Industrial Control System Fingerprinting and Anomaly Detection

Yong Peng, Chong Xiang, Haihui Gao, Dongqing Chen, Wang Ren

Chapter 5

INDUSTRIAL CONTROL SYSTEM FINGERPRINTING AND ANOMALY DETECTION

Yong Peng, Chong Xiang, Haihui Gao, Dongqing Chen and Wang Ren

Abstract Industrial control systems are cyber-physical systems that supervise and control physical processes in critical infrastructures such as electric grids, water and wastewater treatment plants, oil and natural gas pipelines, transportation systems and chemical plants and refineries. Leveraging the stable and persistent control flow communications patterns in industrial control systems, this chapter proposes an innovative control system fingerprinting methodology that analyzes industrial control protocols to capture normal behavior characteristics. The methodology can be used to identify specific physical processes and control system components in industrial facilities and detect abnormal behavior. An experimental testbed that incorporates real systems for the cyber domain and simulated systems for the physical domain is used to validate the methodology. The experimental results demonstrate that the fingerprinting methodology holds promise for detecting anomalies in industrial control systems and cyber-physical systems used in the critical infrastructure.

Keywords: Industrial control systems, fingerprinting, anomaly detection

1. Introduction

Industrial control systems (ICSs), which include supervisory control and data acquisition (SCADA) systems, distributed control systems (DCSs) and programmable logic controllers (PLCs), supervise and control physical processes in critical infrastructure assets such as electric grids, water and wastewater treatment plants, oil and natural gas pipelines, transportation systems and chemical plants and refineries [15, 18]. With the increasing use of commercial-off-the-shelf (COTS) information technology products, TCP/IP-based industrial control protocols and connectivity with other networks, industrial control

systems have become attractive targets for cyber attacks. Malware such as Stuxnet [9], Duqu [3] and Flame [17] have demonstrated the enhanced cyber threats to critical infrastructure assets.

In the information technology field, fingerprinting techniques usually exploit information in TCP/IP protocol headers to automatically identify devices and software; these techniques are used in attacks as well as for protection purposes. Caselli et al. [5] have noted that industrial control system characteristics make device fingerprinting more challenging compared with conventional information technology networks due to device heterogeneity, proprietary protocols, device computational power and long-standing TCP sessions. On the other hand, from the system perspective, industrial control systems – unlike conventional information technology networks – tend to have stable and persistent control flow communications patterns, including characteristics such as long lifecycles, static topologies, periodic behavior and a limited number of applications and protocols [1, 16]. At the same time, every industrial control system is a unique cyber-physical system that is customized to its controlled physical process, control software and hardware.

For these reasons, a methodology is required to discriminate against specific industrial control systems. The fundamental questions are: Can the concept of a fingerprint from the information technology networking field that is used at the component level be translated to the industrial control system field where it is used at the system level? Furthermore, can the system-level fingerprint that represents an industrial control system that is operating normally be used to detect anomalous behavior in the control system?

This chapter attempts to answer these questions. Inspired by device fingerprinting as used in information technology networks, it is argued that industrial control protocol based behavior analysis can derive system-level characteristics of industrial control systems that may be used to discriminate between industrial control systems used in the critical infrastructure. Unlike pure simulation approaches described in the literature, an experimental testbed that incorporates real systems for the cyber domain and simulated systems for the physical domain is employed for validation; such an experimental setup is well suited to analyzing the characteristics of industrial control systems. The experimental results demonstrate that the proposed industrial control system fingerprinting methodology can discriminate between normal system behavior and abnormal behavior.

2. Related Work

Unlike conventional information technology systems that are versatile and variable at the system level, industrial control systems are production systems that are somehow more fixed and regular for long periods of time at the system level. This is one of the characteristics that can be leveraged to extract control system fingerprints. A number of researchers (see, e.g., [6, 15, 18]) have noted that industrial control systems (and cyber-physical systems) have long lifecycles, hierarchical and structural architectures, relatively static topologies and

less variability than information technology systems. Barbosa et al. [1] and Pleijsier et al. [16] have demonstrated that control traffic has characteristics such as periodicity, time-series nature, and static and stable topologies (with stable connections).

