An Introduction to Applying Network Intrusion Detection for Industrial Control Systems

Tom Bartman and Jason Kraft
Schweitzer Engineering Laboratories, Inc.

Published in
Sensible Cybersecurity for Power Systems: A Collection of Technical Papers Representing Modern Solutions, 2018

Originally presented at
AISTech 2016 – The Iron & Steel Technology Conference and Exposition, May 2016

An Introduction to Applying Network Intrusion Detection for Industrial Control Systems

Tom Bartman[1], Jason Kraft[1]

[1] Schweitzer Engineering Laboratories, Inc.
2350 NE Hopkins Court, Pullman, WA 99163 USA
Phone: (509) 332-1890
Email: tom_bartman@selinc.com

Keywords: Cybersecurity, Computer, Power System, Protective Relay, Intrusion Detection.

INTRODUCTION

Industrial control systems (ICSs) and supervisory control and data acquisition (SCADA) networks are facing a growing number of threats, including malware and cyberattacks by nation states. As the power, automation, and industrial control industries transition from switched circuits to switched packet communications, SCADA networks and ICSs are becoming very popular targets of attacks. The ability to detect malicious code or an intrusion is just as important as implementing firewalls.

A network intrusion detection system (IDS) has become a very important piece of the security framework of an organization. It adds security controls not previously available and provides enhanced situational awareness within a single network segment. In addition to antivirus protection and firewalls on SCADA networks, a properly deployed, configured, and managed IDS adds the ability to detect if a network has been breached. An IDS monitors both inbound and outbound communications on a network and among devices, and it records events such as unauthorized access attempts, port scans, probes, buffer overflows, operating system (OS) fingerprinting, and other forms of attack. An IDS monitors activity from within a network by analyzing communications on the network segment that requires protection.

The primary purpose of an IDS is to identify and log incidents. It does this by analyzing data packets, detecting suspicious activity, and logging such activity. The benefit of an IDS is that it allows security professionals to detect and understand exploits and attacks on a network. An IDS also allows security professionals to establish a baseline of expected traffic and to obtain a record and notification when protocols and traffic patterns deviate from that baseline. For example, if DNP3/IP traffic should never appear on a network segment, a rule can be set to trigger an alert and allow for further investigation.

This paper provides a primer on intrusion detection, including an example application using an industrial computer. The paper covers the methods attackers use to penetrate networks and how to know when a network may have been compromised. The paper also discusses the proper methods of network segmentation and the deployment of an IDS.

CYBERATTACK RESULTS IN PHYSICAL DAMAGE TO STEEL PLANT

In 2014, a steel mill in Germany was the target of a cyberattack. The event was the second known cyberattack that resulted in physical damage to ICSs (the first was Stuxnet, which targeted the Iranian nuclear program). In the German steel mill incident, attackers used spear phishing emails to gain access to the steel mill office information technology (IT) network. Spear phishing is a method in which an attacker targets a specific individual with an email, and the email appears to be from someone that the recipient knows. The email typically contains a link to a malicious website or an infected attachment.

After the successful spear phishing attack provides access to the IT network, the attackers then moved to the production network, where critical assets reside. Finally, the attackers were able to control components in the plant, which resulted in an uncontrolled shutdown of a blast furnace and massive damage. According to a report from the German government, the attackers were skilled in both IT networks and ICSs[1].

ADVANCED PERSISTENT THREAT

The report states that the attack on the steel mill was an advanced persistent threat (APT). An APT occurs in stages that require attackers to be stealthy. An APT is targeted and usually involves many trained, well-equipped attackers skilled in avoiding detection with long-term harm in mind. According to the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT), APT attacks illustrate a concerted effort to gain access to critical infrastructure networks[2].

An APT is advanced because of how the intruder obtains access to the network. The example APT life cycle shown in Figure 1 occurs in several stages[3].

