2016 IEEE 17th International Symposium on High Assurance Systems Engineering

Characterization of Cyberattacks aimed at Integrated Industrial Control and Enterprise Systems: A case study

Raymond C. Borges Hink and Katerina Goseva-Popstojanova
Lane Department of Computer Science and Electrical Engineering
West Virginia University. Morgantown, WV, USA
Emails: rborgesh@mix.wvu.edu; Katerina.Goseva@mail.wvu.edu

Abstract—**Industrial control system (ICS) security has been a topic of research for several years now and the growing interconnectedness with enterprise systems (ES) is exacerbating the existing issues. Research efforts, however, are impeded by the lack of data that integrate both types of systems. This paper presents an empirical analysis of malicious activities aimed at integrated ICS and ES environment using the dataset created and released by the SANS Institute. The contributions of our work include classification of the observed malicious activities according to several criteria, such as the number of steps (i.e., single-step vs. multi-step), targeted technology (i.e., ICS, ES or both), types of cyber-probes and cyberattacks (e.g., port scan, vulnerability scan, information disclosure, code injection, and SQL injection), and protocols used. In addition, we quantified the severity of the attacks' impact on systems. The main empirical findings include: (1) More sophisticated multi-step attacks which leveraged multiple vulnerabilities had higher success rate and led to more severe consequences than single-step attacks; (2) Most malicious cyber activities targeted the embedded servers running on ICS devices rather than the ICS protocols. Specifically, cyber activities based only on ICS protocols accounted for a mere 2% of the total malicious traffic. We conclude the paper with a description of a sample of cybersecurity controls that could have prevented or weakened most of the observed attacks.**

Keywords—Industrial control system security; Enterprise system security; SCADA testbed; Attack characterization; Severity.

I. Introduction

Critical infrastructure such as the energy grid and water treatment plants, which consist of industrial automation and control systems (IACS or ICS for short), have become the center of attention in information assurance since the Stuxnet worm was used since at least as early as June 2010 [1]. Other malware like Stuxnet are still being developed to attack Supervisory Control and Data Acquisition (SCADA) systems used in industrial networks. Among these are sophisticated cyberattacks known as advanced persistent threats (APTs) [2]. The individuals behind APTs are typically well organized and well-funded and often target high profile targets. Components of the US critical infrastructure that are based on or use ICS are now at a higher risk than ever from these.

Many challenges are faced when securing ICS environments. Due to reliability and availability (i.e., uptime) requirements these systems cannot be patched as frequently as typical enterprise systems. Moreover, ICS resources are widely shared over the Internet [3], and many open source exploits for SCADA devices are publicly available. The attacks may also be based on undisclosed vulnerabilities and zero-day exploits. To successfully defend industrial control systems and networks from cyberattacks, risk mitigation strategies and resilience to attacks are necessary. The current relationship between ICS and ES need to be better understood to help improve these attack mitigation strategies and improve resilience techniques.

To better understand these relationships, cybersecurity researchers and practitioners need high quality datasets, preferably with different malicious activities being identified (i.e., labeled). However, few labeled ICS attack datasets are publicly available to the research community. The only exceptions appear to be the two datasets [4] produced using the Mississippi State University testbeds. These datasets were described and initially studied in [5], [6], and [7]. Note that these two datasets were generated using systems that included only ICS components and networks and did not integrate ES.

In this paper we present an empirical study of malicious activities aimed at an integrated testbed with both ICS and ES, which used multiple network protocols like IPv4 and IPv6 TCP, UDP, and ICMP, as well as industrial protocols such as the Common Industrial Protocol (CIP) and Modbus. Our work is based on the dataset produced by the SANS Institute using their newly constructed kinetic cyber range called CyberCity [8] and provided to the public as a single large pcap (packet capture) file as a part of the 2013 SANS Holiday Challenge [9]. The actual attackers were SANS employees and likely included CyberCity testbed administrators and developers. They emulated real-world attackers and their actions based on their experience as penetration testers. Therefore, one would expect these cyberattack activities to be representative of real-world malicious activities. Various reports were submitted for the 2013 SANS Holiday Challenge answering several specific questions. The winning reports were released to the public on the SANS website [10].

From the publicly available pcap file, using existing and custom developed tools, we extracted and extrapolated the hardware/software configuration and services provided in the SANS testbed. In addition, we used various parts of the winning challenge reports to develop a more accurate representation and complete description of the testbed. This provided basis to explore the following research questions:

1530-2059/16 $31.00 © 2016 IEEE
DOI 10.1109/HASE.2016.49

RQ1: What were the percentages of single-step and multi-step attacks?

RQ2: How often did malicious activities target only ICS or ES, compared to malicious activities that included both ICS and ES?

RQ3: How were malicious activities distributed across different types of cyber-probes and cyberattacks?

RQ4: Were certain protocols more likely to be used for launching the observed malicious activities?

RQ5: What types of attacks had the biggest impact to cause critical failures in an integrated ICS and ES environment?

