Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 597697, 7 pages
http://dx.doi.org/10.1155/2014/597697

Research Article

Whitelists Based Multiple Filtering Techniques in SCADA Sensor Networks

DongHo Kang,[1] ByoungKoo Kim,[1] JungChan Na,[1] and KyoungSon Jhang[2]

[1] *Convergence Security Research Section, Electronics and Telecommunications Research Institute (ETRI), Daejeon 305-700, Republic of Korea*
[2] *Department of Computer Engineering, Chungnam National University, Daejeon 305-764, Republic of Korea*

Correspondence should be addressed to DongHo Kang; dhkang@etri.re.kr

Received 31 January 2014; Accepted 6 May 2014; Published 28 May 2014

Academic Editor: Young-Sik Jeong

Internet of Things (IoT) consists of several tiny devices connected together to form a collaborative computing environment. Recently IoT technologies begin to merge with supervisory control and data acquisition (SCADA) sensor networks to more efficiently gather and analyze real-time data from sensors in industrial environments. But SCADA sensor networks are becoming more and more vulnerable to cyber-attacks due to increased connectivity. To safely adopt IoT technologies in the SCADA environments, it is important to improve the security of SCADA sensor networks. In this paper we propose a multiple filtering technique based on whitelists to detect illegitimate packets. Our proposed system detects the traffic of network and application protocol attacks with a set of whitelists collected from normal traffic.

1. Introduction

In general, a SCADA network is a network required for effective remote monitoring and control of the devices remotely scattered. These networks interlink and operate the SCADA systems and various controllers needed to monitor field devices in real-time. In the past, SCADA networks operated in close environments isolated from external networks and adopted an undisclosed protocol and software in order to monitor and control various field devices internally. But modern SCADA systems have distributed architecture and are connected to the corporate network and to the Internet. Recently IoT technologies begin to merge with SCADA sensor networks to more efficiently gather and analyze real-time data from sensors in industrial environments. In addition, these systems use general-purpose operation systems and industry-standard communication protocols such as Modbus and DNP3 for communication between a SCADA system and field devices such as programmable logic controller (PLC) and remote terminal unit (RTU). The increased connectivity and the use of standard protocols can help to optimize manufacturing and distribution processes. But, they also expose these networks to the myriad security problems of the Internet [1]. Before we describe our approach we first introduce the SCADA architecture and protocol for understanding SCADA systems.

1.1. The SCADA Architecture. SCADA networks come in various forms and layers according to the target and size. SCADA networks are employed in many industrial domains including manufacturing and electricity generation. In the past, they were isolated from other networks and proprietary protocols and software were adopted to monitor and control the various local devices [2]. Hence, security services in these networks were considered to be unlikely. But, due to the adoption of Ethernet and TCP/IP, they have evolved an architecture strongly based on connectivity to improve efficiency and productivity. The SCADA architecture usually consists of three different domains [3]. A typical SCADA architecture is shown in Figure 1.

A control center includes human machine interface (HMI), SCADA servers, and historian systems for process control, the gathering of data in real-time from field devices in order to control sensors and actuators. A field site includes multiple field devices that send commands to actuators and

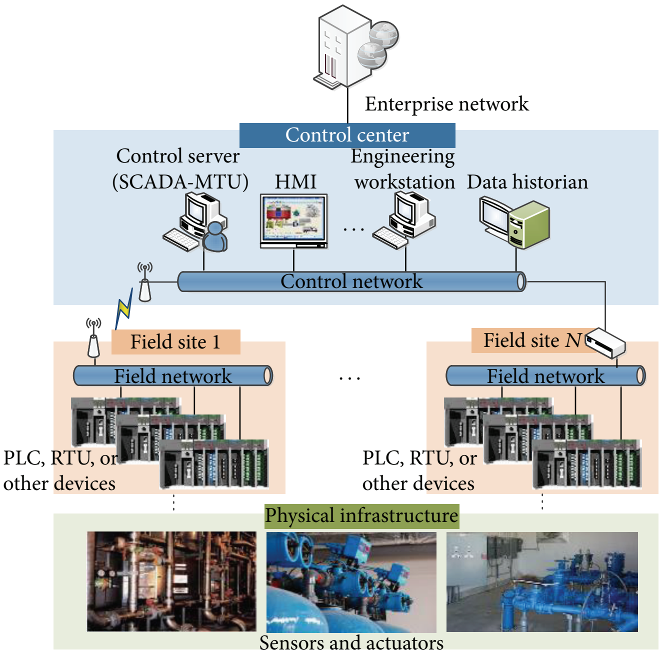

Figure 1: SCADA system general layout.

provide the data received from sensors to SCADA servers. Physical infrastructure consists of many different types of sensors and actuators that are monitored and controlled by a field device.