With regard to fingerprinting information technology systems, Caselli et al. [5] observe that the most widely adopted fingerprinting technique uses a 67-bit signature from TCP/IP protocol headers to identify an operating system on a machine in a standard network. Caselli and colleagues also describe the challenges involved in fingerprinting industrial control devices. Crotti et al. [8] have proposed the concept of a protocol fingerprint and have demonstrated its utility in discriminating between different network protocols. Their protocol fingerprint is based on three simple properties of IP packets: (i) size; (ii) inter-arrival time; and (iii) arrival order. This research has been inspired by their work, but there is a substantial difference. Crotti and colleagues use IP packet features to derive two statistical vectors that correspond to the protocol fingerprint. On the other hand, this research uses industrial control protocol packet features to derive sets of interactive patterns that represent normal industrial control system behavior and use them to identify anomalous behavior.

Garitano et al. [10] have proposed a method for generating realistic industrial control network traffic. This research is inspired by their observation that industrial control systems can be uniquely discriminated by their communications patterns that embody protocol behavior features. However, the research described in this chapter has different goals and employs a different methodology.

Intrusion and anomaly detection in industrial control systems is an emerging area of research. Cheung et al. [7], Goldenberg et al. [11] and Morris et al. [14] have developed intrusion detection systems for industrial control networks that use the Modbus protocol. Barbosa et al. [2] have used flow whitelists to describe legitimate traffic based on the properties of network packets. However, the research described in this chapter differs from these and other efforts in that it focuses on system-level characteristics. In fact, a search of the literature reveals a lack of research on system-level fingerprinting of industrial control systems and its use in discriminating between normal and abnormal system behavior.

3. Background

A reference model provides a common framework and terminology for describing and understanding industrial control systems. The ANSI/ISA-99 [13] and IEC 62443 [12] standards provide a five-level reference model: (i) level 4 is the enterprise system; (ii) level 3 is for operations management; (iii) level 2 is for supervisory control; (iv) level 1 is for local or basic control; and (v) level 0 is the process. Industrial control systems involve levels 3 through 0. As shown in Figure 1, the reference model used in this chapter is simplified as the process network, control network and physical process.

The cyber domain of an industrial control system includes the process network and the control network; the physical domain is the controlled physical

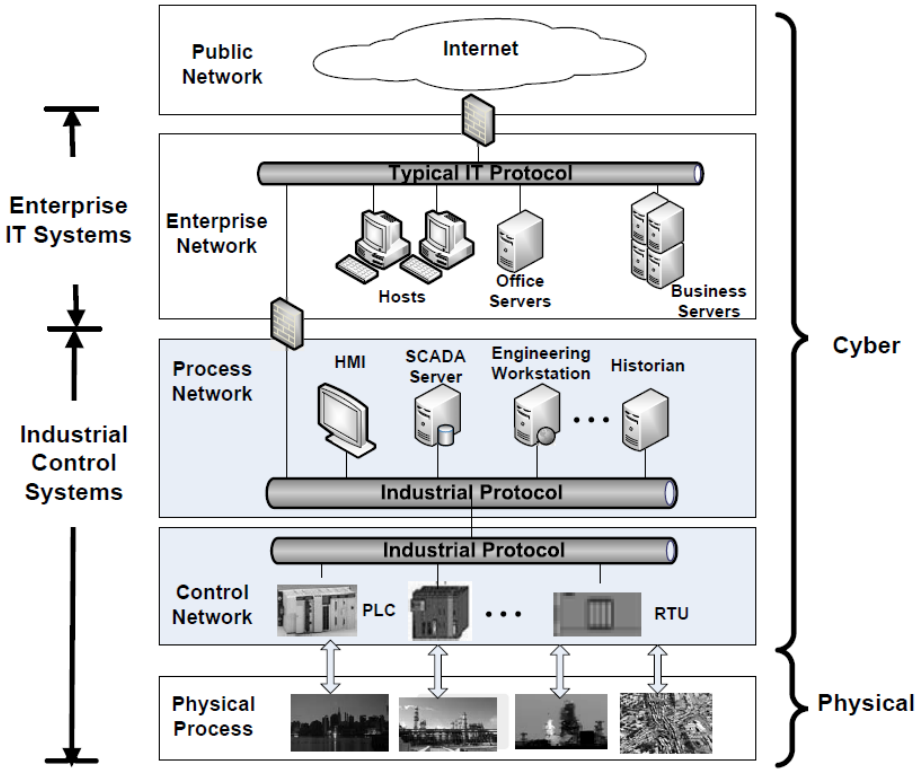


Figure 1. Typical industrial control system architecture.

process (Figure 1). The process network usually hosts human-machine interfaces (HMIs), SCADA servers, engineering workstations and historians. The human-machine interfaces are used by human operators to supervise and control the physical process.