The main contributions of our work are as follows:

- We classified the observed malicious activities according to several criteria, such as the number of steps (i.e., single-step vs. multi-step activities), targeted systems (i.e., ICS, ES, and both), type (e.g., port scan, vulnerability scan, information disclosure, code injection, SQL injection, etc.), and used protocols.
- We assessed the severity of the consequences of these cyberattacks, that is, their impact on the systems' confidentiality, integrity and availability (CIA).

Note that the winning reports of the 2013 Challenge, which were made publicly available on the SANS website [10], described the cyberattacks in terms of the challenge questions (e.g., the main successful attack that caused the power grid outage). These reports neither exhaustively explored the malicious cyber activities nor characterized the cyberattacks in terms of different criteria (i.e., number of steps, targeted systems, types of attacks, and used protocols). Furthermore, they did not systematically or categorically asses the severity of the impact of each of the individual attacks to the ICS and ES.

The main empirical findings based on the analysis of this case study include:

- Multistep malicious activities were more successful than single-step attacks and led to more severe consequences.
- Every cyberattack was preceded by a probe to host(s) or network.
- Cyber activities based only on ICS protocols accounted for a mere 2% of the total malicious traffic.
- Most malicious cyber activities targeted the embedded servers running on ICS devices rather than the devices directly through ICS protocols.

The rest of the paper is organized as follows. Related work is summarized in section II and the SANS testbed is described in section III. Our main findings are presented in Section IV. The threats to validity are described in section V. A discussion of the main findings and possible preventive measures are given in section VI. Section VII concludes the paper.

II. Related work

The related works belong to two main categories: works based on emulated attacks aimed at ICS testbeds and works based on actual cyberattacks observed on deployed ICS honeypots.

A. *Analysis of malicious activities based on ICS testbeds*

There have been many studies on testbeds for ES security but very few on ICS testbeds. This is due to the difficulty of creating a virtual ICS environment and the need to purchase actual physical hardware devices. In [5] we used several machine learning methods to do multiclass classification of the attacks on a gas pipeline system testbed created at the Mississippi State University (MSU). This testbed was composed of a single programmable logic controller (PLC) and a master unit controlling it. The attacks were performed by graduate students at MSU and labeled in collaboration with the Oak Ridge National Laboratory (ORNL) team [11]. Multiple variants of each of the two main attack types (i.e., command injection and data injection) were created. It should be noted that both attack types were straightforward and did not employ stealthy techniques such as cover noise traffic or double spoofing both devices to trick the operator into believing all is normal. Furthermore, the testbed was limited to a single ICS and did not include an ES.

The second testbed [12], which was created at MSU in collaboration with ORNL, was designed to simulate an energy distribution system. This testbed was composed of four intelligent electronic devices (IEDs). Specifically, these IEDs were smart relays that controlled breakers. They contained phasor measurement units (PMUs) which measured the current and voltage and stored the values in log files. The IEDs used a distance protection scheme to trip the breakers on detected faults, where fault is defined as any abnormal electric current. The breakers could also be tripped manually by operators issuing commands. Logs were collected from the PMUs and sent to a central log collection server. In [7] we considered normal scenarios that were caused accidentally by natural phenomena (e.g., lightning) and led to power system disturbances and cyberattack scenarios that led to power system disturbances. The normal scenarios included short-circuit fault and line maintenance, while the cyberattack scenarios consisted of command injection, data injection, and relay setting change. As in [5], in [7] we used various machine learning algorithms to do multiclass classification. However, considering natural faults in addition to cyberattacks added complexity, making it harder to detect and differentiate between naturally-caused faults and attack-induced faults. Note that the second ICS testbed [12], as the first ICS testbed, did not contain ESs and therefore no attacks to ES existed. Even more, in both prior works [5], [7] it was assumed that the attackers had gained access to the on-site internal network because these ICS testbeds had no internet connection.

B. *Analysis of malicious activities based on ICS honeypots*

A combination of one high-interaction and two production ICS honeypots was used to collect actual malicious sessions [13]. The high-interaction honeypot emulated a water pressure station and consisted of an Apache web server with custom-developed web pages to mimic the exact functions of a PLC system. One of the production honeypots provided a human-machine interface (HMI) to control a non-existent PLC device. The second production honeypot was an actual PLC Nano-10 device which was set up as a hypothetical factory temperature controller. Each honeypot had a static public IP address and they were placed at different locations in the United States. The honeypots ran in duration of 28 days and recorded a total of 39 attacks from 14 different countries. Of these attacks, 12 were unique and were classified as "targeted", while 13 were repeated by the same attackers over a period of several days and were considered to be "targeted" and/or "automated." All of the attacks were preceded by port scans performed by the same IP address or an IP address in the same netblock. The observed cyberattacks were grouped into seven attack types: unauthorized access, file modification, traffic modification, setting modification, information disclosure, malware exploitation, and phishing.

