Threat Hunter Playbook for ICS:

Identifying defender tactics for DET&CT-ion in ICS Threat Hunting

By: Alexa Thomsen

Abstract

Every 39 seconds there is a cyber-attack and 43% of cyber-attacks target small businesses (Yoo, 2021). It takes an average of 6 months to detect a data breach and more than 77% of organizations do not have an incident response plan (Databasix UK, 2023). There are many guides, threat intelligence, and detection methods put together for threat hunting on standard enterprise networks, but it becomes more limited when looking at Industrial Control System (ICS) networks. When hunting on ICS, the network needs to be approached differently, and not all tools work the same way that you would normally use them. Nmap is a good example of this, where it is relatively safe to use in an enterprise environment but can have catastrophic effects in an ICS environment if not used properly. Some companies have created very thorough products that can highlight problems in a network, identify entry methods, and provide recommendations for remediation. But what if you are unable to have one of those products connected all the time, or didn't previously have one connected when called to respond to an incident? This talk presents methodology for ICS-based threat hunting and the creation of an open-source Threat Hunter Playbook that identifies both exploitation and detection methods for ICS/SCADA networks and MITRE T-codes, that uses only open-source tools to search for and identify malicious cyber actors. You never know where an incident might occur, and you might get called to investigate something, where a larger company does not already have a product in place. Some of the smaller local networks may not be able to afford a larger company product but are still at risk of being targeted. This playbook would help both local defenders and incident response teams. The detection methods that are listed capture ways to identify if that MITRE T-Code shows any evidence of a malicious cyber actor utilizing that technique, and the exploitation methods capture ways to test the security of your network, by being able to apply mitigations. While some detection methods for enterprise overlap, that is not always the case when it comes to detection on ICS. The playbook (when complete) will have two main parts. The first part will use open-source threat intelligence to map threats to MITRE, as well as include an overlay of detection capabilities for each T-Code. This then allows an intelligence person to highlight the most likely avenue for certain threat actors/groups based on environment type, which feeds the creation of a threat hunt plan. The Hunt Plan is displayed in a table format showing the exploitation methods, detection methods, analytics for detection, and finally defense or mitigation items. Each T-code can be selected for more information, including a description of the T-code, the affected platforms, and the data sources that should be pulled by the operator. Future features would include the ingest of logs and PCAPs to filter detection capabilities further based on actual environment data. I will also provide resources for tools, scripts, YARA or Snort rules, and ICS exploitation tools.

Introduction

Human error accounts for 95% of all data breaches with 75% of them starting with an email (Databasix UK, 2023). If you operate on Industrial Control Networks, you may think that you don't have to worry about email traffic affecting your network, however phishing still accounts for 71% of cyberattack attempts on critical infrastructure (Databasix UK, 2023). That is not the only attack vector and over 50% of companies say that they do not have a sophisticated enough cyber team to handle advanced cyberattacks (Databasix UK, 2023). Without direct threat intelligence, threat hunters need a place to start and a guideline to build their "hunt plan" in an effort to identify suspicious or malicious cyber activity. While there are commercial avenues that offer advanced monitoring, threat intelligence, and comprehensive platforms, not all critical infrastructure teams or organizations can afford or are allowed to implement those. As a threat hunter, you never know where an incident might occur, and you might get called to investigate something where a larger company does not already have a product in place. For those organizations and teams, this whitepaper outlines a threat hunter playbook with both detection and exploitation methods to test mitigations in place and uses only open-source tools and resources. It expands upon MITRE resources in a simple to use format for use by anyone. This paper is not intended to represent the "end-all" solution for threat hunting or fully offer what larger companies are doing for free. Threat hunting still requires a dedicated team of trained individuals and intelligence on threats, in order to proactively look for threats. This paper intends to outline a framework for some industry-identified best practices and a tool to help threat hunting teams orient themselves to the environment they are protecting.

Threat Hunting

Threat hunting has various definitions, with no set definition decided on by the industry. When talking about threat hunting, there are a few key words that stand out in every definition. It is 'proactive', 'hypothesis-based', and involves using various data sources to find evidence of undetected threats before they cause a major breach. The main goal or purpose is to reduce the time it takes to find attackers or threats and prevent damage from occurring. This is not prevention at the gateway but finding those actors who have made it past initial security defenses. Threat hunting is a shift towards identifying the adversary farther left in the kill chain and adopting the mindset that the adversary is already present. A reactive (vs. proactive) response is Incident Response, which seeks to respond once an incident has been noticed in the environment. When threats are not being detected soon enough and an organization is mature enough, they can add threat hunting to start proactively looking for threats and identifying them sooner (Lee & Lee, 2017). It cannot be solved by a single tool or by simply implementing new software. It requires both a manual and semi-automated scanning of systems to look for evil and a combination of both human analysts and behavioral-based analytics.

ICS Threat Hunting

ICS is the generalized term that encompasses multiple industrial and critical infrastructure sectors from electric power (which includes delivery, generation, and load) to water and wastewater management to manufacturing to airfield lighting, and many more.

Threat hunting on ICS is not the same as typical Enterprise threat hunting. There are items that need to be considered in the technology, the network design, system limitations, and the threat groups and their tactics. The systems across ICS vary greatly, which requires the threat hunter to be familiar with the system types and certain considerations for ICS.

ICS Considerations

When hunting and assessing an ICS/SCADA network, there are a few things that need to be taken into consideration before execution The upgrade lifecycle for ICS can be much longer due to the use of proprietary operating systems, compatibility issues between software and newer OSs, and/or the significant cost to upgrade. Besides operating systems there also could be older switches on the network which would mean they are likely unmanaged. An unmanaged switch makes it so you are unable to configure a SPAN port for traffic and must use a network TAP. The operating systems in an OT environment may be outdated or proprietary and can include Real-Time Operating Systems. There are also hundreds of different proprietary protocols, with a few standard protocols.

Availability is the key priority for ICS. The use of agents may be limited based on available hardware and the system owners risk appetite. Owners may be hesitant to add any additional software that has the potential to bring down the system or increase computing load on the system. Some processors are very small, to do a very specific job, and cannot handle the addition of some agents. Previously, some system owners have approved the use of Winlogbeat to transmit logs or the addition of syslog for visibility.

OT operating systems and platforms cannot always support the standard employment of IT cybersecurity tools, and often need ICS specific tools, designed to be compatible with an OT system. Vulnerability scanners or network scanners have knocked ICS equipment offline if they were not configured appropriately. It is important that for every part of set-up, you work directly with the engineers to safely test and install your tools.

Information Technology (IT) vs. Operation Technology (OT)

Many of us are familiar with Information Technology (IT). We tend to use it daily, and it is the configuration of any enterprise network. What most people are not familiar with is Operational technology (OT). This refers to hardware and software that detects or causes a change through the direct monitoring and/or control of physical devices, processes, and events in the enterprise (116 Congress, 2020). This occurs more on a “factory floor” type area, versus an office. OT involves the actual processes controlling industrial assets. Typically, OT requires a special skillset and support can be limited or proprietary. The devices may have limited storage capabilities, and limited abilities to add on additional tools or “agents”. They also can be finicky, and care must be taken to ensure availability. A lot of these differences encompass the considerations that must be taken when working in an industrial environment.