The control network hosts devices such as programmable logic controllers (PLCs) and remote terminal units (RTUs) that, on one side, interact with the physical domain (i.e., the controlled physical process such as a chemical plant) and, on the other side, provide control interfaces to the process network and eventually to human operators. Process network components communicate with control network components using industrial control protocols such as Modbus and DNP3.

4. Experimental Setup

The availability of experimental environments and real-world data pose major barriers to industrial control system security research. Some researchers have used industrial control system traffic traces captured from real installations. However, such traffic contains a lot of noise and is too complex for

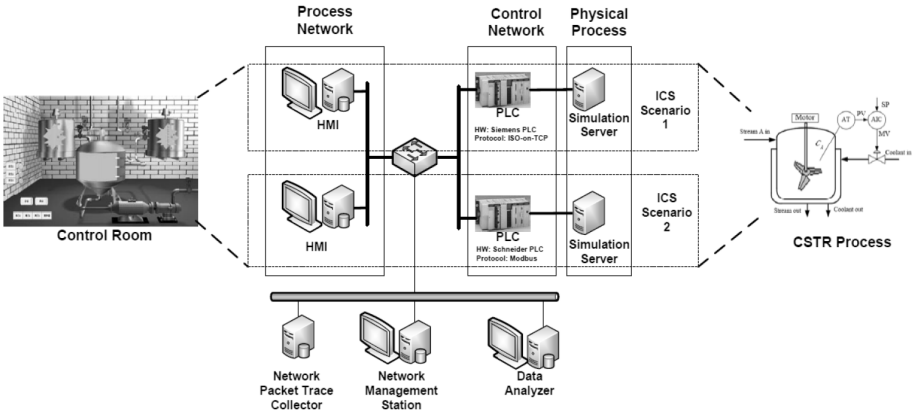


Figure 2. Experimental testbed.

the preliminary research described in this chapter. Other researchers use pure simulations to acquire traffic data, but this data is often inaccurate and the results may be of limited utility. This research focuses on industrial control system traffic in the cyber domain, more specifically, network traffic between a human-machine interface and programmable logic controller. The experimental testbed used in the research engages real hardware and software for the cyber domain and a simulation of the physical domain. This approach yields real control traffic that provides the ground truth of the normal behavior of the industrial control system without any interference or noise. Figure 2 shows the experimental testbed that offers the possibility of acquiring realistic and effective results.

The testbed adheres to the reference architecture presented in Figure 1. It is a part of the larger Cyber-Physical-System-Based Critical Infrastructure Integrated Experimental Platform (C^2I^2EP). The testbed incorporates: (i) an industrial control system that controls a continuous stirred tank reactor (CSTR); and (ii) an experiment analysis system. The process network contains an Intouch human-machine interface. The control network incorporates a Siemens programmable logic controller that communicates with the human-machine interface via the ISO-over-TCP protocol. The physical process is a continuous stirred tank reactor that is simulated in Matlab. The experiment analysis system is used to capture network traffic, perform operations management and analyze data.

Figure 3 shows the continuous stirred tank reactor model used in the research. The model corresponds to a two-state jacketed continuous stirred tank reactor with an exothermic irreversible first-order reaction: $A \rightarrow B$. The process is modeled by two nonlinear ordinary differential equations obtained from the material and energy balances under the assumptions of constant volume, perfect mixing and constant physical properties.

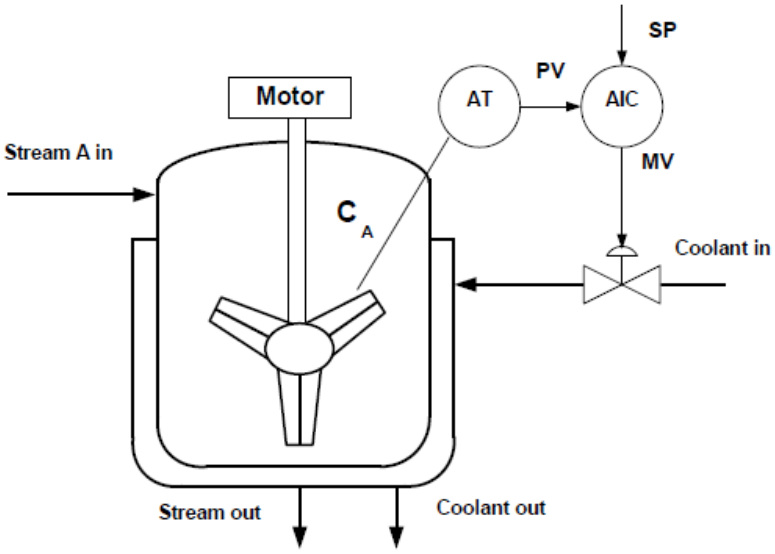


Figure 3. Continuous stirred tank reactor model.