In a follow up study [14] a honeypot architecture consisting of twelve different devices was created and used to collect cyberattacks from March to June 2013. A total of 74 non-automated attacks and 33,466 automated attacks were recorded in this study. In addition to the seven attack types considered in [13], authors considered an attack based on Human Machine Interface (HMI) access. (HMI is the software which allows human operators to monitor and issue commands to systems such as PLCs and IEDs.) The observed cyberattacks were further classified as critical (can cause a device failure) and non-critical (the device can continue to operate). Only non-automated attacks were analyzed in [14].

The research based on ICS honeypots in [13] and [14] provided insight into attacks in the wild against ICS systems, but the data collected by these was never made public.

III. Description of the SANS CyberCity Testbed

The SANS CyberCity testbed is a 1:87 scale miniaturized physical city that features SCADA-controlled electrical power distribution, as well as water, transit, hospital, bank, retail, and residential infrastructures [15]. The CyberCity testbed was built for use in SANS training courses, competitions, and military exercises. It has a mixture of ES and ICS composed of real and virtual devices. In this paper assets are defined to be programmable electronic including hardware or software (e.g., virtual devices or HMI).

The scenario presented by the SANS 2013 Holiday Challenge event included an attackers team and the security admin of the testbed. The actual attackers were SANS employees who emulated real-world attackers. The focus of the 2013 Challenge was on the main successful attack that caused the power grid outage. It should be noted that there were multiple cyberattacks against other infrastructures whose analyses were not required for the competition.

The 2013 Challenge dataset was released as a single pcap file which was aggregated from multiple sensors [9]. This pcap file did not include assets from all infrastructures available in CyberCity, but it did include most of them. The time period from 9th to 25th December 2013 was covered in the pcap file, which contained a total of 170,574 packets, out of which 142,285 packets were a separate video feed from a webcam and 28,289 packets contained the malicious traffic.

A. Testbed Network Description

To determine the network layout we started by examining the end points and conversations using various tools such as Wireshark [16] and TShark [17], and our own custom Python scripts. We developed custom Python scripts using the NetworkX [18] and Graphviz [19] libraries to produce conversation graphs for all the devices in the pcap. Some of these graphs showed additional assets that were either not present or not involved in the network activity. We filtered out the public IP addresses to focus only on the machines involved in the attacks. We corroborated our findings with various network layouts proposed in the winning reports [10].

B. Hardware and Software

We manually inspected the traffic for details on each device. From the pcap file we extracted the html web pages for web interfaces and inspected them in order to identify the model type and version of the device and also used the Passive Real-time Asset Detection System (PRADS), which relies on signature-based techniques. For the passive fingerprinting, we used the following tools: P0fv3 [20], PADSv1.2 [21], PRADSv0.3.1-rc1 [22], PRADS-asset-report [23], and prads2snort [24]. In addition to using passive fingerprinting, we also carried out MAC address to device manufacturer resolution using online public databases [25]. In the case of Virtual Machines (VM) we identified what VM software was used by their MAC address, although it is possible to change the MAC address assigned to their virtual network interface.

C. Roles

To help determine the roles of each host, we examined the exfiltrated documents, and also extracted files transmitted in HTTP and inspected them for login pages to the devices. Many of the ICS devices, in addition to controller functions, ran web interfaces on port 80. Some of these web interfaces provided read and write capability with access credentials.

In the past, information technology (IT) and operational technology (OT) were seen as two distinct domains but recent developments have forced the two areas closer together. To classify each asset as either ICS (under OT) or ES (under IT) we took into account what software services it provided. Note that not each device in the ICS category was running a real-time operating system (RTOS). Some enterprise operating systems like Windows XP were hosting web-based HMI to control PLC devices and also acted as clients used to access email and websites using a web browser like Firefox. These devices, therefore, were classified as both ICS/ES (OT and IT).