1.2. Modbus Protocol. Modbus is an application layer messaging protocol which provides master/slave communication between devices in SCADA systems [4]. The function code of Modbus informs the slave of what type of action to perform. For instance, Modbus function "0 × 01" can be used to read the status of an output in the Modbus slave device.

Figure 2 shows the communication between devices connected on the Modbus TCP /IP network. The master that initiates a Modbus transaction builds the Modbus application data unit (ADU). The Modbus ADU consists of the Modbus application protocol (MBAP) header and the protocol data unit (PDU). The PDU has a function code and function parameters. The function codes indicate to the slave which kind of action to perform. The Modbus TCP/IP uses the TCP/IP stack for communication and extends the PDU with an IP header. But there are no security functions in the protocol. The simplicity of the Modbus protocol makes it relatively simple to attack Modbus slaves [5]. If any attackers have broken into the Modbus master, they may send illegal commands to Modbus slaves to perform abnormal behaviors.

The purpose of this paper is to discuss our approach and confirm the validity of our proposed system for preventing network and application protocol attacks in SCADA senor networks. This paper is organized as follows. Section 2 gives detailed cyber threats. Section 3 describes a detailed explanation of our proposed system. Section 4 presents related works and Section 5 gives conclusion.

Table 1: Network protocol attacks.

Attack type	Attacks
Host discovery	OS fingerprinting
Scan	TCP SYN/ACK scan TCP connect( ) scan TCP FIN stealth scan Xmas tree stealth scan TCP null stealth scan Windows scan RPC scan Version detection scan
DoS attack (Denial-of-service)	TCP/UDP flooding Smurf attack

2. Cyber Threats in SCADA Networks

We surveyed vulnerability assessment tools, Metasploit [6], Nessus [7], and Modscan [8] for the classification of cyber-attacks in SCADA networks. These tools are commonly available to find known and newly discovered vulnerabilities on SCADA systems. And we surveyed some reports that were released by the projects of DigitalBond [9, 10]. As a result of our survey, we describe that various types of attacks on SCADA systems can be grouped into two categories: network protocol attacks and application protocol attacks.

2.1. Network Protocol Attacks. Most network protocol based attacks happened in Internet environment may be caused in SCADA networks were adopted IP network. These types of attacks use weak points of network protocols such as TCP/IP suite that have a number of serious security flaws. We introduce some types of network protocol attacks. Table 1

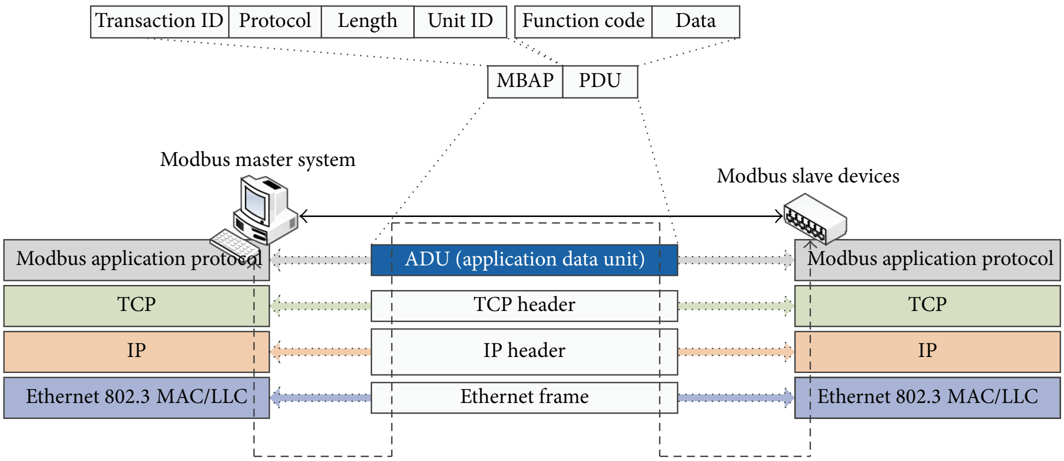

Figure 2: The format of Modbus TCP/IP ADU.

