POSTER: Vendor-Independent Monitoring on Programmable Logic Controller Status for ICS Security Log Management

Jongwon Choi*
The affiliated institute of ETRI
Daejeon, South Korea
jwchoi5790@nsr.re.kr

HyungKwan Kim
The affiliated institute of ETRI
Daejeon, South Korea
james@nsr.re.kr

Seungoh Choi
The affiliated institute of ETRI
Daejeon, South Korea
sochoi@nsr.re.kr

Jeong-Han Yun
The affiliated institute of ETRI
Daejeon, South Korea
dolgam@nsr.re.kr

Byung-Gil Min
The affiliated institute of ETRI
Daejeon, South Korea
bgmin@nsr.re.kr

HyoungChun Kim
The affiliated institute of ETRI
Daejeon, South Korea
khche@nsr.re.kr

ABSTRACT

We develop a method for collecting security logs of industrial control systems (ICS) as a preliminary study for ICS security log management and integrated monitoring systems. Although there is scope to collect security logs by using traditional IT technology, this is challenging for special ICS devices such as a programmable logic controller (PLC). PLCs are the major target of APT in ICS because physical damage can be caused by connecting directly with sensors or actuators. According to NIST SP 800-92 [4], that provides importance of log management in computer security, security logs generated from PLCs also need to be managed to enhance ICS security. Therefore, this study analyzes how to collect various information on PLCs. Additionally, we experimented with collecting system logs from a PLC that provides system information via a web interface, and the results are described.

CCS CONCEPTS

• **Computer systems organization → Embedded and cyber-physical systems**.

KEYWORDS

Cyber-Physical System; Security Log Collection; PLC; DCS;

ACM Reference Format:
Jongwon Choi, HyungKwan Kim, Seungoh Choi, Jeong-Han Yun, Byung-Gil Min, and HyoungChun Kim. 2019. POSTER: Vendor-Independent Monitoring on Programmable Logic Controller Status for ICS Security Log Management. In *ACM Asia Conference on Computer and Communications Security (AsiaCCS '19), July 9–12, 2019, Auckland, New Zealand.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3321705.3331007

1 INTRODUCTION

Advanced persistent threat (APT) has been increasing recently, targeting industrial control system (ICS) [3]. An ICS with an integrated physical environment and virtual environment has a broad attack surface because there are various devices. APT attacks against ICS can cause national disaster and even personal accidents. Therefore, attackers consistently launch cyber threats against ICS. In terms of mitigating cyber threats to ICS targets, single security appliances and technologies such as IDS/IPS and Firewalls are somewhat limited. APT attackers focus on various targets over a long period. A single security appliance or technology can block an attack attempt once, but continuous attack attempts such as APT are difficult to defend with a single security appliance or technology. Therefore, the integrated management of logs occurring in various devices is required.

ICS consists of a hierarchical structure arranged according to the role to be performed, as shown in **Figure 1**. The types of devices located in each layer are different, and thus the network communication protocol will also be also different. Because APTs target the entire ICS area, it is necessary to examine the characteristics of the nodes located at each layer and the collectible logs. If various logs are collected, they can be used as an indicator to grasp APT symptoms as shown in **Figure 2**. By analyzing these logs comprehensively through the macro view, the operator can design a