Table I. HOST DETAILS

Technology	Infrastructure Role	Hardware	Manufacturer	OS	Services	Protocol	Ports	Host	MAC Addr	Spoofed	Spoofing MAC
ES	Electric Energy Web Server	Virtual	VMware	Linux Ubuntu	Apache 2.2.22	HTTP	80	10.25.22.250	00:0c:29:cb:da:ef	No	n/a
ICS	Electric Street Lamps Web-based HMI	Click	Koyo Electronics	Novell Netware 5.1	WWW	HTTP(S)	80, 443	10.25.22.30	00:d0:7c:04:6e:98	No	attempted 00:0c:29:4e:85:2a
ICS	Energy Grid Control System	CompactLogix	Rockwell Automation	VxWorks	Controller	CIP	445, 44818	10.25.22.20	00:00:bc:d0:34:3e	No	n/a
ICS/ES	Electric Grid PLC Web-based HMI	Virtual	VMware	Windows 7, 8 or Win Server 2008	MS-AD, NetBios, VNC	CIP, SMB, VNC	4444, 445	10.25.22.58	00:0c:29:01:40:92	No	n/a
ES	Mail server	Email server	Cisco	Cisco	Email server	POP3	110	10.16.11.5	e0:2f:6d:35:ab:41	No	n/a
ICS	Main Traffic & Potter	Netduino+	Secret Labs	.NET Micro Framework 4.x	Controller, WWW	Modbus, HTTP (S)	502, 443,80	10.21.22.23	5c:86:4a:00:6c:02	Yes	00:0c:29:f7:f4:9a
ES	Proxy or Router	Router	Cisco	MAC OS X 10.x	WWW	n/a	37180, 80, 8081	10.2.2.2	e0:2f:6d:35:ab:20	No	n/a
ICS	Railroad Traffic Controller	Simatic S7-1200	Siemens	Simatic RTOS STEP 7 Firmware	WWW	HTTP	80	10.25.22.23	00:1c:06:0d:3d:3f	No	n/a
All	Router	Gateway router eth0	Cisco	Cisco	n/a	n/a	80	10.21.22.1	e0:2f:6d:35:ab:41	Yes	00:0c:29:f7:f4:9a
All	Router	Gateway router eth1	Cisco	Cisco	n/a	n/a	Unknown	10.25.22.254	e0:2f:6d:35:ab:41	No	n/a
ICS	Traffic Grid Controller PLC	Virtual	VMware	Linux 3.x	Controller	Modbus	46509, 46500, 80	10.21.22.10	00:0c:29:cf:46:ba	Yes	00:0c:29:f7:f4:9a
ICS	Traffic Grid Web-based HMI	MicroLogix 1100	RS Automation	VxWorks	WWW	HTTP	80	10.25.22.22	00:0f:73:03:82:d1	No	attempted 00:0c:29:4e:85:2a
ICS	Traffic Main & Elm	Netduino+	Secret Labs	.NET Micro Framework 4.x	Controller, WWW	Modbus, HTTP (S)	502, 443,80	10.21.22.24	5c:86:4a:00:69:07	Yes	00:0c:29:f7:f4:9a
ICS	Traffic Vine & Elm	Netduino+	Secret Labs	.NET Micro Framework 4.x	Controller, WWW	Modbus, HTTP (S)	502, 443,80	10.21.22.22	5c:86:4a:00:69:05	Yes	00:0c:29:f7:f4:9a
ICS/ES	Train Management	Virtual	VMware	WinXP SP1+ or 2000 SP3	Client	POP3, TLS, microsoft-ds	80, 110, 443,445 53,2546, 8081	10.25.22.253	00:0c:29:de:4f:d9	No	Spoofs gateway eth0
ES	Unknown	Unknown	Dell	Unknown	Unknown	Unknown	Unknown	10.25.22.2	d4:be:d9:6c:8a:42	No	n/a
ICS	Unknown	Unknown		Unknown	Unknown	Unknown	Unknown	10.25.22.1	00:a0:45:6f:c9:ee	No	n/a
ICS	Unknown	Unknown	Phoenix Contact	Unknown	Unknown	Unknown	Unknown	10.25.22.200	00:a0:45:6c:bc:0e	No	n/a
ICS	Unknown	Unknown	Rockwell Automation	Unknown	Unknown	Unknown	Unknown	10.25.22.21	00:1d:9c:a8:3a:08	No	n/a
ICS	Unknown	Unknown	Phoenix Contact	Unknown	Unknown	Unknown	Unknown	10.25.22.24	00:a0:45:37:43:74	No	n/a
ICS	Unknown	Unknown	Phoenix Contact	Unknown	Unknown	Unknown	Unknown	10.25.22.25	00:a0:45:69:aa:55	No	n/a
ICS	Water Treatment Web-based HMI	Virtual	VMware	Linux Ubuntu 3.4	Apache 2.2.22 WWW	HTTP	80	10.22.11.9	00:50:56:b2:0f:d9	No	n/a

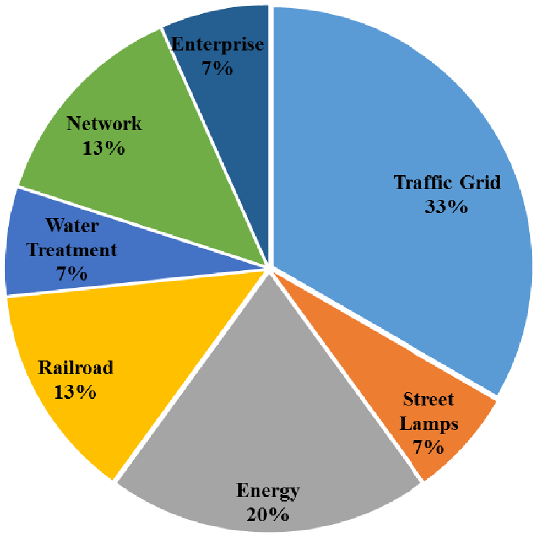

Figure 1. Breakdown of hosts per type of instraructure

D. *Summary of the host details*

Table I lists the host details, including the technology sector they belong to (i.e., ES, ICS, or ICS/ES), the infrastructure role of the hosts, the hardware devices with model and version and manufacturer when identifiable, OS and services running on the host, protocol and ports, host's IP and MAC addresses. The values in the Spoofed column are either "Yes" or "No", indicating whether or not the Layer 2 network address was spoofed by the attacker. The Spoofing MAC indicates the MAC address of the attacker which spoofed this asset if it was indeed spoofed.