Table 2: Application protocol attacks.

Attack type	Attacks
Application scan	Modbus version scanner PLC Modbus mode identification PLC IO scan status Report slave ID Function code scan
Improper command execution	Force listen only mode Read/write request to a PLC Slave device busy exception code delay Acknowledge exception code delay Broadcast request from a client

shows types of network protocol attacks. Host discovery is the process for gathering information about each host such as its operating system and version to verify whether they can be accessed or not. Using the information gathered about each target host in the host discovery step, attackers launch scan to conform what ports are open, with listening services on target systems. Host discovery and scan attack are the common type of passive attacks to collect the fundamental information of vulnerabilities on target systems. Denial-of-service (DoS) attack is active attack to make systems or network resource unavailable. Network protocol attacks have two characteristics as follows.

(i) Random access: host discovery or scan attacks generally send packets with the sequential or random destination addresses and ports to target networks or systems for obtaining the list of target systems and their services.

(ii) Source address spoofing: DoS attack does not consider receiving responses to the attack packets. Therefore, attackers can send packets with a forged source IP address for obscuring the true source of the attack.

2.2. Application Protocol Attacks. In our work, we surveyed Modbus/TCP as an application protocol. Application protocol attacks can cause damage to field devices, being controlled by sending out improper commands, because they do not support integrity checking and authentication mechanism. Like network protocol attacks, these attacks also preceded by a step of gathering information about devices for finding vulnerable targets in a network. Table 2 shows generally types of application protocol attacks.

Application protocol attacks have the following characteristic.

Unpredictable Command. SCADA systems generally produce predictable sets of command used for communication between a SCADA server and field devices. On the contrary, application protocol attacks tend to use unconventional commands at irregular interval.

3. Our Proposed System: The IndusCAP-Gate System

Our proposed system, the so-called IndusCAP-Gate system, automatically generates whitelists by analyzing the traffic and performs multiple filtering based on whitelists for blocking against unauthorized access from external networks. Figure 3 shows the packet processing flows of the IndusCAP-Gate system.

In the analysis phase the system performs the process of packet decoding and extracts data parameters in the captured traffic for building whitelists. After the analysis phase has been completed, the multiple filters inspect all incoming packets to detect abnormal behavior based on whitelists in the detection phase.

3.1. The Analysis Phase. The analysis phase is an initial training stage for building whitelists. The IndusCAP-Gate system captures and analyzes the traffic on communication between SCADA servers and field devices. The phase is executed for a predefined period and generates whitelists by analyzing normal SCADA traffic. Whitelists are the set of policies to help determine whether incoming packets from

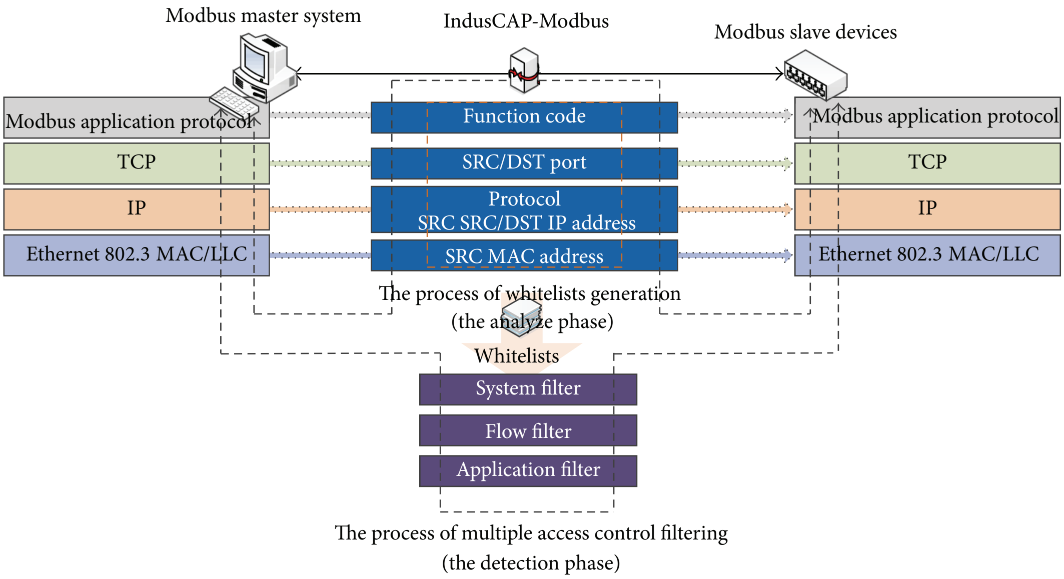

Figure 3: The packet processing flows of the IndusCAP-Gate system.

