public function codes				
Code	Hex	Function	Type	
01	01	Read Coils	Single Bit Access	Data Access
02	02	Read Discrete Inputs		
05	05	Write Single Coil		
15	0F	Write Multiple Coils		
03	03	Read Holding Registers	16 bit Access	
04	04	Read Input Register		
06	06	Write Single Register		
16	10	Write Multiple Registers		
22	16	Mask Write Register		
23	17	Read/Write Multiple Registers		
24	18	Read FIFO queue		
20	14	Read File Record	File record access	
21	15	Write File Recore		
07	07	Read Exception Status	Diagnostics	
08	08	Diagnostic		
11	0B	Get Com event counter		
12	0C	Get Com Event Log		
17	11	Report Server ID		

ICS Protocols

- Distributed Network Protocol 3 (DNP3)
  - The device programming values can be controlled with these function codes.
  - DNP3 specifications provides for Internal Indicator Codes.

Requests (Hex)			
0	Confirm	10	Initialize application
1	Read	11	Start application
2	Write	12	Stop application
3	Select	13	Save configuration
4	Operate	14	Enable unsolicited
5	Dir operate	15	Disable unsolicited
6	Dir operate – No resp	16	Assign class
7	Freeze	17	Delay measurement
8	Freeze – No resp	18	Record current time
9	Freeze clear	19	Open file
A	Freeze clear – No resp	1A	Close file
B	Freeze at time	1B	Delete file
C	Freeze at time – No resp	1C	Get file information
D	Cold restart	1D	Authenticate file
E	Warm restart	1E	Abort file
F	Initialize data		
Responses (Hex)			
81	Response		
82	Unsolicited response		

LSB	
IIN1.0	All stations
IIN1.1	Class 1 events
IIN1.2	Class 2 events
IIN1.3	Class 3 events
IIN1.4	Need time
IIN1.5	Local control
IIN1.6	Device trouble
IIN1.7	Device restart
MSB	
IIN2.0	Function code not supported
IIN2.1	Object unknown
IIN2.2	Parameter error
IIN2.3	Event buffer overflow
IIN2.4	Already executing
IIN2.5	Configuration corrupt
IIN2.6	Reserved 1
IIN2.7	Reserved 2

Modbus PCAP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.57	10.0.0.3	TCP	60	2387 → 502 [FIN, ACK] Seq=1 Ack=1 Win=64099 Len=0
2	0.000122	10.0.0.3	10.0.0.57	TCP	60	502 → 2387 [ACK] Seq=1 Ack=2 Win=65439 Len=0
3	0.000493	10.0.0.3	10.0.0.57	TCP	60	502 → 2387 [FIN, ACK] Seq=1 Ack=2 Win=65439 Len=0
4	0.000536	10.0.0.57	10.0.0.3	TCP	60	2387 → 502 [ACK] Seq=2 Ack=2 Win=64099 Len=0
5	2.751380	10.0.0.57	10.0.0.3	TCP	62	2578 → 502 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1
6	2.751546	10.0.0.3	10.0.0.57	TCP	62	502 → 2578 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM=1
7	2.751605	10.0.0.57	10.0.0.3	TCP	60	2578 → 502 [ACK] Seq=1 Ack=1 Win=64240 Len=0
8	15.266493	10.0.0.57	10.0.0.3	Modbus…	66	Query: Trans: 0; Unit: 10, Func: 8/ 1: Force Listen Only Mode
9	15.268405	10.0.0.3	10.0.0.57	Modbus…	63	Response: Trans: 0; Unit: 10, Func: 8: Diagnostics. Exception returned
10	15.268888	10.0.0.57	10.0.0.3	Modbus…	66	Query: Trans: 0; Unit: 10, Func: 8/ 1: Force Listen Only Mode
11	15.271020	10.0.0.3	10.0.0.57	Modbus…	63	Response: Trans: 0; Unit: 10, Func: 8: Diagnostics. Exception returned
12	15.271447	10.0.0.57	10.0.0.3	Modbus…	66	Query: Trans: 0; Unit: 10, Func: 8/ 1: Force Listen Only Mode
13	15.273608	10.0.0.3	10.0.0.57	Modbus…	63	Response: Trans: 0; Unit: 10, Func: 8: Diagnostics. Exception returned
14	15.458432	10.0.0.57	10.0.0.3	TCP	60	2578 → 502 [ACK] Seq=37 Ack=28 Win=64213 Len=0
15	25.889380	10.0.0.57	10.0.0.3	Modbus…	66	Query: Trans: 0; Unit: 10, Func: 8/ 1: Restart Communications Option