5. Fingerprinting Methodology

The core of an industrial control system is its physical domain, which comprises the controlled physical process. The cyber domain of the industrial control system is used to interactively and/or automatically control the physical process. From the viewpoint of control system designers, every unique industrial control system is a combination of control logic and parameter values [10]. The designers need to specify the control logic and download it to programmable logic controllers and design human-machine interfaces so that human operators can interact with the control system and, thus, the physical system. Specifically, the human-machine interfaces and programmable logic controllers interact via sensor variable values and control variable values using an industrial control protocol. Therefore, by analyzing the interactive behavior characteristics of an industrial control protocol, it is possible to obtain a fingerprint that represents the designers' understanding of the mission requirements of the controlled physical process and the characteristics of the industrial control system components, as long as there are no changes to the physical process and the industrial control system components.

In the context of this research, an industrial control system fingerprint is a set of transaction patterns between a human-machine interface and programmable logic controller. A transaction pattern is, itself, a set of interactive industrial protocol packets that are characterized by properties such as packet arrival order, packet size, direction (from the human-machine interface to the programmable logic controller or from the programmable logic controller to the human-machine interface) and inter-arrival time.

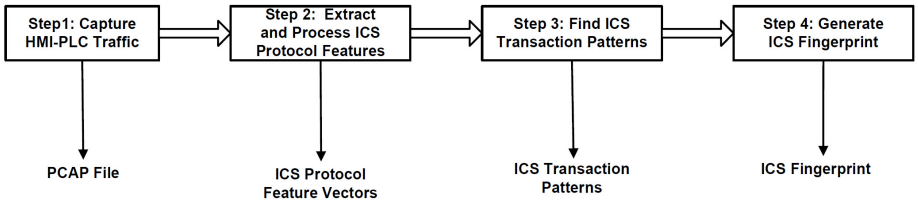
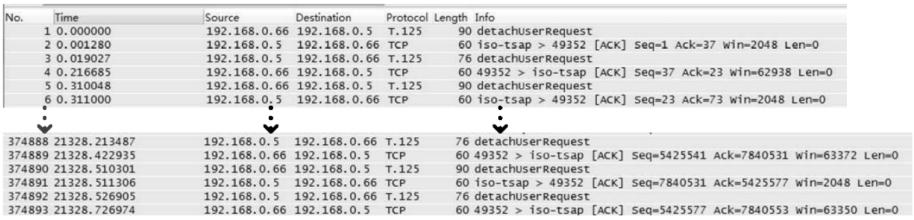


Figure 4. Fingerprinting methodology.

Figure 4 presents the methodology for acquiring an industrial control system fingerprint.



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.66	192.168.0.5	T.125	90	detachUserRequest
2	0.001280	192.168.0.5	192.168.0.66	TCP	60	iso-tsap > 49352 [ACK] Seq=1 Ack=37 Win=2048 Len=0
3	0.019027	192.168.0.5	192.168.0.66	T.125	76	detachUserRequest
4	0.216685	192.168.0.66	192.168.0.5	TCP	60	49352 > iso-tsap [ACK] Seq=37 Ack=23 Win=62938 Len=0
5	0.310048	192.168.0.66	192.168.0.5	T.125	90	detachUserRequest
6	0.311000	192.168.0.5	192.168.0.66	TCP	60	iso-tsap > 49352 [ACK] Seq=23 Ack=73 Win=2048 Len=0
374888	21328.213487	192.168.0.5	192.168.0.66	T.125	76	detachUserRequest
374889	21328.422935	192.168.0.66	192.168.0.5	TCP	60	49352 > iso-tsap [ACK] Seq=5425541 Ack=7840531 Win=63372 Len=0
374890	21328.510301	192.168.0.66	192.168.0.5	T.125	90	detachUserRequest
374891	21328.511306	192.168.0.5	192.168.0.66	TCP	60	iso-tsap > 49352 [ACK] Seq=7840531 Ack=5425577 Win=2048 Len=0
374892	21328.526905	192.168.0.5	192.168.0.66	T.125	76	detachUserRequest
374893	21328.726974	192.168.0.66	192.168.0.5	TCP	60	49352 > iso-tsap [ACK] Seq=5425577 Ack=7840553 Win=63350 Len=0

Figure 5. PCAP file of network traffic.