IV. DATA ANALYSIS AND RESULTS

Figure 1 provides a breakdown of device distribution per type of infrastructure. It was made by enumerating all of the devices on the network which supported a specific infrastructure. Table II summarizes the main malicious cyber activities, some consisting of a single step, other of multiple steps. Some malicious activities targeted only ES or ICS, while other malicious activities were carried out against both technologies. Some assets were acting in double capacity as servers and also clients used for accessing email, browsing websites, and downloading files from the internet. The last two columns in Table II specify if the attack was stealthy and if there was pivoting involved. (Pivoting is the act of using a compromised machine to gain deeper access to other internal machines.)

TABLE II. MALICIOUS ACTIVITIES

Steps	Infrastructure	Role	Type	Success	Severity	Stealthy	Pivoting
Singlestep	ES	Web Server	Information Disclosure	Yes	2	No	No
Singlestep	ES	Web Server	Information Disclosure	Yes	2	No	No
Singlestep	ICS	Traffic Grid	Port Scan	Yes	1	No	No
Singlestep	ICS	Traffic Grid	Scan	Yes	1	No	No
Singlestep	ICS	Traffic Grid	Modscan	Yes	2	No	No
Singlestep	ICS	Traffic Grid	Command Injection	No	1	No	No
Singlestep	ICS	Traffic Grid	ARP Poisoning	No	0	No	No
Singlestep	ICS	Street Lamps	ARP Poisoning	No	0	No	No
Singlestep	ICS	Street Lamps	Port Scan	Yes	1	No	No
Singlestep	ICS	Traffic Grid	Port Scan	Yes	1	No	No
Singlestep	ICS	Street Lamps	Information Disclosure	Yes	1	No	No
Singlestep	ICS	Traffic Grid	Information Disclosure	Yes	1	No	No
Singlestep	ICS	Traffic Grid	Password Guessing	No	0	No	No
Singlestep	ICS	Street Lamps	Password Guessing	No	0	No	No
Singlestep	ICS/ES	Multiple	Scan	Yes	1	No	No
Singlestep	ICS/ES	Multiple	Scan	Yes	1	No	No
Multistep	ES	Water treatment	SQLi + Code Injection	Yes	4	Yes	Proxy
Multistep	ICS	Traffic Grid	MitM + DoS	Yes	4	No	No
Multistep	ICS/ES	Email+Railroad	Phishing + XSS	No	0	Yes	Proxy
Multistep	ICS/ES	Email+Energy	Phishing + Malware	Yes	4	Yes	No
Multistep	ICS/ES	HMI+ Energy	Password Guess + Malware	Yes	5	Yes	Yes

We examined and organized the malicious activities by the number of steps (RQ1), targeted infrastructure (RQ2), types of cyber-probes and cyberattacks (RQ3), protocols used (RQ4), and the severity of the attack impact (RQ5).

RQ1: What were the percentages of single-step and multi-step attacks?

As can be seen in Table II, the ratio of single-step to multi-step attacks was close to 3:1 (i.e., 76% vs. 24%). Examples of single-step activities included browsing a website and downloading a pdf file or scanning an individual host or subnet. An example of a multi-step activity is sending a malicious phishing email in combination with a malicious file and then waiting for a reverse TCP connection on another host. We considered a multi-step activity something that involved more than one type of activity or more than one host. Thus, a Password Guessing attack is considered a single-step attack because it consists of multiple actions of one activity. A Man-in-the-Middle (MitM) attack, on the other side, consisted of the same type of activity aimed at multiple hosts so it was considered a multi-step attack.

The results showed that both single-step and multi-step malicious activities targeted ES and ICS individually, as well as combination of both (i.e., ICS/ES). From Table II we further observe that multi-step malicious activities were more successful than single-step activities, with a success rate of 80% versus 68%. Multi-step activities were also more severe.

RQ2: How often did malicious activities target only ICS or ES, compared to malicious activities that included both ICS and ES?

As shown in Table II, 14% of malicious activities were aimed only at ES, 62% were aimed at ICS, and 24% targeted both ICS and ES. The most interesting malicious activities were the cyberattacks that used pivoting, which is a technique to use a compromised machine to attack other machines, and then gained control of an HMI to control an ICS device like a PLC. PLC devices can control physical things and may affect the physical world by shutting down power, changing traffic lights, affecting the water quality, or switching train tracks.

TABLE III. PROBES AND ATTACKS

Probes	Frequency		Attacks	Frequency	
Scan	3	7.5%	ARP Poisoning	7	17.5%
Port Scan	3	7.5%	Code Injection	5	12.5%
Vulnerability Scan	1	2.5%	Information Disclosure	5	12.5%
Modbus Scan	1	2.5%	Password Guessing	3	7.5%
			SQL injection	2	5.0%
			Reverse Connection	2	5.0%
			Phishing	2	5.0%
			XSS	1	2.5%
			Malware Trojan	1	2.5%
			Malware Shell	1	2.5%
			Malware Backdoor	1	2.5%
			HMI Control	1	2.5%
			Command Injection	1	2.5%

RQ3: How were malicious activities distributed across different types of cyber-probes and cyberattacks?