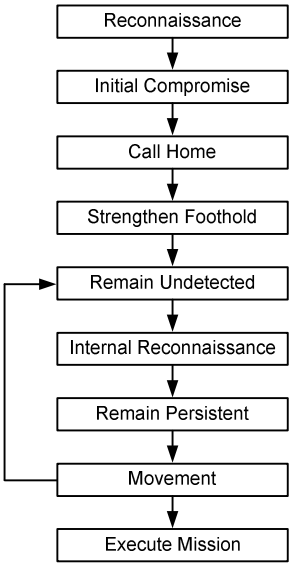

Figure 1. APT life cycle

First, the attacker carries out social engineering methods such as spear phishing. Spear phishing can occur in several ways. One method is for the attacker to develop a friendly relationship over time, perhaps via email, with the targeted user. Once trust is established with the intruder, the targeted user is more likely to share confidential information, such as a password or information about the company. Another method is the use of email that appears to be from a friend or known company. The email may contain a malicious attachment or link to a website. Once the attachment or link is opened, small pieces of code are copied to the recipient's computer.

The next step occurs when small pieces of malware or shell code are installed onto the target computer. Once the code is on the computer, it calls home (sends information to the attacker) and waits for further instructions. Next, objective-specific malware is installed remotely to the target computer. The attacker now waits (the persistence stage). One way persistence is achieved is by running "sleep" commands in between "run" commands to avoid detection. Being low and slow to avoid detection is the key to an APT. The final step is the objectives phase. This is when the attack, such as sabotage or data harvesting, is executed.

A targeted APT attack on a specific organization system indicates that the attacker knows the system being attacked. APTs are difficult to detect because of the nature of how the attacker targets specific users—by keeping a low profile and slowly attacking the system. One of the key points in the German steel plant attack is how the attackers first gained access to the office (IT) network and jumped to the industrial network, or operations technology (OT) network. This illustrates the need for strict separation between business and production networks to keep hackers from leaping from one network to another.

SEPARATION

According to a 2015 report published by the U.S. Department of Homeland Security, only 32 percent of electric utilities have integrated security systems with the "proper segmentation, monitoring, and redundancies" needed for cyberattack protection. Forty-eight percent of electric utilities said they did not[4].

Multilayer security allows for unique security controls at each layer, as shown in Figure 2. Critical assets are placed in the most reliable and secure layer. In most attacks, not only must an attacker compromise the perimeter, but the security controls at each additional layer must also be compromised to reach the target asset. Reliable and rugged Ethernet switches and security gateways allow for this type of layered framework in SCADA and ICS environments. However, ICSs with multiple layers of networking are not without risk. Alternative attack vectors, such as USB drives, can put an OT network at risk. Therefore, having multiple layers of defense plays an important role in cybersecurity infrastructure.

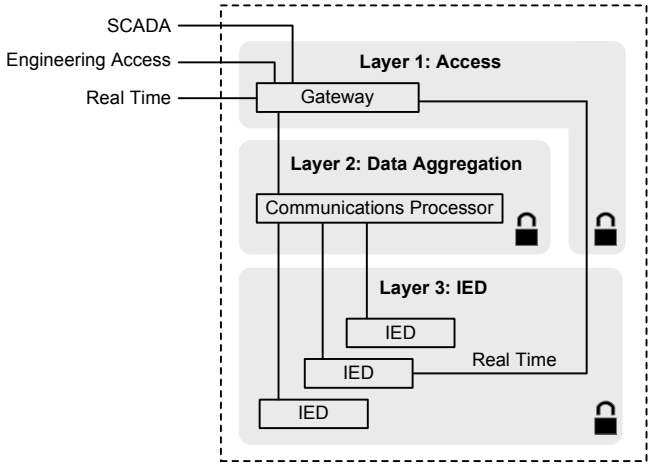

Figure 2. Multilayer security framework

In Figure 2, Layer 1 is the access zone where perimeter gateway security exists. The function of this zone is to manage access security. The firewall and port server at this level provide secure transport and manage traffic flows and encryption for machine-to-machine connections and human-to-machine interactions. The gateway secures the pathway between the industrial and central control networks. Also at this layer are firewall functions, virtual private network (VPN) and encryption technology, port server access, and SCADA rules and logging.

Layer 2 is the aggregation zone. Data concentration, switches, and controllers are at this layer, and this is where Ethernet port security, Media Access Control (MAC) filtering, virtual local-area networks (VLANs), and traffic policing occur. VLANs are used to restrict broadcast domains.