Category	Information Technology (IT)	Operation Technology (OT)
Skillset/Support	Generalized skillset, extensive support available	Specialized skills, limited support, specialized engineers
Computing Power/Storage	Expandable power and storage capabilities can be added on	Limited to what exists at the time, and typically includes only what it needs
Availability	Able to have limited periods of downtime; still aim for 99.999% uptime	Availability is #1 priority, and downtime can have drastic consequences
Consequences of Risk	Might lose money or productivity	Can result in loss of life if compromised
Operating Systems	Share a common standard; Typically, the latest version of Linux or Windows	Wide variety depending on the system; potentially proprietary to vendor; RTOS at lower levels
Environmental Conditions	Controlled conditions with HVAC, debris filtering, clean rooms	Environments can vary to extremes as needed depending on the factory type
Maintenance Schedule	Regular updates, patching, and maintenance	Must be scheduled well in advance, and only what is needed
Device Policy	Some places allow BYOD, or may be controlled depending on the job	Very limited to what is provided by the vendor and authorized to connect to the system
Security	Starting to be built-in, and is considered as part of the design	"Bolt-On" security; was of the lowest concern and is slowly being added in
Communication Protocols	Standardized set of protocols and ports, align to OSI model	Hundreds of proprietary protocols, with a few standard protocols
Lifecycle	Typically, 3–5-year tech refresh lifecycle; not difficult to replace	Long lifecycle; designed to last for 10+ years and often expensive to replace

Table 1 Generalized IT and OT Differences

Purdue Model

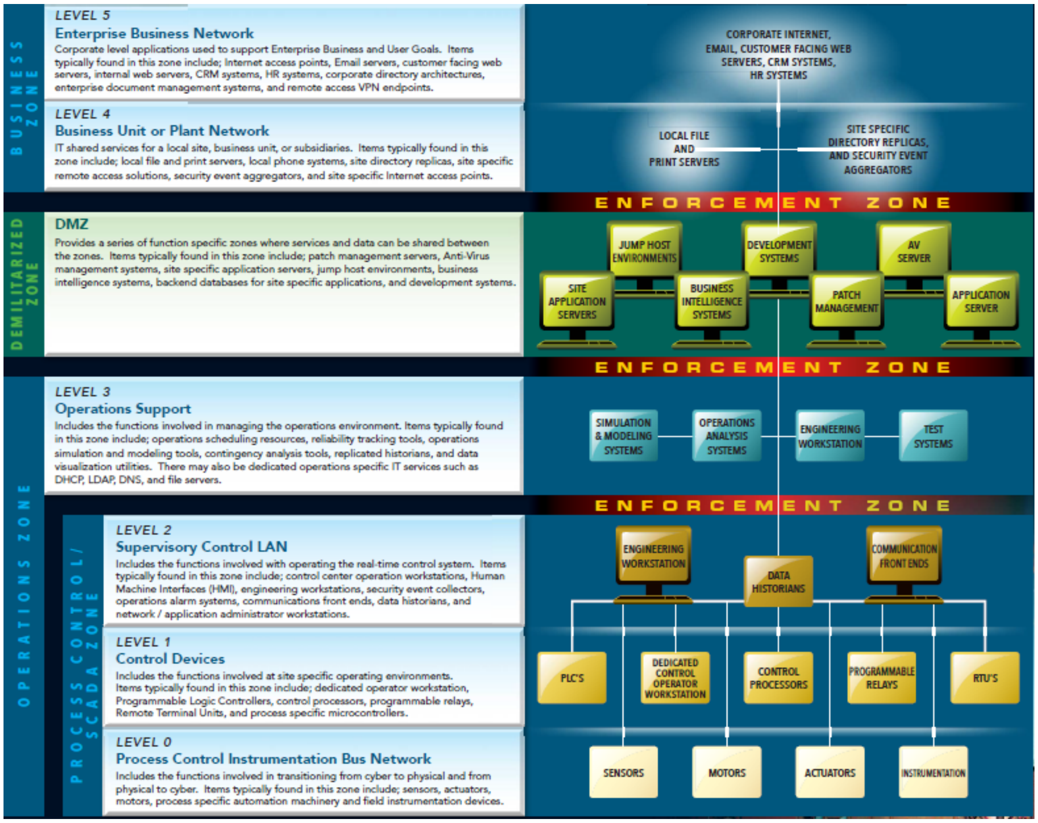
The Purdue Model was created to define best practices for the relationship between industrial control systems (OT) and business networks (IT) (Mathezer, 2021). The model has grown over time to include guidance for the architecture and the systems and technologies that reside at each level (Mathezer, 2021). There are 6 levels and as you move down the hierarchy, from Level 5 to Level 0, devices have more access to critical processes but fewer intrinsic security capabilities (Mathezer, 2021). Devices at lower levels are therefore more reliant on network and architectural defenses found in the upper levels to protect them.

Levels 4 and 5 encompass the "Enterprise Zone" or the typical IT network that is required for the business. While "technically not part of the ICS, the enterprise zone relies on connectivity with the ICS networks to feed the data that drives the business decisions" (Mathezer, 2021). Level 5 is typically representative of the corporate level and potentially spans multiple facilities or plants. Level 4 is a smaller representation of Level 5 and contains all the IT systems that support a local production process in a plant or facility (Ackerman, 2017).

Between the Enterprise Zone and the Manufacturing Zone, which contains Levels 0-3, there is the Industrial Demilitarized Zone (DMZ). The DMZ is an information sharing barrier and prevents direct communication between IT and OT systems. Systems in lower levels send production data to data collection and aggregation servers in this level, which can then send the data to higher levels or can be queried by systems in higher levels (push versus pull operations) (Ackerman, 2017).

The Manufacturing Zone, also sometimes called the Industrial Security Zone or Operations Zone, is where the action is and where the heart of the Industrial Control Network takes place. (Ackerman, 2017). These levels include the OT equipment and controls that get down to the process level. It encompasses local site operations, supervisory control of systems and finally, the process itself. Systems in this zone include human-machine interfaces (HMIs), PLCs, engineering workstations, supervisory control systems, valves, sensors, and actuators. Level 3 is the beginning of the OT layers containing everything necessary for managing control plant operations but may still include local installations of IT services such as DNS, Active Directory, or DHCP. Levels 0-2 are sometimes referred to as the Cell/Area Zone and are responsible for process specific operations. Level 2 handles the supervisory control such as HMIs or engineering workstations, Level 1 contains the control devices such as the PLCs or RTUs, and the field devices such as sensors and actuators are in Level 0.

ICS Threat Hunting focuses on targeting the Manufacturing/Operations Zone.



LEVEL 5
Enterprise Business Network
Corporate level applications used to support Enterprise Business and User Goals. Items typically found in this zone include; Internet access points, Email servers, customer facing web servers, internal web servers, CRM systems, HR systems, corporate directory architectures, enterprise document management systems, and remote access VPN endpoints.

LEVEL 4
Business Unit or Plant Network
IT shared services for a local site, business unit, or subsidiaries. Items typically found in this zone include; local file and print servers, local phone systems, site directory replicas, site specific remote access solutions, security event aggregators, and site specific Internet access points.

DMZ
Provides a series of function specific zones where services and data can be shared between the zones. Items typically found in this zone include; patch management servers, Anti-Virus management systems, site specific application servers, jump host environments, business intelligence systems, backend databases for site specific applications, and development systems.

LEVEL 3
Operations Support
Includes the functions involved in managing the operations environment. Items typically found in this zone include; operations scheduling resources, reliability tracking tools, operations simulation and modeling tools, contingency analysis tools, replicated historians, and data visualization utilities. There may also be dedicated operations specific IT services such as DHCP, LDAP, DNS, and file servers.