3 way Handshake

```
> Frame 15: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
> Ethernet II, Src: Runtop_00:62:0d (00:20:78:00:62:0d), Dst: Intel_ce:70:51 (00:02:b3:ce:70:51)
> Internet Protocol Version 4, Src: 10.0.0.57, Dst: 10.0.0.3
> Transmission Control Protocol, Src Port: 2578, Dst Port: 502, Seq: 37, Ack: 28, Len: 12
v Modbus/TCP
    Transaction Identifier: 0
    Protocol Identifier: 0
    Length: 6
    Unit Identifier: 10
v Modbus
    .000 1000 = Function Code: Diagnostics (8)
    Diagnostic Code: Restart Communications Option (1)
    Restart Communication Option: Leave Log (0x0000)
```

Netsecuris LLC
Cybersecurity Driven by Human Intelligence ™

DNP3 PCAP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.8	10.0.0.3	TCP	62	2789 → 20000 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
2	0.000201	10.0.0.3	10.0.0.8	TCP	62	20000 → 2789 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM=1
3	0.000411	10.0.0.8	10.0.0.3	TCP	60	2789 → 20000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
4	0.001765	10.0.0.3	10.0.0.8	DNP 3.0	71	Unsolicited Response
5	0.152060	10.0.0.8	10.0.0.3	TCP	60	2789 → 20000 [ACK] Seq=1 Ack=18 Win=65518 Len=0
6	3.043942	10.0.0.8	10.0.0.3	DNP 3.0	69	Confirm
7	3.044737	10.0.0.8	10.0.0.3	DNP 3.0	79	Write, Time and Date
8	3.044845	10.0.0.3	10.0.0.8	TCP	60	20000 → 2789 [ACK] Seq=18 Ack=41 Win=65495 Len=0
9	3.066055	10.0.0.3	10.0.0.8	DNP 3.0	71	Response
10	3.256739	10.0.0.8	10.0.0.3	TCP	60	2789 → 20000 [ACK] Seq=41 Ack=35 Win=65501 Len=0
11	123.402417	10.0.0.8	10.0.0.3	DNP 3.0	78	Disable Spontaneous Messages
12	123.409014	10.0.0.3	10.0.0.8	DNP 3.0	71	Response
13	123.537063	10.0.0.8	10.0.0.3	TCP	60	2789 → 20000 [ACK] Seq=65 Ack=52 Win=65484 Len=0
14	684.542677	10.0.0.8	10.0.0.3	TCP	62	2803 → 20000 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
15	684.542851	10.0.0.3	10.0.0.8	TCP	62	20000 → 2803 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 SACK_PERM=1

DNP3 Response C ode

Write Request

0x15 Function Code

```
> Frame 4: 71 bytes on wire (568 bits), 71 bytes captured (568 bits)
> Ethernet II, Src: Intel_ce:70:51 (00:02:b3:ce:70:51), Dst: 3com3c90_93:70:67 (00:50:04:93:70:67)
> Internet Protocol Version 4, Src: 10.0.0.3, Dst: 10.0.0.8
> Transmission Control Protocol, Src Port: 20000, Dst Port: 2789, Seq: 1, Ack: 1, Len: 17
v Distributed Network Protocol 3.0
  > Data Link Layer, Len: 10, From: 4, To: 3, PRM, Unconfirmed User Data
  > Transport Control: 0xe6, Final, First(FIR, FIN, Sequence 38)
  > Data Chunks
  > [1 DNP 3.0 AL Fragment (4 bytes): #4(4)]
  v Application Layer: (FIR, FIN, CON, UNS, Sequence 7, Unsolicited Response)
    > Application Control: 0xf7, First, Final, Confirm, Unsolicited(FIR, FIN, CON, UNS, Sequence 7)
      Function Code: Unsolicited Response (0x82)
    > Internal Indications: 0x1000, Time Sync Required
```