The single-step and multi-step malicious activities shown in Table II actually consisted of 40 separate malicious activities, which we grouped in the twenty one single-step and multi-step more complex malicious activities. Each of these 40 activities was composed of one or more packets, most commonly multiple packets forming a session. Table III shows how these 40 malicious activates were grouped into different types of probes and attacks. Overall, 8 out of 40 malicious activities (20%) were probes and the remaining 32 (80%) were attacks.

The malicious activities originated from only two different hosts, except when an attacker was pivoting from a compromised host. In general, each attack was preceded by a probe (i.e., some type of scan) of a host or network segment. Probing activities included various types of scans ranging from basic network exploration via pings to targeted vulnerability scans and specialized Modbus scans.

The attacks included ARP poisoning which was used to perform Man-in-the-Middle (MitM) attacks. Various code injection techniques were used, which involved injecting commands into some part of the accessible web interface. Information Disclosure activities were aimed at gaining access to sensitive information. Password Guessing was performed by brute force and also by reusing passwords previously compromised from other hosts. Phishing emails were sent using fake or spoofed email addresses with a goal to get the user to install malicious software.

Three types of malware were used in the observed attacks. All three malware examples were standard and can be found in the Metasploit penetration testing software. The Trojan malware was created from the base Apache Bench utility using a Metasploit module. The malware shell was the Meterpreter shell used as a payload after a Trojan file connects back to the attacker. Finally, the backdoor malware was a VNC payload which can also be made using Metasploit. We replayed the capture file through both Snort and Suricata Intrusion Detection Systems (IDS) and both detected the malware files.

TABLE IV. PROTOCOL DISTRIBUTION

Protocols	Packets		Size (bytes)	
Ethernet	100.00%	28289	100.00%	19964747
IPv4	90.30%	25544	99.23%	198114.4
Transmission Control Protocol (TCP)	86.79%	24551	98.50%	19665669
TCP.Hypertext Transfer Protocol (HTTP)	2.21%	624	1.65%	329225
TCP.Secure Sockets Layer (SSL)	1.46%	413	2.01%	401299
TCP.Post Office Protocol (POP)	0.11%	30	0.02%	4218
TCP.Modbus	0.35%	100	0.03%	6773
TCP.NetBIOS	1.66%	470	0.59%	118666
TCP.EtherNet/IP.Common Industrial Protocol (CIP)	1.47%	416	0.47%	94022
User Datagram Protocol (UDP)	3.04%	860	0.69%	136903
UDP.Domain Name Service (DNS)	2.36%	667	0.36%	71917
UDP.NetBIOS_Datagram	0.01%	3	0.00%	756
Internet Control Message Protocol (ICMP)	0.47%	133	0.04%	8868
Address Resolution Protocol (ARP)	7.65%	2164	0.52%	2164
IPv6	0.07%	19	0.01%	2857
Link Layer Discovery Protocol (LLDP)	0.56%	159	0.15%	29574

RQ4: Are certain protocols more likely to be used for launching the observed malicious activities?

The distribution of the protocols used by the malicious activities is shown in Table IV Note that the protocol layers can consist of packets that do not contain any higher layer protocol, so the sum of all higher layer packets may not sum up to the protocols packet count. This could be caused by continuation frames and TCP protocol overhead among other things [26].

The results given in Table IV show that even though almost half of the devices in the CyberCity were ICS devices running real-time operating systems (RTOS) and ICS-only protocols (i.e., TCP.Modbus and TCP.EtherNet/IP.Common Industrial Protocol (CIP)), these protocols accounted for less than 2% of the total malicious packets. This was due to the fact that most malicious activities targeted the embedded servers running on these devices. Embedded servers were used by most of the HMI which controlled and monitored the specialized RTOS hardware devices such as PLCs and typically ran on the same ports (i.e., 80 and 443) as enterprise web servers.

RQ5: What types of attacks had the biggest impact to cause critical failures in an integrated ICS and ES environment?

To answer RQ5 we first discuss the security impact of the cybersecurity attributes: confidentiality, integrity and availability (CIA). Thus, when dealing with ICS the importance and criticality of the security attributes typically are different than for ES. While in ES systems the priority is in the stated order (i.e., CIA), for ICS, based on the ISA/IEC 62443 standard, the priority changes to availability, then integrity, and finally confidentiality (i.e., AIC).

We assigned the severity scores based on the observed consequences to the targeted assets' ability to continue to provide confidentiality, integrity and availability during and after the malicious activity occurred. When assigning the severity, we took into account the different priorities for ES and ICS (i.e., CIA vs. AIC). We decided to assign zero severity scores for the unsuccessful attacks because there were practically no discernable overall effects with the exception of one case where repeated attempts caused a slight unintended denial-of-service (DoS). For severity levels

1 and 2 there was some discernable effect, but no critical information or services were compromised. It appeared that attackers usually used many severity level 1 and 2 activities to gain information about the weaknesses of the system and subsequently cause higher level impact. Malicious activities with severity level 3 led to some critical impact, but caused minimal damages to the corresponding system. Severity levels 4 and 5 were reserved for those attacks that led to critical consequences to one or more real physical systems. Level 5 was assigned for cases of full compromise of one or more critical systems.