LEVEL 2
Supervisory Control LAN
Includes the functions involved with operating the real-time control system. Items typically found in this zone include; control center operation workstations, Human Machine Interfaces (HMI), engineering workstations, security event collectors, operations alarm systems, communications front ends, data historians, and network / application administrator workstations.

LEVEL 1
Control Devices
Includes the functions involved at site specific operating environments. Items typically found in this zone include; dedicated operator workstation, Programmable Logic Controllers, control processors, programmable relays, Remote Terminal Units, and process specific microcontrollers.

LEVEL 0
Process Control Instrumentation Bus Network
Includes the functions involved in transitioning from cyber to physical and from physical to cyber. Items typically found in this zone include; sensors, actuators, motors, process specific automation machinery and field instrumentation devices.

Figure 1 Purdue Reference Model (SANS Poster)

Best Practices for ICS Asset Owners

When making recommendations to the ICS owners after threat hunting, there are some standard Industry-Recommended best practices. The first one is always to add a centralized network security monitoring solution. You can't find anything if you are not looking, and there are tools that are designed specifically for passive network monitoring of ICS. One such example is Malcolm, developed by CISA and is freely available for download. Along the same lines, ensure that you know what is supposed to be on your network and maintain thorough documentation of all devices. A good method for determining the key items is Crown Jewel Analysis.

An OT network should be completely isolated from the IT layer either physically or logically. There is a trend of increasing connectivity between OT and IT due to the growing Internet of Things and the desire for product owners to get regular reports and updates on the systems. While they can be connected, ensure there are only uni-directional gateways between the layers to allow only select traffic through and ensure it is not a pivot point for attackers. Similarly, all network connections should be minimized to only those that are necessary and remote access should be limited to specific accounts, permissions, and computers, or eliminated if possible.

One of the most common findings is the use of insecure passwords, default accounts, or shared logins. While it is convenient to just leave the computer up and logged in, and allow everyone access, that also makes it easier for an attacker to get in, and there is no way to track who is conducting which actions.

It is difficult to patch ICS components without affecting availability or due to fears that it will break the system. There needs to be some sort of vulnerability mitigation plan in place to address identified high-risk vulnerabilities and a plan to test and apply patches in a safe manner. This plan should result in making it harder for the attacker to get in, or if they do get in, it is harder for them to take actions and they are seen.

Best Practices for ICS Threat Hunters

As a threat hunter, you have a responsibility to protect the network you are hunting on, from both your actions and from the actions of malicious cyber actors. The biggest thing you can do to prepare for this is spend time to increase your knowledge on ICS networks, and those that like to attack them. You need to be able to think like an attacker in order to find an attacker. Spend time learning attack methodology, and stay up to date on the latest threats, malware, and attack vectors occurring on ICS networks.

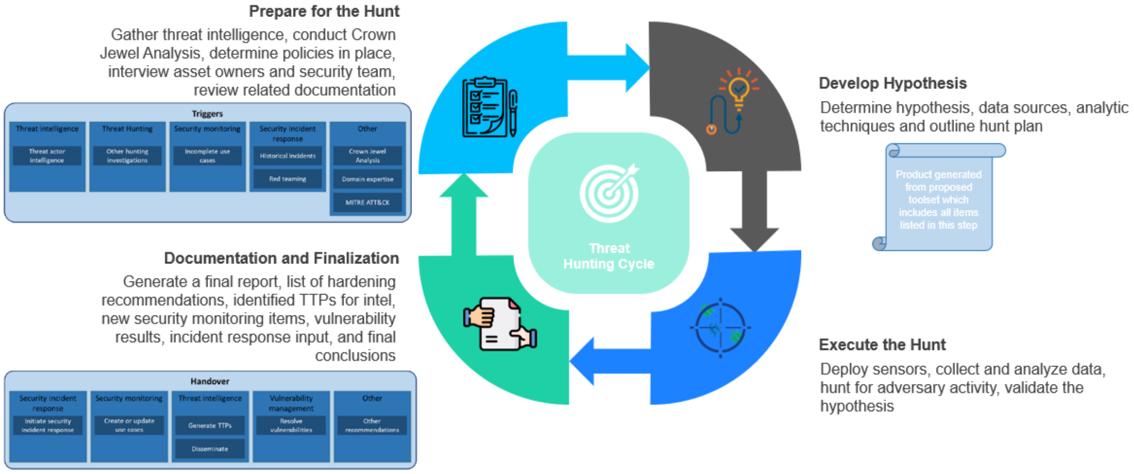
Take the time to increase your knowledge on all terms, components, protocols, and practices related to ICS. The engineers you talk to are going to be very knowledgeable about the components they are responsible for, and you want to ensure that you are able to clearly discuss what you want to do with them and use the proper terminology.

Finally, ICS threat hunting is not done in a vacuum, and work to collaborate with other teams who also conduct ICS threat hunting, to learn more best practices and lessons learned from them.

Framework for Threat Hunting

While there are a variety of frameworks for threat hunting, there is no set standard for a framework. A few have been proposed for Enterprise Threat Hunting, but none exist for ICS Threat Hunting.

The proposed framework for this paper, includes four phases based off the recommendations from above. Phase One is preparation for the hunt. This includes threat intelligence gathering, understanding of the environment, and determining the information you would need to hunt on the terrain. Phase Two is development of the hypothesis, the hunt plan, and a data collection plan. Phase Three is execution of the hunt, where you would deploy sensors, collect and analyze the gathered data, hunt for adversary activity, and attempt to validate the hypothesis. Finally, Phase Four is the documentation and finalization of the hunt, feeding back into further preparation for a future hunt.



Prepare for the Hunt

Gather threat intelligence, conduct Crown Jewel Analysis, determine policies in place, interview asset owners and security team, review related documentation

Develop Hypothesis

Determine hypothesis, data sources, analytic techniques and outline hunt plan

Execute the Hunt

Deploy sensors, collect and analyze data, hunt for adversary activity, validate the hypothesis

Documentation and Finalization

Generate a final report, list of hardening recommendations, identified TTPs for intel, new security monitoring items, vulnerability results, incident response input, and final conclusions

Figure 2 Proposed Threat Hunting Framework

ATT&CK for ICS

The MITRE ATT&CK for ICS Matrix is similar to the standard MITRE ATT&CK matrix but customized for the ICS knowledge base. According to the creators, it stemmed from a need to better understand, concentrate, and disseminate knowledge about adversary behavior in the ICS technology domain (Daszcyszak et al., 2019). Initial stages of attacks that involved IT infrastructure could be mapped to the Enterprise knowledge base, but later stages of the attack were out of scope of the Enterprise ATT&CK. "ATT&CK for ICS seeks to fill this gap, adversary behavior out-of-scope of non-ICS technology domains and address the unique concerns of the ICS domain" (Daszcyszak et al., 2019). If you are unfamiliar with ATT&CK for ICS, I recommend reading the ATT&CK for ICS Philosophy Paper on MITRE.

Tactics

"Tactics represent the "why" of an ATT&CK technique. They are the adversary's tactical objective: the reason for performing an action" (Daszcyszak et al., 2019). There is some overlap between ICS ATT&CK tactics and Enterprise tactics, however, there is also the addition of some categories that are not found in any of the other domains.