Suricata ICS Protocols Keywords

- Modbus Keywords in Signatures
  - Three ways to use the keywords:
    1. Matching on function's properties with the setting "function"
    2. Matching on directly on data access with the setting "access"
    3. Matching on unit identifier with the setting "unit" only or with the previous setting "function" or "access"
  - Example Signature

alert modbus $MODBUS_CLIENT any -> $MODBUS_SERVER 502
(modbus.access: write; msg: "Modbus write access to host"; sid:999999; rev:1;)

Source: Suricata User Guide

Suricata ICS Protocols Keywords

- DNP3 Keywords in Signatures
  - Match on Function Codes
  - Match on internal indicator fields
  - Match on application data objects
  - Match on re-assembled application buffer
  - Example Signature

  alert dnp3 $DNP3_CLIENT any -> $DNP3_SERVER 20000 (dnp3_ind: no_func_code_support; msg: ”Detect Illegal DNP3 Function Code”; sid:999999; rev:1;)

Source: Suricata User Guide

Suricata ICS Protocol Keywords

- Additional ICS protocol keywords supported:
  - ENIP/CIP
    - Ethernet/IP is commonly found in use within many industrial control environments.
    - Ethernet/IP adapts CIP (Common Industrial Protocol) to standard Ethernet.
  - MQTT (Message Queue Telemetry Transport)
    - MQTT is being used in more ICS environments with both servers and ICS devices.
    - It has been integrated into some brands of PLCs.

ICS Protocol Keyword Candidates

- BACnet/IP
  - Protocol is predominately utilized in Building Management Systems.
  - Protocol utilizes UDP for the speed afforded by connection-less communications.
  - Protocol can broadcast messages to devices or devices can communicate directly with other devices.
  - Type Codes, Function Codes, Version Codes, and Control Codes to specify various aspects of the protocol.
- Siemens S7comm
  - Protocol is proprietary to Siemens ICS equipment but is widely used.
  - It is used for PLC programming, exchanging data between PLCs, accessing PLC data from SCADA systems and diagnostic purposes.
  - Protocol is Function oriented or Command oriented.
  - Each command consists of a header, a set of parameters, a parameters data, and a data block.
    - The first two elements are always present, the other are optional.
  - Example command structure is Write this data into DB 10 starting from the offset 4.
  - Commands are divided into Data Read/Write, Cyclic Data Read/Write, Directory info, System Info, Blocks move, PLC Control, Date and Time, Security, and Programming.

ICS Protocol Keyword Candidates

- OPC UA (Open Platform Communication Unified Architecture)
  - Protocol is the next-generation data exchange standard for machine-to-machine (M2M) and sensor-to-cloud use-cases.
  - Protocol can help solve the challenge on how to connect the factory floor architectures with the enterprise systems.
  - Information is securely conveyed using OPC-UA-defined and vendor-defined data types, and servers define object models that clients can dynamically discover.
  - There is a Rust implementation of OPC UA. https://github.com/locka99/opcua

Want to know more about ICS Protocols?

https://en.wikipedia.org/wiki/List_of_automation_protocols

Netsecuris LLC

Cybersecurity Driven by Human Intelligence ™

Whoami

Leonard Jacobs, M.S., MBA, CISSP, CSSA

Ljacobs@netsecuris.com

(662) 667-7796

https://www.netsecuris.com

https://www.cybersafeday.com

Netsecuris focuses on providing cybersecurity protections for critical infrastructure. Protecting critical infrastructure is one of the most important, if not the topmost task today.

Without cyber-securing critical infrastructure, there would be no reliable electricity to power the rest of the technology, including IT systems, that runs the World.