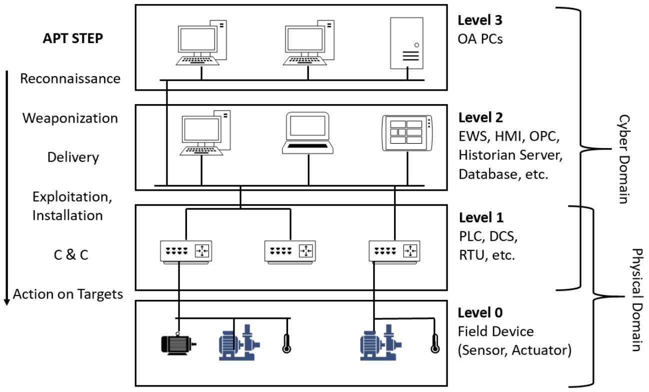
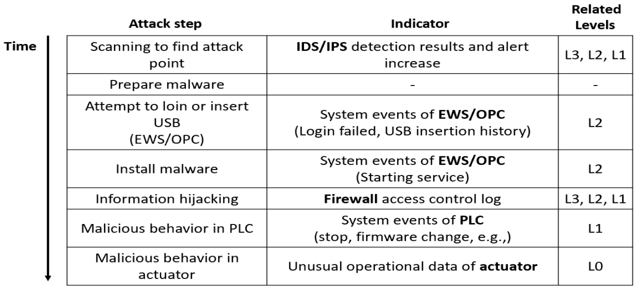


Figure 1: General ICS architecture and APT overview

[*] Corresponding author

AsiaCCS '19, July 9–12, 2019, Auckland, New Zealand
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-6752-3/19/07.
https://doi.org/10.1145/3321705.3331007



Attack step	Indicator	Related Levels
Scanning to find attack point	**IDS/IPS** detection results and alert increase	L3, L2, L1
Prepare malware	-	-
Attempt to loin or insert USB (EWS/OPC)	System events of **EWS/OPC** (Login failed, USB insertion history)	L2
Install malware	System events of **EWS/OPC** (Starting service)	L2
Information hijacking	**Firewall** access control log	L3, L2, L1
Malicious behavior in PLC	System events of **PLC** (stop, firmware change, e.g.,)	L1
Malicious behavior in actuator	Unusual operational data of **actuator**	L0

Figure 2: Need for integrated analysis

strategy to respond to the APT and make decisions to mitigate the APT at the appropriate timing.

The National Institute of Standards and Technology (NIST) under the U.S. Department of Commerce also mentions that it is important to manage the security logs generated in the systems and networks to cope with the malicious activity occurring in each organization [4]. According to the document, security logs include logs generated by security appliances such as IDS/IPS, logs generated by security software such as anti-malware, and vulnerability management. It also contains system logs of system events (shutdown, starting a service, etc.) and audit records (authentication failed, account change, etc.), which occur on various nodes located in the organization.

We are developing a system that collects security logs from Level 0 to Level 3 and detects ICS attacks through correlation analysis. To achieve this, the security logs collection method must be designed for each level.

Level 3 is the operation and control network that confirms the operation status of the control system and uses it for the work. Level 2 includes OPC[1], historian server, and HMI, as well as EWS, which integrates and manages control devices. Existing IT security technologies can collect the security logs of these nodes. An example is the Windows event log and syslog. Additionally, in the Level 3 and Level 2 areas, security appliances such as IDS/IPS and Firewall are deployed. Our research team also deployed three types of IDS and one type of firewall in the area to build a testbed environment, and we conducted a preliminary study to utilize the various security logs in the integrated monitoring system for ICS security log management [1]. For Level 0, we also confirmed that there are related technical and commercial devices, such as Blackbox[2].

Technologies that collect security logs at level 1 are often difficult to apply to legacy devices owing to the technology, which depends on a specific manufacturer's product. In general, a dedicated solution for each PLC vendor is used to collect the security log of their PLCs. Therefore, it is not suitable for a PLC-specific solution because it requires high cost and cannot be used in a continuously connected state during operation. Furthermore, legacy PLCs may need to be replaced. Recently, various PLCs have provided web interface functions. A tendency has emerged to provide information through web interfaces in various PLCs. The Siemens[3] S7 series PLC provides information including system events, CPU usage, and memory usage via web interface. In addition, PLCs from various vendors such as Allen-Bradley[4], Omron[5] and Wago[6] provide web interface functions. By collecting the information provided through the existing web interface, it is possible to solve the problems caused when using the dedicated solution. A software developed to collect logs using this web interface function can also collect security logs from various PLCs regardless of PLC specification and vendor. Therefore, in this study, we develop a web crawling-based software that collects security logs and conduct experiments to lay the foundation for ICS security log management. In the future, this can be used to build an integrated monitoring system for ICS. **Table 1** shows the collected security logs from Siemens PLC corresponding to NIST SP 800-92 [4].