Layer 3 is the intelligent electronic device (IED) zone where critical assets are located. It is here that the operator interacts with the human-machine interface (HMI). This layer is where individual devices are configured by operators and time-critical data are received and forwarded to other devices. Ethernet switches with port security are used here. This layer's network can be expressed by multiple types of topologies, such as ring or mesh, depending on the need. If there is a requirement to separate application traffic within the zone, two Ethernet channels can be built within a single-ring (or two-ring) design with a different VLAN for each.

IT NETWORKS VS. OT NETWORKS

Recall that the attack on the German steel mill went from an IT office network to an OT network, and the attackers appeared to possess advanced knowledge of both IT security and ICSs. This is important to understand in order to protect networks and mitigate attacks.

IT is technology that communicates, shares, stores, and processes information. An IT network is used in the typical office or corporate environment for workers to communicate, store files, and print. In terms of security, an IT network is expected to suffer some delays and outages. Antivirus protection is common and widely deployed. Patch management occurs on a regular basis. Security awareness is generally good.

OT is hardware and software that monitors and controls physical devices and processes. An OT network is used in IDSs, where machines communicate with each other to control a process. In terms of security, an OT network must be available around the clock. Antivirus protection is difficult to deploy because of the many manufacturer-specific devices and protocols. Patch management is also manufacturer-specific.

For example, consider a crafted email that is opened on a computer in the administrative department of an industrial plant or electric utility, releasing a remote access Trojan (RAT). The RAT targets control systems. When it finds a control system, it sends out a connection to one of its command and control servers via the Internet. If that computer is able to reach systems in the control center, the worst possible scenario is that the control center is compromised. Antivirus protection alone cannot be trusted to prevent all threats.

Another dangerous scenario is described in the following example. An engineer or technician falls prey to a phishing attack while connected to the corporate network. That person then takes his or her laptop to service equipment at an OT/ICS location. Because the computer is now infected, even if rules were written to protect the bridge between these IT and OT networks, the engineer or technician effectively bypassed that firewall by plugging directly into engineering access ports at the ICS/OT location. If the RAT has code to search for ICS gear, this could be a very dangerous situation. This is why OT networks are not necessarily safe just because they are air gapped from IT networks. It only takes a USB drive or an infected machine plugged into the center where the IED devices live to compromise the OT network. The center tends to have very poor security controls.

A typical IT perimeter firewall must be configured to block certain ports and protocols. The firewall analyzes traffic coming into the network. It is Internet-facing (an untrusted network), and the back end is in a trusted zone. An IT perimeter firewall usually passes all internal traffic from inside the trusted zone to the outside, or untrusted zone. This is default firewall behavior. Unlike an IT firewall, an OT firewall makes no such assumptions and starts with deny-by-default. This means that no traffic is allowed in either direction until it is configured to open certain ports and protocols. Some companies may not use a combination of in/out security rules, so there is no rule to block that connection.

How to bridge the gap between IT and OT networks is a common discussion today. The attack on the German steel mill that migrated networks indicates the need to consider designer threats designed for specific targets.

INTRUSION DETECTION TECHNOLOGY

As mentioned previously, the main goal of an IDS is to detect a breach into a network or device attached to a network. In addition to detecting malicious threats, an IDS is valuable in the detection of policy violations. For example, a security incident occurred in which a utility engineer placed an Ethernet cable with Internet access into a device on a secure network with the intention of using the connection temporarily to update the device. The cable was forgotten and found months later after exposing the system to the outside world. An IDS would have detected this policy violation if the device started communicating outside of the network.

Components of an IDS

A typical IDS is composed of the following components:

- Rugged industrial computer.
- Managed Ethernet switch with port mirroring technology.
- UNIX® OS for the rugged industrial computer.
- Network intrusion detection software (open source).
- Syslog server and a log access policy.

A rugged industrial computer should be considered when deploying intrusion detection technology. In the application described later in this paper, an industrial computer with an anticipated mean time between failures (MTBF) many times that of typical industrial computers was used. The reliability of the computer is due to the lack of moving parts, such as fans and spinning drives. Solid-state drives reduce wear and tear and can be used in a redundant array of independent disks (RAID) configuration. Error-correcting code (ECC) memory protects against bit flips that produce digital logic errors.