Techniques

Techniques represent 'how' an adversary achieves a tactical objective by performing an action, 'what' an adversary gains by performing an action, or 'the consequence' an adversary causes by performing an action. (Daszcyszak et al., 2019)

Data Sources

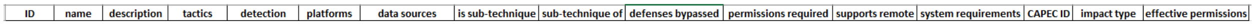
The data source listing is the "source of information collected by a sensor or logging system, e.g., packet capture, file monitoring, that can be used to obtain relevant information for identifying the action being performed, sequence of events, or the results of the actions by an adversary, including the state of systems and processes. The data source list can incorporate different variations of how the action could be performed across different assets for a particular technique" (Daszcyszak et al., 2019). The data sources are listed using vague terms, so the threat hunter playbook converts those into relevant logs and items that a threat hunter should collect to identify certain techniques.

Threat Groups

Named adversary groups are "associations of related intrusion activity that the security community has tracked under a common name or reliably associated across multiple incidents to specific threats. By focusing on the adversary groups which have or are actively targeting ICS and those responsible for ICS can better identify the most pertinent threats to them" (Daszcyszak et al., 2019). One well-known group is Sandworm Team, who were responsible for the 2015 and 2016 attacks on the Ukrainian power grid. Dragos Threat Intelligence has created "[their own] profiles of known [threat]groups targeting ICS environments to provide industrial defenders with context on behaviors that can signal evidence of a potential cyberattack" (Dragos Inc., 2020). Dragos outlines 19 different threat groups based on industry, location, and tactics used by the group. Dragos is an example of one threat intelligence team, but other teams also have different names and groupings for threat groups. Examples of this include MITRE, CrowdStrike, and RecordedFuture, although there are many others. I have chosen to work with the MITRE and Dragos threat groups as they are readily viewable and tried to cross correlate them with other names when possible.

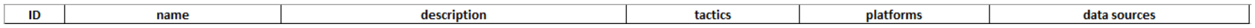
What's Missing

If you are to compare the Enterprise techniques and the ICS techniques, you may notice that for Enterprise, detection methods are clearly outlined. While no specific commands are mentioned, a broad description of what to look for can easily guide an operator to specific commands in conjunction with the needed logs/data sources. A review of the relevant categories of information available results in this table below, for Enterprise:



ID	name	description	tactics	detection	platforms	data sources	is sub-technique	sub-technique of	defenses bypassed	permissions required	supports remote	system requirements	CAPEC ID	impact type	effective permissions

Not every one of these items is 100% filled in, but it gives an operator a very large starting point from which to begin hunting. Also, depending on the technique type, some of the categories may not be relevant which explains some of the blank spaces. In comparison, the ICS techniques only offer a small subset of categories of information:



ID	name	description	tactics	platforms	data sources

There is a large gap in coverage regarding detection methods and relevant hunting information on the ICS Matrix. This gap is one of the major areas the Threat Hunter Playbook seeks to fill. The starting format is via simple Excel sheets which are easy to manually review to build a hunt plan. The overall goal is to make it as easy as possible for the threat hunter to develop a hunt plan and outline verified detection methods for each potential technique. By publishing the spreadsheets to GitHub allows for the possibility of crowdsourcing additional detection methods and the ability to continue to fill in missing data to make threat hunting more reliable and detection doable.

Threat Hunter Playbook

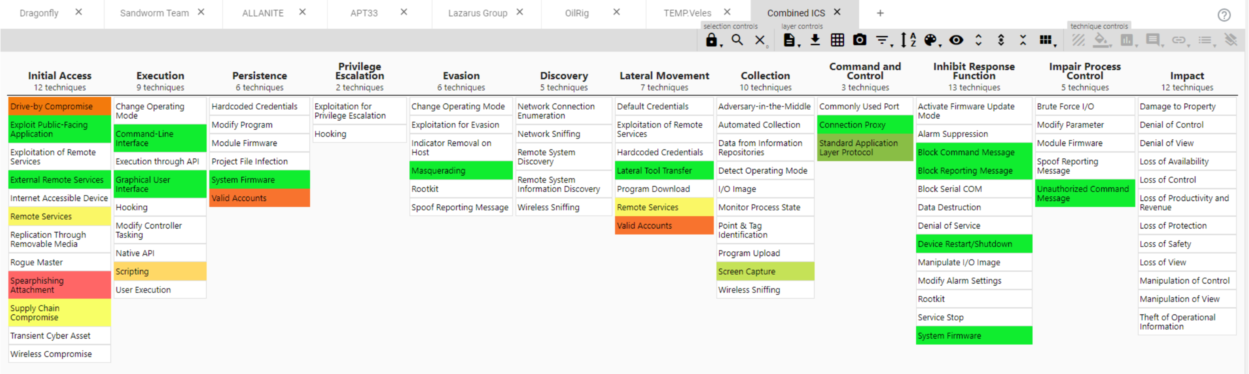
The THP is designed to combine the MITRE ATT&CK matrix with a hunt plan and recommended open-source tools. I am aware that certain SIEM or purchased security platforms contain information like this, but they tend to be platform-specific and you must be a paying customer to access the information. Not all companies are able to implement or afford one of those platforms, so this represents a starting point for companies or a useful tool for threat hunters who may be deployed to a location where they have no sensors or platforms already in place.

Mapping out Coverages

It is important to map out both adversary and defender coverage onto the ATT&CK matrices. ATT&CK navigator provides one method for mapping adversary coverage, but it doesn't capture all areas that I would like when I am preparing for my threat hunt. It is very good at what it does, but it was not designed to aid threat hunters.

Currently, in the ATT&CK navigator, you can create a layer based on Threat Groups, but there is no way to adjust the gradient based on knowledge of the terrain, adversary likelihood, or additional knowledge of the threat group showing their more likely methods. It is also a MITRE product, so it is limited to only MITRE threat groups. These threat groups often overlap with other threat intel sources and there may be different names or additional threat information that is not tracked by MITRE.

ATT&CK navigator is limited in its abilities to show multiple threat groups at one time, and when trying to examine multiple, the overlap areas can become messy. The frequency that each tactic is selected is shown changes, but all the threat groups blend together, and there is no variance based on the other threat group limitations I mentioned.



Initial Access 12 techniques	Execution 9 techniques	Persistence 6 techniques	Privilege Escalation 2 techniques	Evasion 6 techniques	Discovery 5 techniques	Lateral Movement 7 techniques	Collection 10 techniques	Command and Control 3 techniques	Inhibit Response Function 13 techniques	Impair Process Control 5 techniques	Impact 12 techniques
Drive-by Compromise	Change Operating Mode	Hardcoded Credentials	Exploitation for Privilege Escalation	Change Operating Mode	Network Connection Enumeration	Default Credentials	Adversary-in-the-Middle	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Exploit Public-Facing Application	Command-Line Interface	Modify Program	Hooking	Exploitation for Evasion	Network Sniffing	Exploitation of Remote Services	Automated Collection	Connection Proxy	Alarm Suppression	Modify Parameter	Denial of Control
Exploitation of Remote Services	Execution through API	Module Firmware		Indicator Removal on Host	Remote System Discovery	Hardcoded Credentials	Data from Information Repositories	Standard Application Layer Protocol	Block Command Message	Module Firmware	Denial of View
External Remote Services	Graphical User Interface	Project File Infection		Masquerading	Remote System Information Discovery	Lateral Tool Transfer	Detect Operating Mode		Block Reporting Message	Spoof Reporting Message	Loss of Availability
Internet Accessible Device	Hooking	System Firmware		Rootkit	Wireless Sniffing	Program Download	I/O Image		Block Serial COM	Unauthorized Command Message	Loss of Control
Remote Services	Modify Controller Tasking	Valid Accounts		Spoof Reporting Message		Remote Services	Monitor Process State		Data Destruction		Loss of Productivity and Revenue
Replication Through Removable Media	Native API					Valid Accounts	Point & Tag Identification		Denial of Service		Loss of Protection
Rogue Master	Scripting						Program Upload		Device Restart/Shutdown		Loss of Safety
Spearphishing Attachment	User Execution						Screen Capture		Manipulate I/O Image		Loss of View
Supply Chain Compromise							Wireless Sniffing		Modify Alarm Settings		Manipulation of Control
Transient Cyber Asset									Rootkit		Manipulation of View
Wireless Compromise									Service Stop		Theft of Operational Information
									System Firmware		