The fingerprinting methodology incorporates four steps:

- **Step 1:** The first step is to capture traffic traces between the human-machine interface and programmable logic controller. Network traffic capture software such as Wireshark can be used to passively capture the traffic. Figure 5 shows the captured PCAP file for the experimental testbed.

- **Step 2:** The second step is to extract and process the industrial control protocol features. A custom data analyzer or tool such as Scapy [4] may be used to extract packet properties such as packet arrival order, packet size, direction and inter-arrival time. Next, the data is filtered and processed to obtain a set of industrial control protocol packet feature vectors P_i:

$$P_i = (s_i, \Delta t_i, d_i) \tag{1}$$

  where i is the sequence number of a packet exchanged between the human-machine interface and programmable logic controller, s_i is the size of the packet, Δt_i is the packet inter-arrival time between packet_{i-1} and packet_i, and d_i is the direction of the packet flow (d_i has a value of +1 for HMI→PLC and –1 for PLC→HMI). Note that Δt_i is a discretized value that is obtained using a discretization algorithm.

  In the experiment, the continuous stirred tank reactor simulation was run for six hours and the network traffic between the human-machine

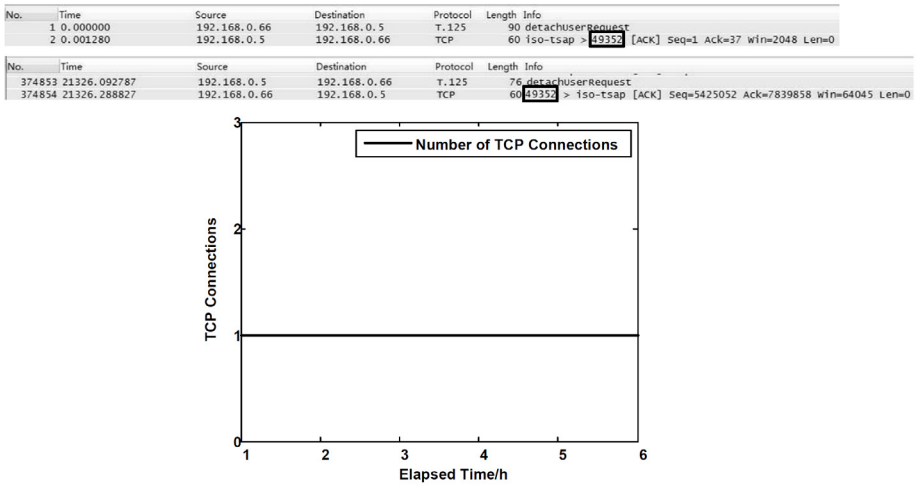


Figure 6. Long-standing TCP connection between the HMI and PLC.

interface and the programmable logic controller was collected. The following interactive industrial control protocol packet characteristics were discerned from the collected data:

1. The industrial control protocol has a long-standing TCP connection that spans several hours (Figure 6). This observation matches that of Caselli et al. [5] and shows that TCP characteristics are not well suited to industrial control system fingerprinting.
2. Each protocol packet has a limited size s_i and a limited number of vectors (s_i, d_i). In the experiment, six types of vectors (s_i, d_i) were distinguished from among the millions of packets that were captured: (i) (60, +1); (ii) (90, +1); (iii) (133, +1); (iv) (60, –1); (v) (76, –1); and (vi) (227, –1). An analysis of the timescales revealed that almost fixed numbers of (s_i, d_i) vectors were observed each hour (Table 1).
3. The packet inter-arrival times can help discriminate between interactive sessions or transaction patterns between the human-machine interface and programmable logic controller. Figure 7 shows that Δt_i has three orders of magnitude: 100 ms, 10 ms and 1 ms.

- **Step 3:** The third step is to find the transaction patterns. The observations in Step 2 imply that certain transaction patterns exist between the human-machine interface and programmable logic controller. Each transaction pattern M_j is a set of bi-directional packet feature vectors:

$$M_j = \{P_1, P_2, \ldots, P_m\} \tag{2}$$

where m is the number of feature vectors.

Table 1. Numbers of vectors (s_i, d_i) at different timescales.