Most IDS deployments use a method known as port mirroring and, therefore, a managed Ethernet switch with port mirroring is required. Port mirroring is used on a managed network switch that copies packets the network sees on one port of the switch to another port. The IDS can be connected to the mirrored port, allowing it to see all data on the network. This is discussed in the Deploying the System section of this paper.

A UNIX OS is required for stability. The IDS in the example application uses Ubuntu, which is a Linux OS.

Network intrusion detection software is the key to the system. The IDS software performs real-time traffic analysis and deep packet inspection, and it logs the data coming in and out of the network.

The final component is a syslog server and a log access policy. A syslog server is used to store the log entries. It is bad practice to only store the log entries on the IDS computer. If the IDS is compromised in an attack, the logs could be modified to cover the tracks of the attacker. Therefore, logs stored by the IDS computer should be sent to an independent syslog server.

It is also important to have a log access policy. It is meaningless to have IDS technology if the logs are accessible to those who do not need the information. The policy should contain some language requiring that the logs be periodically reviewed. It is pointless to have logs if no one is reviewing them. Some attacks on corporations in the past occurred on systems that detected the malware; however, the attacks were not prevented because no one took action on the alerts. Some organizations may already have a Security Information and Event Manager (SIEM) system or log analysis tool available. A SIEM system gathers and organizes logs that were generated, and it can apply strong need-to-know permissions, alerting mechanisms, and sophisticated correlation rules. A SIEM system should be considered when deploying an IDS.

Technical Overview

The core of an IDS consists of three pieces: the server running the IDS software, the rules (which are discussed later in the paper), and the logs that are generated. The example application described in this subsection uses a rugged industrial computer with Ubuntu Linux® as the OS. Snort® was used for the IDS software. Snort is a no-cost open source software licensed under the GNU General Public License. Snort supports SCADA rule sets and performs real-time packet logging and analysis.

The rugged computer acts as the server. The server monitors all packets in and out of the network in a passive manner. It does this by performing a deep packet inspection on all Ethernet frames. Packet inspection is achieved passively through port mirroring technology so that the IDS is not placed in line with the traffic moving through it. It is good practice not to interfere with network traffic, so the passive approach is recommended.

In the passive placement shown in Figure 3, the server with Ubuntu Linux and Snort IDS software is referred to as an intrusion detection sensor. It is connected to the managed switch in a passive mode, meaning the Ethernet frames do not pass through it but rather are sent to both the end device and the sensor. An Ethernet frame egresses the firewall. Traffic on Port 1 is mirrored to Port 7. The Ethernet frame is also sent to the end device on Port 2. Because of the port mirroring, the sensor can see all traffic and perform analysis and deep packet inspection.

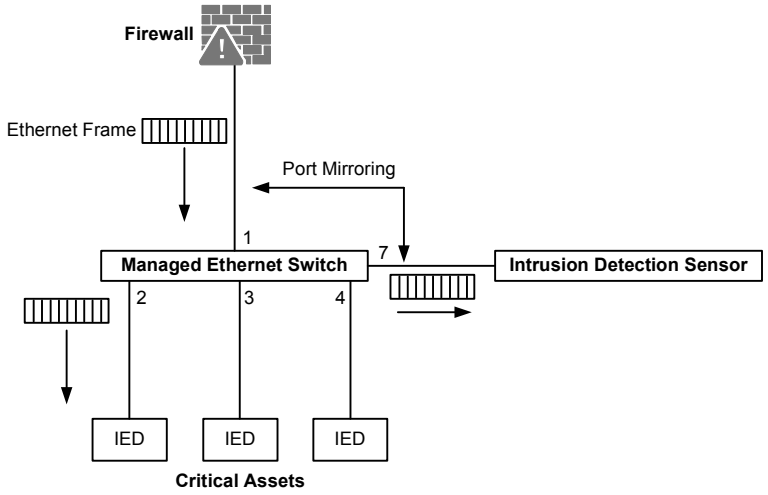

Figure 3. Passive intrusion detection

Rules

A rule-based IDS uses predefined rules to analyze traffic on the SCADA or ICS network. The IDS inspects each packet for information, such as the source and destination, protocol, port, and message content, based on the rule format shown in Figure 4. The rule contains information on how to inspect each packet and how to alert if action is necessary.