Figure 3 MITRE ATT&CK Navigator Combined ICS Rendering

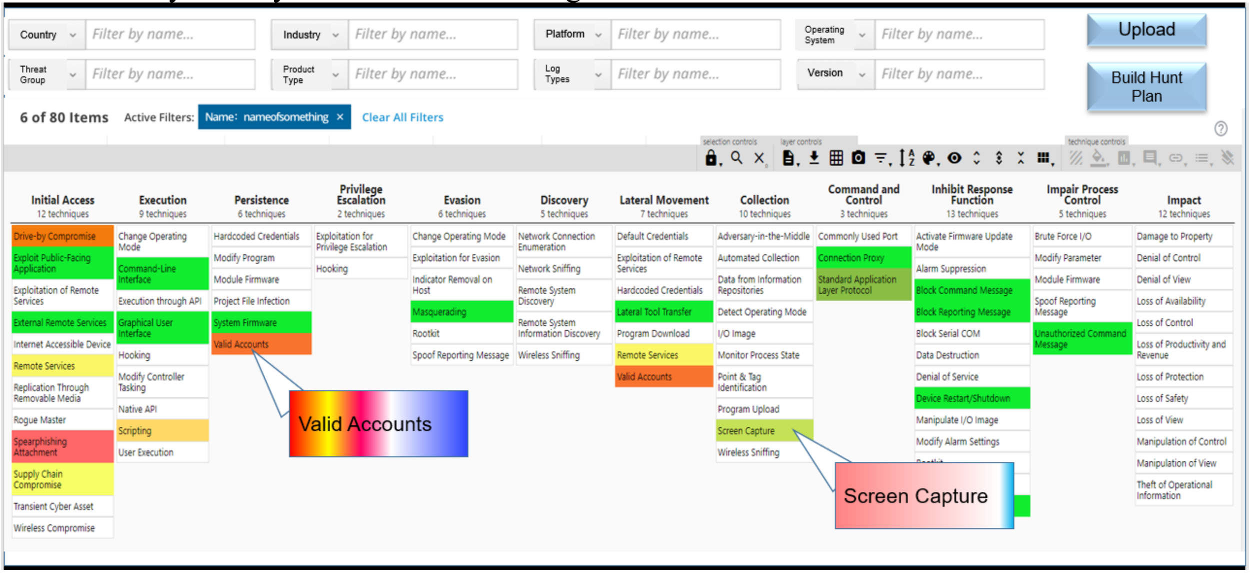
Finally, there is no ability to input defender capabilities, to see what you are able to detect and defend against, to find any gaps in your defense capabilities.

I wanted to create a way to combine the information I described in the last slide and to view the information most relevant to your terrain. Coverage should be able to be filtered by terrain, operating system, protocol types, country, industry, etc. This can expand past ICS, but I started with ICS since that is my focus area and because the MITRE matrix for ICS is smaller to experiment with. This should eventually be able to view both Enterprise and ICS Matrices for ICS attacks, since not all ICS-based attacks include only ICS equipment. The mapping of attacker coverage can be visualized in color with shading for likelihood/skills/previous attacks and cross-correlated with defender coverage that gets visualized in shades of blue in a gradient format. Defender coverage is based on the recommended data sets, if you can collect items for coverage, toolsets available, etc. It does not consider defender abilities, just toolset coverage of techniques. The key areas here are the ability to filter queries and limit all the areas you want to view to just the results relevant to your terrain and environment.

See Figure 4 below for an example of the prototype for the Threat Hunter Playbook display. We can start by examining Screen Capture as an example. Only one adversary out of the

5 possible uses this technique and they don't use it very often. We also don't have a very good way to detect this.

Then if we compare that to valid accounts, 3 out of the 5 adversaries use this frequently (and we know which 3, because each adversary is assigned a color). However, we have a lot of logs covering this technique, and we can state that it is reasonable for our defenders to be able to find adversary activity based on available logs.