Arrrival Order	Packet Size	Direction	Packet Vector	Inter-Arrival Time (ms)
1	90	+1	90	0
2	60	–1	–60	1
3	76	–1	–76	20
4	60	+1	60	200
5	133	+1	133	100
6	60	–1	–60	1
7	227	–1	–227	30
8	133	+1	–133	1
9	60	–1	–60	1
10	90	+1	90	10

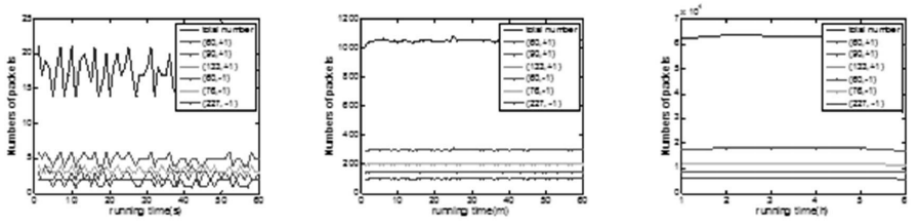


Figure 7. Orders of magnitude of interactive packet inter-arrival times.

Algorithm 1 specifies the procedure for obtaining the industrial control system transaction patterns.

Analysis of the data from the continuous stirred tank reactor simulation revealed exactly eight types of transaction patterns. Figure 8 shows the transaction patterns.

- **Step 4:** The fourth and final step is to obtain the industrial control system fingerprint. The fingerprint Φ^S is given by:

$$\Phi^S = (M_1, M_2, \ldots, M_n) \tag{3}$$

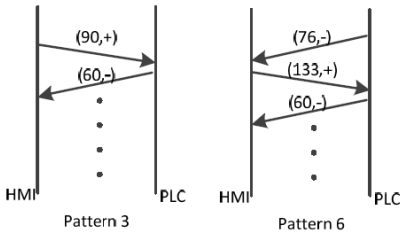
where n is the number of transaction patterns.

In the case of the continuous stirred tank reactor, it is adequate to use the set of transaction patterns as the industrial control system fingerprint. The processing of transaction patterns to obtain a more compact and more accurate fingerprint is a topic for future research.

Algorithm 1: Obtaining industrial control system transaction patterns.

Input: $P_i = (s_i, \Delta t_i, d_i)$, i = 1, 2, ... I
Output: $\Phi^S = \{M_1, M_2, \ldots M_n\}$
% After the analysis, each interaction is observed to end with a packet
% of length 60 without any data
function Patterns($P_i = (s_i,\Delta t_i,d_i)$)
  i = 1
  $\Phi^S = \phi$
  while(i < I)
    k = i
    while($s_k \neq 60$)
      k = k + 1
    end while
      if $\{(s_i, d_i), \ldots, (s_k, d_k)\} \in \Phi^S$
        then $M_j = \{(s_i, d_i), \ldots, (s_k, d_k)\}$
        $\Phi^S = \Phi^S \cup M_j$
      end if
    i = k + 1
  end while
return Φ^S

No.	Patterns			Amounts
1	-227	60	-	5021
2	-76	60	-	47703
3	90	-60	-	45269
4	133	-60	-	7475
5	-76	90	-60	9464
6	-76	133	-60	13910
7	-227	90	-60	16366
8	-227	133	-60	14909



154	8.754135	192.168.0.5	192.168.0.66		76	detachUserRequest
155	8.895291	192.168.0.66	192.168.0.5	6	133	detachUserRequest
156	8.896454	192.168.0.5	192.168.0.66		60	iso-tsap > 49352 [ACK] Seq=3083 Ack=2266 Win=2048 Len=0
157	8.922717	192.168.0.5	192.168.0.66		227	detachUserRequest
158	8.923159	192.168.0.66	192.168.0.5	8	133	detachUserRequest
159	8.924499	192.168.0.5	192.168.0.66		60	iso-tsap > 49352 [ACK] Seq=3256 Ack=2345 Win=2048 Len=0
160	8.950919	192.168.0.5	192.168.0.66		227	detachUserRequest
161	9.047513	192.168.0.66	192.168.0.5	7	90	detachUserRequest
162	9.048643	192.168.0.5	192.168.0.66		60	iso-tsap > 49352 [ACK] Seq=3429 Ack=2381 Win=2048 Len=0
163	9.065441	192.168.0.5	192.168.0.66		76	detachUserRequest
164	9.235588	192.168.0.66	192.168.0.5	5	90	detachUserRequest
165	9.236733	192.168.0.5	192.168.0.66		60	iso-tsap > 49352 [ACK] Seq=3451 Ack=2417 Win=2048 Len=0

Figure 8. Continuous stirred tank reactor transaction patterns.