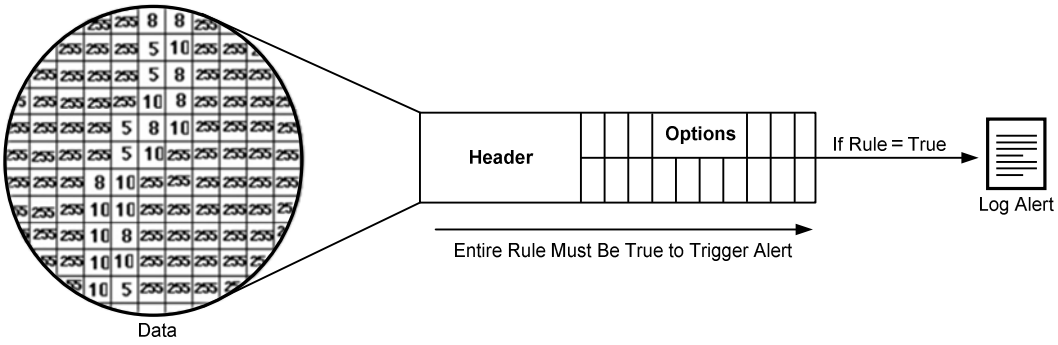

Figure 4. Snort rule format contents

Snort uses user-defined rules to analyze network traffic. The rules are written with a powerful and flexible rule description language. It is beyond the scope of this paper to cover every aspect of writing rules. Rules can be very complex and specific to an application. Therefore, this section discusses the basics of Snort rules and covers the necessary items needed for writing custom rules. It is very important to understand that rules must be in place for Snort to detect intrusions and exploits. Custom rules can be written, rules can be purchased from a third party, or free rule sets can be downloaded from Snort[5]. After installing Snort, the software and rules must be configured.

Snort rules are divided into two sections—the rule header and the rule options. The rule header contains the action the rule is to take if triggered, the protocol to which the rule pertains, the source and destination IP addresses, the netmask information, and the port number. The rule options contain information on how to inspect each packet and how to alert if action is necessary. All conditions in the rule must be true for action to be taken.

A typical rule begins with the action to take if the rule is true and is followed by the protocol of the data, source IP address, and port, as shown in Figure 5.

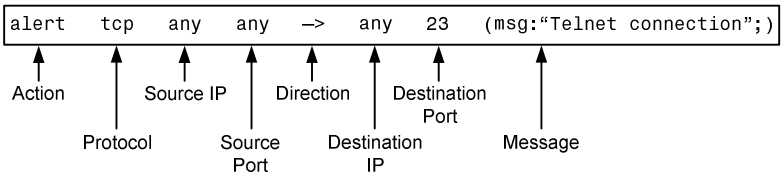

Figure 5. Intrusion detection rule

The example in Figure 5 is a simple rule. It begins with the action to be taken if the rule is true—in this case, the rule generates an alert. Next, the rule examines whether the protocol of the data packet is Transmission Control Protocol (TCP). If so, the rule continues. The rule next examines the source IP address and port—in this case, any IP address and port advances the rule to the next step. The following direction operator indicates the direction of the traffic in which the rule applies. IP addresses and port numbers on the left side of the direction operator constitute source traffic. IP addresses and port numbers on the right side of the direction operator constitute destination traffic. In this example, the rule examines data forming an inbound packet. So far, the rule matches true if an inbound packet is TCP from any IP address.

Next, the rule instructs Snort to analyze the destination IP address and TCP port. In this example, the rule matches any destination IP address. This is common for analyzing a broad range of IP addresses on the network. The rule then checks whether the destination TCP port of the packet is 23, indicating a Telnet connection. If so, the rule has met all requirements. The system logs the packet with the message "Telnet connection," indicating that a Telnet connection has been attempted.