6 of 80 Items Active Filters: Name: nameofsomething × Clear All Filters

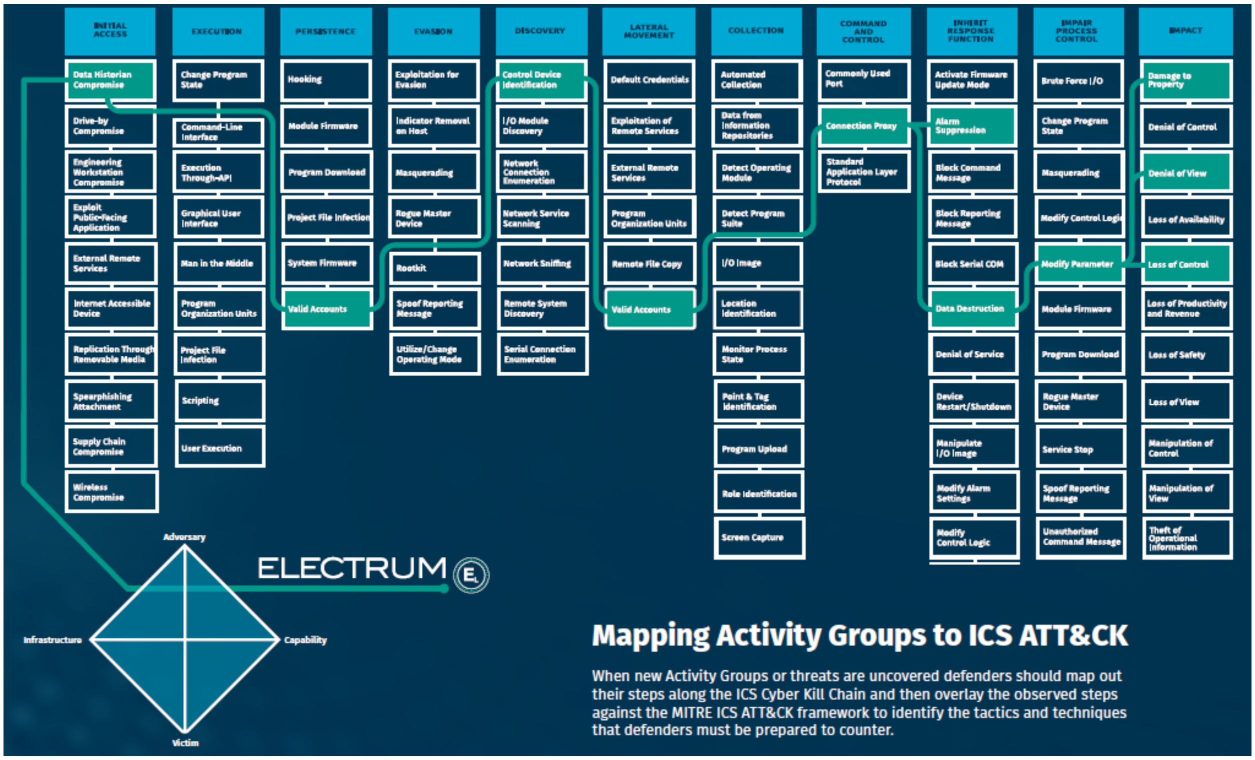
Initial Access 12 techniques	Execution 9 techniques	Persistence 6 techniques	Privilege Escalation 2 techniques	Evasion 6 techniques	Discovery 5 techniques	Lateral Movement 7 techniques	Collection 10 techniques	Command and Control 3 techniques	Inhibit Response Function 13 techniques	Impair Process Control 5 techniques	Impact 12 techniques
Drive-by Compromise	Change Operating Mode	Hardcoded Credentials	Exploitation for Privilege Escalation	Change Operating Mode	Network Connection Enumeration	Default Credentials	Adversary-in-the-Middle	Commonly Used Port	Activate Firmware Update Mode	Brute Force I/O	Damage to Property
Exploit Public-Facing Application	Command-Line Interface	Modify Program	Hooking	Exploitation for Evasion	Network Sniffing	Exploitation of Remote Services	Automated Collection	Connection Proxy	Alarm Suppression	Modify Parameter	Denial of Control
Exploitation of Remote Services	Execution through API	Module Firmware		Indicator Removal on Host	Remote System Discovery	Hardcoded Credentials	Data from Information Repositories	Standard Application Layer Protocol	Block Command Message	Module Firmware	Denial of View
External Remote Services	Graphical User Interface	Project File Infection		Masquerading	Remote System Information Discovery	Lateral Tool Transfer	Detect Operating Mode		Block Reporting Message	Spoof Reporting Message	Loss of Availability
Internet Accessible Device	Hooking	System Firmware		Rootkit	Wireless Sniffing	Program Download	I/O Image		Block Serial COM	Unauthorized Command Message	Loss of Control
Remote Services	Modify Controller Tasking	Valid Accounts		Spoof Reporting Message		Remote Services	Monitor Process State		Data Destruction		Loss of Productivity and Revenue
Replication Through Removable Media	Native API					Valid Accounts	Point & Tag Identification		Denial of Service		Loss of Protection
Rogue Master	Scripting						Program Upload		Device Restart/Shutdown		Loss of Safety
Spearphishing Attachment	User Execution						Screen Capture		Manipulate I/O Image		Loss of View
Supply Chain Compromise							Wireless Sniffing		Modify Alarm Settings		Manipulation of Control
Transient Cyber Asset									Rootkit		Manipulation of View
Wireless Compromise											Theft of Operational Information

Figure 4 Threat Hunter Playbook Prototype Rendering

Spider Charts

The team I work with maps out the potential Adversary Scheme of Maneuver against ATT&CK in an interconnected web or path of events, which some might refer to as a "spider chart", based off known tactics of APTs. They create the "kill chain" across with the matrix for the most likely enemy actions, and the most dangerous enemy actions, which potentially occurs a few times depending on the number of likely threat actors. Dragos performs similar mappings of Activity Groups to the ICS ATT&CK Matrix, and this visualization of the kill chain feeds directly into which behavioral TTPs need to be focused on by defenders.

The analyst can first filter the potential threat groups based on country, industry, platforms, etc., as we saw on the previous slide, and then trace out potential spider charts for the different threat actors. Selecting the potential adversary scheme of maneuver selects specific T-codes for analysis, which are then fed into generating the hunt plan.



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 Module	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 Suite		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

Mapping Activity Groups to ICS ATT&CK

When new Activity Groups or threats are uncovered defenders should map out their steps along the ICS Cyber Kill Chain and then overlay the observed steps against the MITRE ICS ATT&CK framework to identify the tactics and techniques that defenders must be prepared to counter.

Figure 2 Dragos Sample of Mapping (Dragos Inc., 2020)

Detection Methods

The backend of the tool outlines detection methods for each T-Code. As I mentioned when describing what was missing for the ICS ATT&CK matrix, these pictures are samples of the built out excel sheets, that outline detection information for each T-code, that will then be fed into the hunt plan. These are based on the available logs and data collected and recommend specific commands or queries to run either in ELK or other tools to help find behavioral based TTPs. This has been a community collaboration to build out methods that have worked to find adversary activity. The goal is that this community collaboration continues to fill in and expand upon capabilities for detection.

Building out a Hunt Plan

The hunt plan itself is the crux of the tool. There are many different recommended "threat hunt plans" that can be found on the internet, however the one I am presenting is one that has worked for our team and has been refined after multiple different hunts. We used to have a plan that was dozens of pages in "Outline" format for each T-code and each T-code got repeated multiple times for each tactic. It was very long and finding the actual "hunt" items got lost in the plan. The new format is tabular and consolidates the hunt for each T-code. Each T-code, based off the hypothesized adversary scheme of maneuver, contains a detection overview, a list of recommended data sources for collection, host and network rules, commands, and hunting methods, and finally contains a list of available analytics. The analytics portion is limited to the capabilities of your team; however, it references all MITRE CAR pseudocode analytics that could be developed for the hunt. Finally, it lists recommended tools for collecting, hunting, and analyzing that T-code.

T-Code: T0859
Associated Tactics: Persistence, Lateral Movement

Detection Overview: Monitor for logon behavior that may abuse credentials of existing accounts as a means of gaining Lateral Movement or Persistence. Correlate other security systems with login information (e.g., a user has an active login session but has not entered the building or does not have VPN access). Monitor for suspicious account behavior across systems that share accounts, either user, admin, or service accounts. Examples: one account logged into multiple systems simultaneously; multiple accounts logged into the same machine simultaneously; accounts logged in at odd times or outside of business hours. Activity may be from interactive login sessions or process ownership from accounts being used to execute binaries on a remote system as a particular account. Monitor for an authentication attempt by a user that may obtain and abuse credentials of existing accounts as a means of gaining Initial Access, Persistence, Privilege Escalation, or Defense Evasion.

Host Queries/Commands	**Network Queries/Commands**
Kibana: event.code:4624 or event.code:4625 or event.code:4648 or event.code:4672 or event.code:4720 or event.code:4724 or event.code:4776 or event.code:4778 or event.code:4779 or event.code:4798 or event.code:4799 **PowerShell:** Get-EventLog -LogName Security -InstanceId 4624 -After “<INSERT DATE>” | Where {$_.Message -match “Logon Type:\s+3”} | Export-Csv <output.csv>	**Arkime:** protocols == krb5 && (krb5.realm == EXISTS! || krb5.cname == EXISTS! || krb5.sname == EXISTS!) protocols == dcerpc && dcerpc.cmd == "*Log*" **Kibana:** protocol: "krb5" AND (krb5.cname: * OR krb5.sname: *) protocol:"dcerpc" and dcerpc.cmd: *Log*
Tools Required:	**Data Collection Requirements:**
Kibana Powershell Metasponse Arkime	Security Log (EID 4624, 4634, 4672, 4625, 5379) /var/log/auth.log /var/log/utmp /var/log/wmtp

Analytics:
MITRE CAR: Simultaneous Logins on a Host, User Logged into Multiple Hosts, SMB Write Request, SMB Copy and Execution, User Login Activity Monitoring, All Logins since Last Boot
Valid Accounts

Figure 5 Sample Format for Output Hunt Plan

Tools used for "Plays"

Finally, I want to highlight some tools that can be used to augment threat hunting capabilities and are freely available open source. Appendix A contains a list of references to where these tools and capabilities can be found online for teams to access. The best tool is the one that your team is familiar with and feels comfortable using, that is safe and effective for the terrain it is being used against. There is no one-tool-fits-all and it is best to use a combination of tools that provide broad coverage and detection capabilities to find the adversary.