Table 1: Example of security logs on PLC

Categories according to NIST [4]	Siemens PLC security logs [5] corresponding to NIST
System	Name, Version, CPU model, Firmware version Deployed location (rack and slot number) Current Operation mode (RUN, STOP) Operation mode change
Audit	Command execution error, Synchronous error User authentication (Success, Failure, Locked) PLC fault(stop) reason Utilization rate (CPU, Memory)

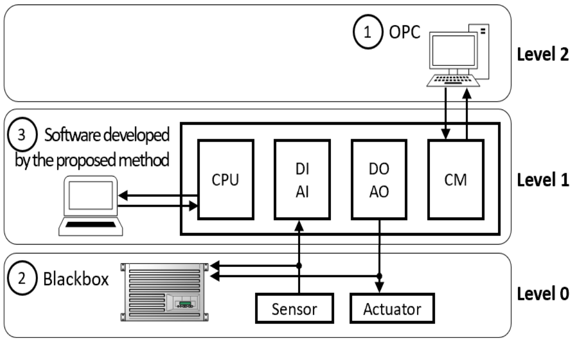


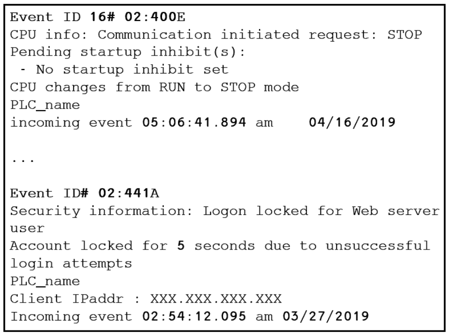
Figure 3: PLC security log collection point

2 ANALYSIS OF PLC SECURITY LOG COLLECTION

Figure 3 shows the collection point of the security logs in a generic PLC. The PLC consists of a CPU module that executes logic and processes operations, an I/O module (DI, AI, DO, AO) that exchanges values with sensors and actuators, and a communication module (CM) that transmits operational data to a higher level. Furthermore, we can collect security logs from each of these modules.

First, it is a method of collecting the security log transferred to the OPC through the CM module of the PLC. The OPC can collect operational data through relatively simple manipulations and without any specific physical changes. However, the method

[1] https://opcfoundation.org/products
[2] https://www.autem.de/products/blackbox_e/
[3] https://new.siemens.com/global/en.html
[4] https://ab.rockwellautomation.com/
[5] https://www.omron.com/
[6] https://www.wago.com/global



```
Event ID 16# 02:400E
CPU info: Communication initiated request: STOP
Pending startup inhibit(s):
 - No startup inhibit set
CPU changes from RUN to STOP mode
PLC_name
incoming event 05:06:41.894 am    04/16/2019

...

Event ID# 02:441A
Security information: Logon locked for Web server
user
Account locked for 5 seconds due to unsuccessful
login attempts
PLC_name
Client IPaddr : XXX.XXX.XXX.XXX
Incoming event 02:54:12.095 am 03/27/2019
```

Figure 4: Example of PLC security logs

involves a considerable cost in comparison with other methods. Furthermore, it does not collect data directly from the PLC; it is an indirect way of reading the data stored in the OPC. In this case, an attack that manipulates information in the PLC and OPC communication sections may occur, and the collected security logs may be unreliable.

Second, data can be collected in the communication between the I/O module and the sensors/actuators. In this case, commercial solutions like Blackbox can still be used, but the required costs would be high. We also used a PLC-in-the-middle method to collect data at the points [2]. Another PLC is placed between the PLC and the sensor/actuator. The PLC then reads the information in the middle and transfers this information to the collection system. However, the placement of the additional PLC may result in cost and latency issues; furthermore, the use of physical wiring should also be attempted.