One problem arises with this rule—all legitimate Telnet connections are logged. To address this, all attempts from within the network can be filtered using the $EXTERNAL_NET variable as defined in the configuration file (created as part of the Snort installation) located in the server at /etc/snort.conf. Figure 6 shows this filter applied to the example rule.

```
alert tcp $EXTERNAL_NET any -> any 23 (msg:"Telnet connection";)
```

Figure 6. Filter attempts within network

Some threats may originate from within an organization. These threats are discoverable because an IDS also analyzes traffic between devices. In this example, a rule is used to detect a possible buffer overflow or denial-of-service (DoS) attack. Because the maximum size of a Modbus® TCP packet is 260 bytes, the rule checks for a packet size of greater than 300 bytes. In the event that such a packet is seen, a formatted log event is generated by the rule shown in Figure 7.

```
alert tcp $MODBUS_Client any -> $MODBUS_Server \
502 (dsize:>300; msg:"Illegal Modbus TCP Packet Size";)
```

Figure 7. Buffer overflow rule

DEPLOYING THE SYSTEM

Placement of the IDS is critical to ensuring that the system sees the data intended for evaluation. Deploying the IDS requires working with the network administrator. One of the most common places to deploy an IDS sensor (Ubuntu server with Snort) is inside the firewall at the perimeter of the network. In this configuration, port mirroring is configured for the network switch port to which the sensor is connected. This configuration allows for passive listening. Again, it is not a good idea to use an active sensor in which all data must pass through the sensor.

In Figure 8, it appears that the firewall blocks inbound data from the Internet that are not allowed into the internal network. In its present location, however, the IDS sensor cannot see these blocked threats. Because of this, some data security personnel prefer to deploy a second IDS sensor in front of the firewall to get a sense of the threats being launched at the network.

The sniffing interface is the port that monitors the port-mirroring traffic in and out of the IDS.

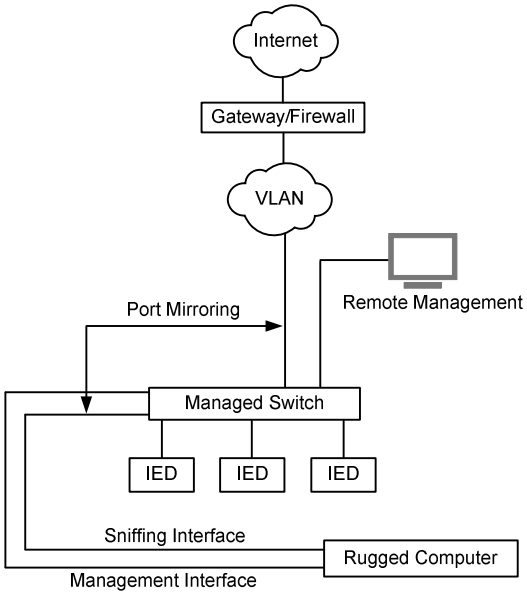

Figure 8. Intrusion detection with port mirroring

The system administrator should be consulted to establish logging to a central logging host and only a subset of the data should be kept at the Snort sensor. One method is to use a syslog server.

TESTING THE CONFIGURATION AND TRIGGERING ALERTS

Testing an IDS is crucial. Triggering some rules verifies the installation and configuration, and it ensures that all rules are working together. It is beyond the scope of this paper to explain how to test every possible set of rules or custom rules. This section looks at a few rules and demonstrates ways to trigger them.

Testing a Telnet Connection

The first example uses the simple rule shown in Figure 9.

```
alert tcp !$HOME_NET any -> any 23 (msg:"Incoming Telnet connection request";)
```

Figure 9. Incoming Telnet connection rule

This rule triggers an alert if incoming data received from outside the local network (the !$HOME_NET term) have a destination port of 23, indicating a Telnet connection.