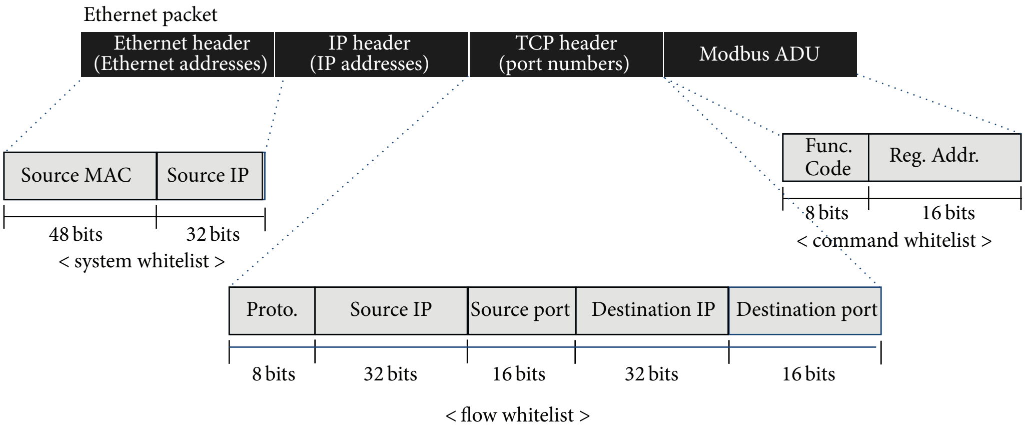

Figure 4: The format of whitelists.

external networks are abnormal. They have three types and Figure 4 shows the format of whitelists.

The system whitelist has multiple source MAC/IP address pairs. We treat these pairs as authenticated systems during the detection phase. The flow whitelist contains a set of the 5-tuple (i.e., the source and destination IP address, the same source and destination port, and the same protocol) information. The whitelists are referred to the flow filter in order to identify the abnormal flows. The command whitelist is used to detect unauthorized Modbus commands by the command filter. Upon completion of the phase, the detection phase uses the result of the analysis phase to identify abnormal traffic. Each whitelist maintained by the system that monitors incoming packets will add entries without a need for human intervention. We assume that the traffic gathered in the analysis phase includes only normal data that does not contain packets generated by the attack. Since SCADA networks, unlike conventional networks, have relatively limited connections to outside networks, attack attempts do not occur frequently and the analysis phase is executed only for a defined short period after the initial installation. We are confident that the assumption will be valid for our approach.

3.2. *The Detection Phase.*

The IndusCAP-Gate system provides whitelists based multiple filters to block unauthorized access to field devices in field networks. The system is positioned between SCADA network and field networks. Figure 5 shows the system architecture.

The IndusCAP-Gate system was designed to protect field devices from various cyber-attacks. For archiving the purpose, the system consists of four functions. The packet collection and control function perform the role of forwarding or blocking packets according to the result of multiple filters. The network layer access control function determines whether to drop or route the packet by inspecting the Ethernet, IP, TCP, and UDP headers. If the incoming packet