No.	Time	Source	Destination	Length	Info
1	0.000000	140.90.36.1	140.90.37.1	60	56219 > iso-tsap [SYN] Seq=0 Win=0 Len=0 MSS=1460
2	0.001262	140.90.37.1	140.90.36.1	60	iso-tsap > 56219 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
3	0.001764	140.90.36.1	140.90.37.1	60	56219 > iso-tsap [ACK] Seq=1 Ack=1 Win=1024 Len=0
4	0.244687	140.90.36.1	140.90.37.1	82	CR TPDU src-ref: 0x4431 dst-ref: 0x0000[Malformed Packet]
5	0.246343	140.90.37.1	140.90.36.1	60	iso-tsap > 56219 [ACK] Seq=1 Ack=29 Win=1024 Len=0
6	0.247109	140.90.37.1	140.90.36.1	82	CC TPDU src-ref: 0x4431 dst-ref: 0x4431[Malformed Packet]
7	0.247854	140.90.36.1	140.90.37.1	60	56219 > iso-tsap [ACK] Seq=29 Ack=29 Win=1024 Len=0

Figure 9. PCAP file for the ISO-on-TCP protocol.

6. Fingerprint-Based Anomaly Detection

The industrial control system fingerprint that is derived from normal system behavior can be used to detect anomalous behavior. To detect an anomaly, it is necessary to repeat Steps 1 through 3 for the industrial control system of interest and obtain the set of transaction patterns for the new traffic. Each transaction pattern corresponding to the new traffic is then compared with the corresponding transaction pattern in the fingerprint; a transaction pattern mismatch indicates anomalous behavior.

Two examples are presented to demonstrate the utility of the fingerprint-based anomaly detection methodology. Note that the two examples involve attack traffic with legitimate protocol messages sent from legitimate sources. These correspond to highly stealthy and dangerous attacks on an industrial control system.

The first example involves ISO-on-TCP traffic from a vendor. Figure 9 shows the PCAP file of the ISO-on-TCP traffic. Although the same protocol is used, the packet characteristics are different from those in the original experiment. In particular, packets of length 82 were not seen under normal operating conditions.

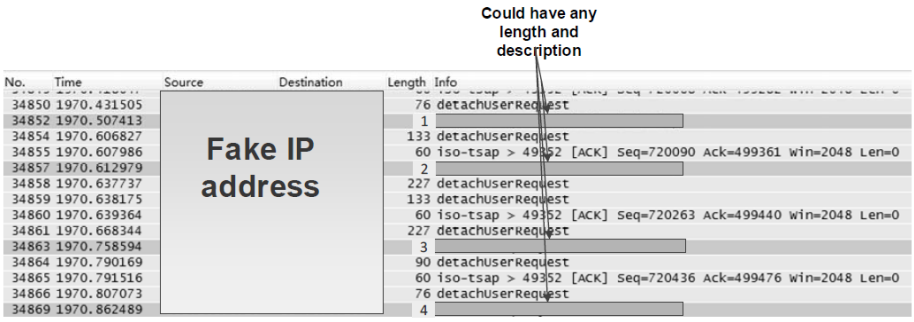


Figure 10. PCAP file for a programmable logic controller scanning attack.

The second example involves a programmable logic controller scanning attack. The assumption is that the attacker is sophisticated enough to use a fake IP address to defeat whitelisting and can adjust the lengths of attack packets. Figure 10 shows the PCAP file corresponding to the attack. Note that any fake length of the second attack packet does not match any combination of the patterns and is, therefore, detected as an anomaly.

7. Conclusions

The industrial control system fingerprinting methodology presented in this chapter leverages the stable and persistent control flow communications patterns in industrial control systems to create fingerprints that correspond to normal behavior of industrial control systems. The fingerprinting methodology is validated using an experimental testbed that incorporates real systems for the cyber domain and simulated systems for the physical domain. The experimental results demonstrate that the fingerprinting methodology holds promise for detecting anomalies in industrial control systems and cyber-physical systems used in the critical infrastructure.