In this example, a Telnet connection is attempted from a network outside the protected network into a host on the network in which Snort is monitoring. The system triggers an alert similar to Figure 10.

```
[**] Incoming Telnet connection request [**]
07/07-14:10:54.081190 10.100.0.25:3319 —> 10.10.55.187:23
TCP TTL:120 TOS:0xO ID:5736 IpLen:20 DgmLen:48 DF
******S* Seq: 0xC9943A8A Ack: 0xO Win: 0xFAFO TcpLen: 28
TCP Options (4) => MSS: 1460 NOP SackOK
```

Figure 10. Logged Telnet connection request

Testing a Modbus TCP Write

The next rule triggers an alert for any attempt at a Modbus TCP write. The Modbus device for the rule operates on Port 502. This rule looks at the content of the payload—a Modbus TCP write begins with five pairs of zeroes followed by 06. This pattern matches a Modbus TCP write command. If any incoming data match this pattern (regardless of the source) with a destination port of 502, the rule triggers an alert similar to Figure 11.

```
alert tcp any any —> any 502 \
(msg:"Modbus TCP Write"; content:"|00 00 00 00 00 06|" \
sid:40000003; rev:1;)
```

Figure 11. Rule detects Modbus TCP write commands

When a Modbus TCP write is performed, the system triggers an alert similar to Figure 12.

```
[**] Modbus TCP WRITE command issued [**]
12/04-14:23:09.972798 10.100.0.25:4111 -> 10.10.55.187:502
TCP TTL:62 TOS:0xO ID:52432 IpLen:20 DgmLen:62 DF
***AP*** Seq: 0xEB8B38AC Ack: 0x710516FO Win: 0x5AC TcpLen: 32
TCP Options (3) => NOP NOP TS: 3275291157 2553000
```

Figure 12. Logged Modbus TCP write command

Note that there are free tools available to create TCP packets for testing protocols such as the above example of a Modbus TCP write. The Ostinato packet generator created by Google® is an example of one such tool. One common mistake, which could result in a false sense of security, is to deploy an IDS with an error in a rule that results in the rule never triggering an alert. It is important to fully understand the rule to test it properly. Be sure to test all rules to verify that each alert is properly written and that the system is configured.

ADDITIONAL RESOURCES

This paper is an introduction to intrusion detection. Further information is available, including sources such as the white paper "Using SNORT® for Intrusion Detection in MODBUS TCP/IP Communications" by Javier Jiménez Díaz[6]. In addition, Snort rule subscriptions are available for SCADA and ICS applications. SCADA IDS preprocessors are available for Snort[7]. Snort rule subscriptions are also available from Talos (formerly the Snort vulnerability research team)[8]. Cyberthreat intelligence providers, such as Emerging Threats, often provide SCADA-specific rules via a paid subscription.

CONCLUSIONS

It is extremely important to develop a policy regarding those with authority to access an IDS. Deploying an IDS requires careful planning and research. When properly configured and tested, an IDS adds layers of defense by complementing a network with the ability to detect malicious activity or policy violations. In addition, unlike a firewall that is only looking at traffic coming in or out of the network, an IDS looks at traffic within the network.

REFERENCES

1. Federal Office for Information Security, "The State of IT Security in Germany 2014," November 2014. Available: https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Securitysituation/IT-Security-Situation-in-Germany-2014.pdf?__blob=publicationFile.
2. U.S. Department of Homeland Security ISC-CERT Monitor, "Incident Response Activity", July/August 2015. Available: https://ics-cert.us-cert.gov/sites/default/files/Monitors/ICS-CERT_Monitor_Jul-Aug2015.pdf.
3. McAfee, "Combating Advanced Persistent Threats: How to Prevent, Detect, and Remediate APTs," 2011. Available: http://www.mcafee.com/us/resources/white-papers/wp-combat-advanced-persist-threats.pdf.
4. U.S. News and World Report, "Cybersecurity Among Top Energy Industry Concerns," August 2014. Available: http://www.usnews.com/news/articles/2014/08/12/cybersecurity-among-top-energy-industry-concerns.
5. Snort. Available: http://www.snort.org.
6. J. J. Díaz, "Using SNORT® for Intrusion Detection in MODBUS TCP/IP Communications," December 2011. Available: http://www.sans.org/reading-room/whitepapers/detection/snort-intrusion-detection-modbus-tcp-ip-communications-33844.
7. Digital Bond, "Quickdraw SCADA IDS." Available: http://www.digitalbond.com/tools/quickdraw/.
8. Talos. Available: https://www.snort.org/talos.