The assigned severity scores are given in Table II (see the 6th column). As an illustration of the different priority of security attributes for ES and ICS (i.e., CIA vs. AIC) we compare the assigned security scores for Information Disclosure. Thus, Information Disclosure is typically dire to ES, but does not affect ICS assets as much. Therefore, Information Disclosure activity aimed at ES was assigned severity score of 2, while it was assigned severity score of 1 in case of ICS.

As show in Table II single-step attacks had severity scores of 1 and 2, and were less severe than multi-step attacks. The most severe attack was a combination of two multistep attacks – Phishing+Malware attack followed by Password Guessing+Malware attack – which allowed the attackers to control the electric grid and cause a total blackout of the city. Using multistep SQLi+Code Injection attack, the attackers were able to modify the parameters of the water treatment plant and cause chemical imbalances and water purity issues. Another severe attack, which was based on MitM+DoS, led to changing the lights at the intersections incorrectly, and possibly causing accidents.

V. Threats to Validity

As any empirical study, this study has threats to validity, which are described in this section. The first threat to validity is related to the realism of the used testbed. The hardware and software used in the SANS testbed included some exploited in the Stuxnet attack, such as the Siemens Step7 firmware [1]. The MicroLogix 1100 has also been shown to contain vulnerabilities which may result in a DoS attack, a controller fault, a Man-in-the-Middle (MitM) or replay attacks [27]. The Netduino+ controller devices are nowadays being widely used within home devices and are part of the trend towards the Internet of Things (IOTs) [28]. They are also becoming widely used in ICS. These Netduino+ controller devices have also been shown to contain demonstrated vulnerabilities [29].

Another threat is related to the realism of the malicious activities and how representative they are of real attacks. Even though the observed attacks were not carried on by real-world malicious actors, we believe that they are realistic and representative because these malicious activities were done by professional penetration testers. Similar attacks have been observed in several honeypots discussed in the related work section. A potential threat to validity is due to the fact that neither of the attacks was based on zero-day vulnerability nor included complex malware undetectable by standard IDS.

It should be noted that all observed malicious activities were targeted (i.e., search-based), that is, each attack was carefully chosen and targeted specific ports known to be commonly used by ICS and ES devices. The dataset did not include any automated attacks performed by random targeting of IP ranges. We believe that this is not a significant threat to validity due to the fact that in our previous work, which was based on publicly hosted honeypots, we found that there were many more targeted attacks (using search-based strategy) than random attacks (using IP-based strategy) [30].

The final threat to validity, as for any other case study, is related to the external validity of the results. Namely, it is impossible for research based on one case study to claim that its results would be valid for other studies. Different configuration of ICS and ES environment and different type of attacks may lead to different results. This paper presents the first attempt to empirically characterize cyberattacks aimed at integrated ICS and ES. The generalizability of the results remains to be explored in future studies.

VI. Discussion

As shown in Table II, ICS assets were reached both directly and indirectly, via ES assets. The more sophisticated multistep attacks, which successfully exploited multiple vulnerabilities to perform a sequence of actions ultimately led to achieving persistence and sometimes remained unnoticed to go on and perform secondary attacks, pivoting from compromised hosts to other hosts. These attacks led to failures with much more severe consequences than the simpler single-step cyberattacks.

Table V presents some of the possible preventative measures (i.e., security controls) that could have been used to prevent the observed malicious activates. Awareness training would help educating employees about cybersecurity threats and various management, operational, and technical controls available and/or required to protect the ICS and ES resources. System hardening appears to be effective in almost all cases. (System hardening is the process of securely configuring a computer system.) Network segmentation can be used to isolate security breaches, prevent pivoting events and limit the ability of the attacker to compromise additional hosts. Other preventive measures include whitelisting, regularly patching the systems for known vulnerabilities, and running IDS, anti-virus software, and spam filters. All of these promote protection from and detection of malware and may have helped preventing the two most severe multi-step attacks shown in Table II. Note, however, that neither one of these preventive measures is 100% effective.

TABLE V. PREVENTATIVE MEASURES

Malicious activities	Awareness training	System patching	System hardening	Network segmentation	White-listing	IDS	Anti-virus	Spam filters
Scanning			X	X		X		
Pivoting			X	X		X		
Password Guessing			X			X		
Reused password attack	X		X					
Phishing	X				X			X
Malware	X	X	X	X	X		X	X
HMI control			X					
Man in the Middle			X	X		X		
XSS		X	X			X		
SQL/code/command injection		X	X			X		
Web-based attacks		X	X	X		X		

VII. Conclusion

This paper presents an empirical study of the malicious cyber activities aimed at integrated ICS and ES environment. The results are based on using the dataset made publicly available by the SANS Institute. This dataset was produced using their CyberCity testbed.