Future research will focus on incorporating real-world industrial control equipment in the Cyber-Physical-System-Based Critical Infrastructure Integrated Experimental Platform (C^2I^2EP) and evaluating complex attack scenarios. Efforts will also be made to acquire and experiment with real traffic from production environments. Additionally, the industrial control system fingerprinting research will attempt to extend the feature set to incorporate lower-level TCP/IP characteristics extracted from real traffic using data mining and statistical analysis techniques.

References

[1] R. Barbosa, R. Sadre and A. Pras, A first look into SCADA network traffic, *Proceedings of the IEEE Network Operations and Management Symposium*, pp. 518–521, 2012.

[2] R. Barbosa, R. Sadre and A. Pras, Flow whitelisting in SCADA networks, *International Journal of Critical Infrastructure Protection*, vol. 6(3-4), pp. 150–158, 2013.

[3] B. Bencsath, G. Pek, L. Buttyan and M. Felegyhazi, Duqu: A Stuxnet-Like Malware Found in the Wild, Laboratory of Cryptography and System Security (CrySyS Lab), Department of Telecommunications, Budapest University of Technology and Economics, Budapest, Hungary (`www.crysys.hu/publications/files/bencsathPBF11duqu.pdf`), 2011.

[4] P. Biondi, Scapy (`www.secdev.org/projects/scapy`), 2014.

[5] M. Caselli, D. Hadziosmanovic, E. Zambon and F. Kargl, On the feasibility of device fingerprinting in industrial control systems, in *Critical Information Infrastructures Security*, E. Luiijf and P. Hartel (Eds.), pp. 155–166, 2013.

[6] M. Cheminod, L. Durante and A. Valenzano, Review of security issues in industrial networks, *IEEE Transactions on Industrial Informatics*, vol. 9(1), pp. 277–293, 2013.

[7] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner and A. Valdes, Using model-based intrusion detection for SCADA networks, *Proceedings of the SCADA Security Scientific Symposium*, 2007.

[8] M. Crotti, M. Dusi, F. Gringoli and L. Salgarelli, Traffic classification through simple statistical fingerprinting, *ACM SIGCOMM Computer Communication Review*, vol. 37(1), pp. 5–16, 2007.

[9] N. Falliere, L. O'Murchu and E. Chien, W32.Stuxnet Dossier, Version 1.4, Symantec, Mountain View, California, 2011.

[10] I. Garitano, C. Siaterlis, B. Genge, R. Uribeetxeberria and U. Zurutuza, A method to construct network traffic models for process control systems, *Proceedings of the Seventeenth IEEE International Conference on Emerging Technologies and Factory Automation*, 2012.

[11] N. Goldenberg and A. Wool, Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems, *International Journal of Critical Infrastructure Protection*, vol. 6(2), pp. 63–75, 2013.

[12] International Electrotechnical Commission, IEC TS 62443-1-1:2009, Industrial Communication Networks – Network and System Security – Part 1-1: Terminology, Concepts and Models, Geneva, Switzerland, 2009.

[13] International Society of Automation, ANSI/ISA-62443-1-1 (99.01.01)-2007, Security for Industrial Automation and Control Systems: Terminology, Concepts and Models, Research Triangle Park, North Carolina, 2007.

[14] T. Morris, R. Vaughn and Y. Dandass, A retrofit network intrusion detection system for Modbus RTU and ASCII industrial control systems, *Proceedings of the Forty-Fifth Hawaii International Conference on System Science*, pp. 2338–2345, 2012.

[15] Y. Peng, C. Jiang, F. Xie, Z. Dai, Q. Xiong and Y. Gao, Industrial control system cybersecurity research, *Journal of Tsinghua University*, vol. 52(10), pp. 1396–1408, 2012.

[16] E. Pleijsier, Towards anomaly detection in SCADA networks using connection patterns, presented at the *Eighteenth Twente Student Conference on Information Technology*, 2013.

[17] sKyWIper Analysis Team, sKyWIper (a.k.a. Flame a.k.a. Flamer): A Complex Malware for Targeted Attacks, v1.05, Technical Report, Laboratory of Cryptography and System Security (CrySyS Lab), Department of Telecommunications, Budapest University of Technology and Economics, Budapest, Hungary (`www.crysys.hu/skywiper/skywiper.pdf`), 2012.

[18] K. Stouffer, J. Falco and K. Scarfone, Guide to Industrial Control Systems (ICS) Security, NIST Special Publication 800-82, National Institute of Standards and Technology, Gaithersburg, Maryland, 2011.