Operating Systems

The operating system you choose is a personal choice, however there are some operating systems that are specifically designed for cybersecurity. Security Onion is my teams OS of choice on mission because it contains many tools we use, and the distributed collection capabilities deploy very nicely. There are also two other defensive based operating systems that I can recommend: Parrot Security OS and the new Kali "Purple" operating system. All these systems are Linux based and have the ability to easily add tools needed for threat hunting. Finally, while not an operating system, Malcolm is a new network monitoring tool that is highly recommended and built specifically for ICS and should be added on to Parrot Security or Kali Purple to capture the logs and data in a way that is easily reviewable for ICS security.

YARA Rules

YARA is a tool that is used to help malware researchers identify and classify malware samples (Alvarez, n.d.). The rules can be used to create descriptions of malware families based on textual or binary patterns. Each rule contains a set of strings and a Boolean expression which determines its logic (Alvarez, n.d.). They can vary in complexity, run on multiple-platforms, or augmented with Python scripts. Various YARA rules have already been created for Havex malware, Stuxnet, Triton/TRISIS, CrashOverride, BlackEnergy, and some APTs. These rules can help safely scan devices for indicators of compromise, and potentially point your team towards new findings.

Snort/Suricata Rules

Snort and Suricata are open-source Intrusion Detection Systems, and which one you choose is based on your preferences, and potentially what is included with the operating system chosen above. They both have a substantial ruleset and community added rules that can be freely downloaded and implemented. Most rules between Snort and Suricata are interchangeable, with the exception of Talos rules. Talos rules only work in Snort. These tools are useful as an alert system but do need to be tuned to ensure there is not alert overload. Some custom ICS-specific rules have been loaded into GitHub and can provide a good starting point for generating alerts. These are not a catch-all and do not identify all possible threats and may also alert on benign activity.

Scripts

Scripts can be run on your analyst system against the collected data and run a series of instructions to process the data or to automate batch processes. NMAP also contains the potential to run scripts for additional information, and because it can be dangerous to run NMAP against ICS assets, there are special scripts that work with NMAP specifically designed for ICS components. Most of the open-source scripts listed in Appendix A refer to enumeration and discovery for ICS assets, which can contribute to developing a greater understanding of the environment you are analyzing. Other types of scripts can include developed analytics run against the data to look for anomalies or specific behavior or IOC types.

Threat Hunting Tools

There are a variety of tools available to augment your team and support the hunt. Research is required to determine exactly which tools will work best for you. Appendix A contains a reference to GRFICS, which is five virtual machines to create your own ICS security lab. This can be used to test new tools and their affects on the ICS components, as well as get more familiar with the systems. Appendix A also contains links to repositories with large collections of tools, all designed specifically for ICS/SCADA architecture. Your team should experiment with the available tools and create a custom tool suite that you use for every hunt.

Conclusion

In conclusion, threat hunting is a proactive process looking for evidence of adversary activity based around a hypothesis of predicted activity. A hunt plan should be used to guide a threat hunting teams actions and what behavioral TTPs to target. The Threat Hunter Playbook outlines a tool of reference that generates the hunt plan for teams, and guides threat hunting teams on their course of action, with the goal of countering the enemy as far left as possible.

Defense is not a solo activity, and protecting our critical infrastructure is of utmost importance in the continuous functioning of our society. The Threat Hunter Playbook looks to codify some standard practices and lower the bar of entry for threat hunting teams.

References

116 Congress. (2020, December 4). *Public law 116–321 116th Congress an act - govinfo*. Internet of Things Cybersecurity Improvement Act. Retrieved from https://www.govinfo.gov/content/pkg/PLAW-116publ321/pdf/PLAW-116publ321.pdf

Ackerman, P. (2017). *The Purdue model for Industrial control systems*. Industrial Cybersecurity. Retrieved from https://subscription.packtpub.com/book/networking-&-servers/9781788395151/1/ch01lvl1sec10/the-purdue-model-for-industrial-control-systems

Alvarez, V. (n.d.) *Yara: The pattern matching swiss knife for malware researchers*. VirusTotal. https://virustotal.github.io/yara/

Arafune, M., Rajalakshmi, S., Jaldon, L., Jadidi, Z., Pal, S., Foo, E., & Venkatachalam, N. (2022). Design and development of automated threat hunting in Industrial Control Systems. *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops)*. https://doi.org/10.1109/percomworkshops53856.2022.9767375

Chen, L., Jiang, R., Lin, C., & Li, A. (2022). A survey on threat hunting: Approaches and applications. *2022 7th IEEE International Conference on Data Science in Cyberspace (DSC)*, 340–344. https://doi.org/10.1109/dsc55868.2022.00053

Daszcyszak, R., Ellis, D., Luke, S., & Whitley, S. (2019). (tech.). *TTP-Based Hunting* (pp. 1–32). Annapolis Junction, MD: MITRE.

Databasix UK. (2023). *20 frightening cyber security facts and stats*. Databasix UK Ltd. Retrieved from https://www.dbxuk.com/statistics/cyber-security

de Volksbank, R. van O., Bakker, M., Bouman, R., Doctors van Leeuwen, M., van der Kraan, M., Mentges, W., & Piers, A. (2019). (tech.). *TaHiTI: a threat hunting methodology*. FI-ISAC NL Publication. Retrieved from https://www.betaalvereniging.nl/wp-content/uploads/DEF-TaHiTI-Threat-Hunting-Methodology.pdf.

Dragos, Inc. (2020). (rep.). *Mapping Industrial Cybersecurity Threats to MITRE ATT&CK for ICS* (pp. 1–15). Hanover, MD: Dragos.

Hazel, T. (2022, September 12). *Threat hunting frameworks and methodologies*. ChaosSearch. Retrieved from https://www.chaossearch.io/blog/threat-hunting-methods-and-frameworks

Jadidi, Z., & Lu, Y. (2021). A threat hunting framework for industrial control systems. *IEEE Access*, *9*, 164118–164130. https://doi.org/10.1109/access.2021.3133260

Lee, R., & Lee, R. M. (2017). (rep.). *The Hunter Strikes Back: The SANS2017 Threat Hunting Survey* (pp. 1–25). Rockville, MD: SANS Institute.