Finally, it provides a way to collect system logs that are generated on the CPU module itself. The manufacturer of the PLC has recently recognized the importance of log management; thus, the PLC now automatically generates a system log from the CPU itself. In a specific PLC, a web server exists in the CPU module so that the operator can check the PLC status remotely. This is important information for determining whether the PLC is operational or not. Software development is required to collect this information, and the details of this are covered in the next section. It is necessary to collect PLC system logs by analyzing various PLCs, even if it is not a web server method.

3 COLLECTING PLC SECURITY LOG USING WEB INTERFACE

In this section, we explain how the experiment is conducted to collect security logs from PLCs that provides information via the web. In this study, we targeted Siemens S7-1500 and applied the security log collection method through a web interface that can be customized according to requirements. The implemented software can be found at the following link[7].

3.1 Identify Collectable PLC Security Log

Figure 4 shows the result of checking the PLC log by connecting to the web server built in Siemens S7-1500. The information that can be collected by the function comprises the PLC system events and the audit records. This information is useful for indicating the current status of the PLC in real-time. However, from the perspective of ICS integrated security management, it is necessary to centralize and collect the information of all PLCs located in ICS, rather than only monitoring the status of each PLC. The identified security log is information related to the operation mode, integrity failure, and access record of the PLC.

3.2 PLC Security Log Collection Result

Figure 5 illustrates log generation trends for five months through a time series graph. We have confirmed that when a polling method of every 1 second is used in our experiment environment, we collect it without affecting PLC operation. At the time of the cyberattack experiment, various attempts were made to perform malicious activity on the PLC, and it was confirmed that a large number of security logs from the time of the attack remained to be analyzed.

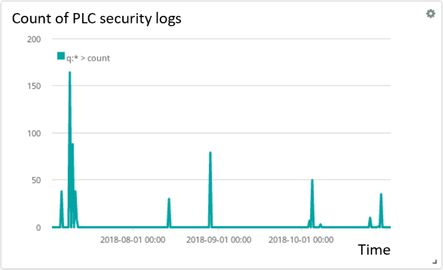


Figure 5: Overall PLC security log trends

4 CONCLUSION

In this study, we identified and analyzed various nodes that can collect security logs for cyber threat response in an ICS environment. In addition, we collected security logs such as system events and audit records without affecting availability from an experimental environment that comprises an actual PLC providing web interface. We could collect security logs for PLCs supporting the web interface regardless of the manufacturer. To use these web interfaces safely, each PLC vendor should strive to improve web interface security. We expect that this will be useful for correlation analysis in future integrated monitoring systems.

REFERENCES

[1] Seungoh Choi, Yesol Kim, Jeong-Han Yun, Byung-Gil Min, and HyoungChun Kim. 2019. Data-Driven Field Mapping of Heterogeneous Security Events for Integrated Monitoring. In *Thirteenth Annual IFIP WG 11.10 International conference on Critical Infrastructure Protection.*

[2] Seungoh Choi, Woomyo Lee, Hyeok-Ki Shin, Jeong-Han Yun, and Sin-Kyu Kim. 2018. POSTER: CPS Security Testbed Development Using Controller-in-the-Middle. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security (ASIACCS '18).* ACM, New York, NY, USA, 829–831. https://doi.org/10.1145/3196494.3201589

[3] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response* 5, 6 (2011), 29.

[4] Karen Kent and Murugiah P Souppaya. 2006. SP 800-92: guide to computer security log management. (2006).

[5] Siemens. 2015. Which information is entered in the diagnostic buffer of the SIMATIC S7 CPU with STEP 7. Retrieved April 15, 2019 from https://support.industry.siemens.com/cs/document/14960968/

[7] https://github.com/Jongwon-Choi/Siemens-S71500-SecurityLog-Collector