The main contributions of our work includes classification of the observed malicious activities according to several criteria, such as the number of steps (i.e., single-step vs. multi-step), targeted technology (i.e., ICS, ES or both), types of cyber-probes and cyberattacks (e.g., port scan, vulnerability scan, information disclosure, code injection, SQL injection, and so on), and protocols used. We also assessed the severity of the observed cyberattacks in terms of their consequences to the systems. Furthermore, we presented a set of cybersecurity controls that could have prevented or at least weakened the observed cyberattacks.

Further research focused on integrated ICS and ES environments may shed additional light to the findings presented in this paper, as well as explore the external validity of our results, which provide a solid starting point for such research efforts.

References

[1] D. Kushner, "The Real Story of Stuxnet," http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet/

[2] P. Paganini, "CyberCriminals and their APT and AVT Techniques," http://securityaffairs.co/wordpress/33999/cyber-crime/apt-and-avt-techniques.html

[3] R. C. Bodenheim, "Impact of the Shodan Computer Search Engine on Internet facing Industrial Control System Devices." Master's thesis, Air Force Institute of Technology, 2014.

[4] T. Morris, "Industrial Control System (ICS) Cyber Attack Datasets,"http://www.ece.msstate.edu/wiki/index.php/ICS_Attack_Dataset#Industrial_Control_System_.28ICS.29_Cyber_Attack_Data_Set

[5] J. Beaver, R. Borges Hink, M. Buckner, "An Evaluation of Machine Learning Methods to Detect Malicious SCADA Communications", 13th International Conf. on Machine Learning and Applications, 2013.

[6] T. Morris, R. Vaughn, and Y.S. Dandass, "A Testbed for SCADA Control System Cybersecurity Research and Pedagogy," 7th Ann. Cyber Security and Information Intelligence Research Workshop (CSIIRW 11), ACM, 2011.

[7] R. Borges Hink, J. Beaver, M. A. Buckner, T. Morris, U. Adhikari, S. Pan, "Machine Learning for Power System Disturbance and Cyber-attack Discrimination", 7th International Symposium on Resilient Control Systems (ISRCS), 2014.

[8] E. Skoudis, CyberCity, http://www.sans.org/netwars/cybercity (Accessed: July 2015)

[9] E. Skoudis, 2013 Holiday Challenge, http://pen-testing.sans.org/holiday-challenge/2013

[10] E. Skoudis, J. Wright, and T. Hessman, "Holiday Challenge 2013: Winners and Answers,"

http://pen-testing.sans.org/blog/2014/01/20/holiday-challenge-2013-winners-and-answers (Accessed: July 2015)

[11] Gas pipeline dataset, http://bespin.ece.msstate.edu/wiki/index.php/ICS_Attack_Dataset#Dataset_1:_Gas_Pipeline

[12] Power system datasets, http://bespin.ece.msstate.edu/wiki/index.php/ICS_Attack_Dataset#Power_system_datasets (Accessed: July 2015)

[13] K. Wilhoit, "Who's Really Attacking Your ICS Equipment?" Trend Micro, 2013, http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-whos-really-attacking-your-ics-equipment.pdf

[14] K. Wilhoit, "The SCADA That Didn't Cry Wolf," Trend Micro, Tech. Rep. Part 2, 2013, http://goo.gl/Amw1VG.

[15] "What is NetWars CyberCity?", https://www.sans.org/netwars/cybercity

[16] Wireshark, https://www.wireshark.org/

[17] Tshark, https://www.wireshark.org/docs/man-pages/tshark.html.

[18] NetworkX https://networkx.github.io/documentation/latest/reference/drawing.html

[19] Graphviz https://pypi.python.org/pypi/graphviz

[20] P0fv3 http://lcamtuf.coredump.cx/p0f3/

[21] PADSv1.2 http://sourceforge.net/projects/passive/

[22] PRADSv0.3.1-rc1 http://gamelinux.github.io/PRADS/

[23] prads-asset-report - PRADS Text Reporting Module, http://manpages.ubuntu.com/manpages/precise/man1/prads-asset-report.1.html

[24] prads2snort – Snort autotuning of Frag3 and Stream5, http://manpages.ubuntu.com/manpages/vivid/en/man1/prads2snort.1.html

[25] MAC Address Lookup, http://www.coffer.com/mac_find/

[26] Protocol Hierarchy http://bit.ly/1MkBXC3

[27] Advisory (ICSA-13-011-03) Rockwell Automation ControlLogix PLC Vulnerabilities, https://ics-cert.us-cert.gov/advisories/ICSA-13-011-03

[28] C. Pfister, Getting Started with the Internet of Things, O'Reilly, 2011

[29] SANS Institute, "Critical Control System Vulnerabilities Demonstrated and What to Do About Them," https://www.sans.org/reading-room/whitepapers/analyst/critical-control-system-vulnerabilities-demonstrated-about-35110

[30] K. Goseva-Popstojanova, G. Anastasovski, A. Dimitrijevikj, R. Pantev, and B. Miller, "Characterization and classification of malicious web traffic," Computers and Security, Vol. 42, 2014, pp. 92–115.