Mathezer, S. (2021, December 8). *Introduction to ICS Security Part 2*. SANS Institute. Retrieved from https://www.sans.org/blog/introduction-to-ics-security-part-2/

Sobers, R. (2022, August 3). *166 cybersecurity statistics and trends [updated 2022]*. Varonis. Retrieved from https://www.varonis.com/blog/cybersecurity-statistics

TactiKoolSec. (2022, August 22). *Open Threat Hunting Framework*. GitHub. Retrieved from https://github.com/TactiKoolSec/OTHF

Taschler, S. (2022, August 30). *What is cyber threat hunting?* Cybersecurity 101. Retrieved from https://www.crowdstrike.com/cybersecurity-101/threat-hunting/

Yoo, G. (2021, January 21). *The importance of time and speed in cybersecurity*. Forbes. Retrieved from https://www.forbes.com/sites/forbestechcouncil/2021/01/22/the-importance-of-time-and-speed-in-cybersecurity/?sh=4348799636a9

Appendix A: Tools and Resources

Operating Systems

Security Onion: https://securityonionsolutions.com/ - free open platform for threat hunting, security monitoring, and log management

Parrot Security: https://www.parrotsec.org/ - Linux distro focused on security, forensics, privacy, and development

Kali Purple: https://www.kali.org/blog/kali-linux-2023-1-release/ - ultimate SOC-in-a-box community project

ICS Tools and Repositories

Malcolm: https://github.com/cisagov/Malcolm – Network traffic analysis tool suite

GrassMarlin: https://github.com/nsacyber/GRASSMARLIN - passively maps ICS networks

Ettercap: https://www.ettercap-project.org/

https://github.com/MDudek-ICS?tab=repositories – **Massive** Industrial Control Systems security related repositories collection

https://github.com/ITI/ICS-Security-Tools/tree/master/tools – repository of a variety of tools designed for ICS

https://github.com/paulveillard/cybersecurity-industrial-control-systems-security - collection of software, libraries, documents, books, and resources about industrial control systems

https://socprime.com/blog/siem-edr/threat-hunting-tools-our-recommendations/ - list of recommended threat hunting tools

https://securityboulevard.com/2022/03/5-best-threat-hunting-tools-for-your-security-team/ - list of 5 recommended tools for threat hunting

https://github.com/rmusser01/Infosec_Reference/blob/master/Draft/SCADA.md - contains articles, tools, simulators, and honeypots

YARA Rules

YARA tool: https://github.com/virustotal/yara/releases - yara download

https://virustotal.github.io/yara/ - yara facts

https://yara.readthedocs.io/en/stable/ - yara documentation

https://www.cisa.gov/uscert/sites/default/files/FactSheets/NCCIC%20ICS_FactSheet_YARA_S508C.pdf - NCCIC Factsheet

https://github.com/BayshoreNetworks/yextend - extension for yara to target zipped files

https://rhisac.org/threat-intelligence/new-cyber-tools-targeting-ics-scada-devices/ - ASRock driver exploit yara rules

https://github.com/dragosinc/CRASHOVERRIDE - crashoverride yara rules

https://github.com/MDudek-ICS/TRISIS-TRITON-HATMAN/blob/master/yara_rules/ics-cert.yara - triton yara rules

https://github.com/Yara-Rules/rules/tree/master/malware - APT list of multiple rules, includes Havex, Stuxnet, BlackEnergy, APT #'s, etc…

Snort/Suricata Rules

https://github.com/digitalbond/Quickdraw-Suricata - A set of ICS IDS rules for use with Suricata

https://github.com/digitalbond/Quickdraw-Snort - Digital Bond's IDS/IPS rules for ICS and ICS protocols

https://github.com/ITI/ICS-Security-Tools/blob/master/configurations/rules/talos-snort.rules – Snort and Talos specific rules for ICS
https://github.com/CyberICS/Suricata-Rules-for-ICS-SCADA - Suricata rule set to detect scan tools targeting PLC interfaces
https://suricon.net/wp-content/uploads/2017/12/SuriCon17-Stevens_Browning.pdf – PowerPoint containing Suricata basics and a few rule examples for ICS

Scripts
NMAP scripts: https://github.com/CyberICS/Nmap-script-SCADA - Nmap scripts for SCADA protocols
NMAP script: https://nmap.org/nsedoc/scripts/iec-identify.html - Attempts to identify IEC 60870-5-104 ICS protocol
NMAP ICS Tutorial: https://github.com/gnebbia/nmap_tutorial/blob/master/sections/ics_scada.md - provides tips and walkthrough for scanning ICS devices with NMAP
Python: https://github.com/ITI/ICS-Security-Tools/tree/master/scripts - various industrial security python scripts
ATT&CK: https://github.com/mitre-attack/attack-scripts/ - contains standalone scripts and utilities for working with ATT&CK

Exploitation (provided to test changes you have made, with permission from company you are evaluating)
Industrial Exploitation Framework: https://github.com/dark-lbp/isf - exploitation framework based on Python, like metasploit framework
ICSsploit: https://github.com/tijldeneut/icssploit - exploitation framework based on Python, it's similar to metasploit framework
ICS Pentesting Tools: https://github.com/kh4sh3i/ICS-Pentesting-Tools - curated list of tools related to ICS security and pentesting
MITRE Caldera: https://github.com/mitre/caldera - automated adversary emulation platform
Pentest Repository: https://github.com/enaqx/awesome-pentest - collection of penetration testing and offensive cybersecurity resources
Attack Graph Generator: https://github.com/mehgrmlhmpf/AttackGraphGeneratorMasterThesis – ICS purple teaming simulation

Other Tools
dnstwist: https://github.com/elceef/dnstwist – phishing domain scanner
exiftool: https://exiftool.org/ – tool for reading and working with PDF metadata
johntheripper: https://github.com/openwall/john - password cracker
MITRE Cascade: https://github.com/mitre/cascade-server - "blue-team" focused automated analytics server
snyk: https://snyk.io/ – software composition analysis tool
steghide: https://steghide.sourceforge.net/ – steganography engine
strings: https://linux.die.net/man/1/strings – finds and prints text strings embedded in files
volatility: https://github.com/volatilityfoundation/volatility – advanced memory forensics framework

Training

MITRE Training: https://attack.mitre.org/resources/training/ - MITRE Training guides and articles

MITRE ATT&CK Training: https://app.cybrary.it/browse/refined?q=Mitre%20ATT%26CK%20Defender – video training of MITRE; can be used in conjunction with MITRE ATT&CK Defender Certification

GRFICS: https://github.com/Fortiphyd/GRFICSv2 - contains 5 ICS VMs to create a training range

Additional Resources

SANS Blog Part 1: https://www.sans.org/blog/ics-threat-hunting-they-are-shootin-at-the-lights-part-1/

SANS Blog Part 2: https://www.sans.org/blog/ics-threat-hunting-they-are-shootin-at-the-lights-part-2/

Threat Hunting Blog: https://blog.cyberproof.com/blog/leveraging-threat-hunting-tools-to-improve-threat-detection-response

https://www.cisa.gov/sites/default/files/publications/2021-seminars-ics-security-508.pdf – ICS security seminar slides

CISA Advisory on APT Tools: https://www.cisa.gov/news-events/cybersecurity-advisories/aa22-103a – list of APT Cyber Tools Targeting ICS/SCADA Devices and mitigations to apply

CISA Advisories: https://www.cisa.gov/news-events/cybersecurity-advisories

PLC Security: https://plc-security.com/ - contains information on secure PLC coding practices and PLC security info

List of Resources: https://github.com/ics-iot-bootcamp/ICS_Awesome_List - list of communities, conferences, exercises, trainings, articles, and more

Default Passwords: https://github.com/arnaudsoullie/ics-default-passwords - default passwords for a few ICS systems

Scanner IPs: https://github.com/CyberICS/list_ics_scanner - list of ICS scanners