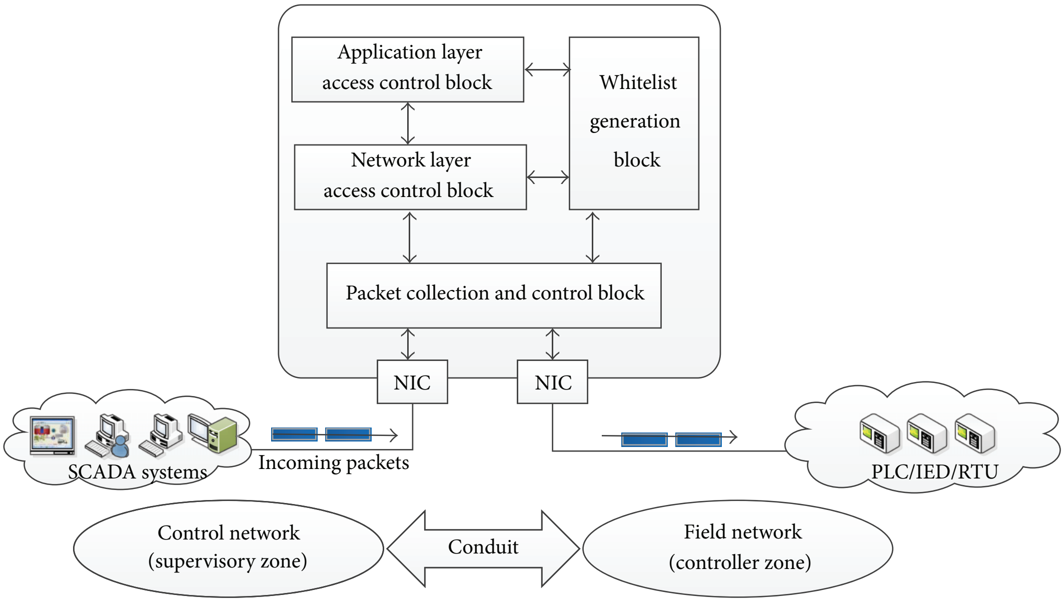
Application layer access control block
Whitelist generation block
Network layer access control block
Packet collection and control block
NIC
NIC
SCADA systems
Incoming packets
PLC/IED/RTU
Control network (supervisory zone)
Conduit
Field network (controller zone)

Figure 5: The system architecture.

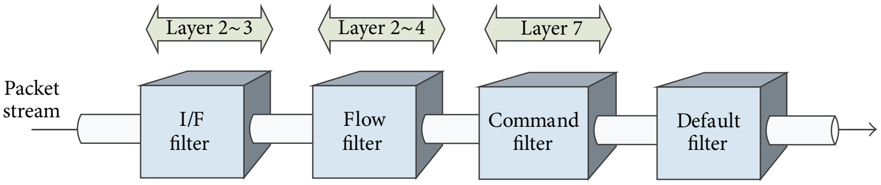

Figure 6: The process of multiple filters.

can meet a condition in system whitelist or the flow whitelist, the function routes it using the system filter or the flow filter. The application layer access control function performs application-level access control at the application layer. This function analyzes the incoming packets using the command filter and then blocks unauthorized access to the command. Using the functions above, the IndusCAP-Gate system blocks unauthorized access from illegitimate traffic.

3.3. Multiple Access Control Filter Based Blocking of Unauthorized Access. As described above, the IndusCAP-Gate system's multiple filters consist of 4 filters.

Figure 6 shows the process of multiple filters. Each filter can be described as follows.

(i) Default filter: a default filter is enabled according to the existence of policy of other access control filters (disabled if there is at least one policy of other access control filters for each interface). It only decides whether the incoming packet will be allowed or denied. Since such enables total access control of incoming packets into a specific interface, it can be useful for special-purpose access control.

(ii) System filter(I/Ffilter): the system whitelist, the policies of MAC/IP pair, is applied for each interface. Only those packets conforming to the applied policies are selected and delivered to the opposite interface.

(iii) Flow filter: the filter performs 5-tuple-based access control with the flow whitelist at the network layer.

(iv) Command filter: the filter performs application-level access control and analyzes the Modbus protocol. It controls access to the command with the command whitelist.

Figure 7 shows the overall packet processing flows of multiple access control filters. As shown in the figure, processing of incoming packets into the interfaces is the same except for those branching into each interface. Only the packets allowed through a filter can be delivered to the next filter. In other words, only those packets allowed through all filters are delivered to the opposite interface. The process allows the IndusCAP-Gate system to block unauthorized access to the control system and apply access control policies efficiently according to the size and nature of the control system intranet.

The IndusCAP-Gate system was implemented to run in Linux OS, adopting the UNO-3072L platform to suit the nature of the SCADA environment. The packet processing performance of the IndusCAP-Gate system was tested using the IXIA traffic generator. Since the SCADA networks generally have low bandwidth, up to 20 Mbps packets transfers

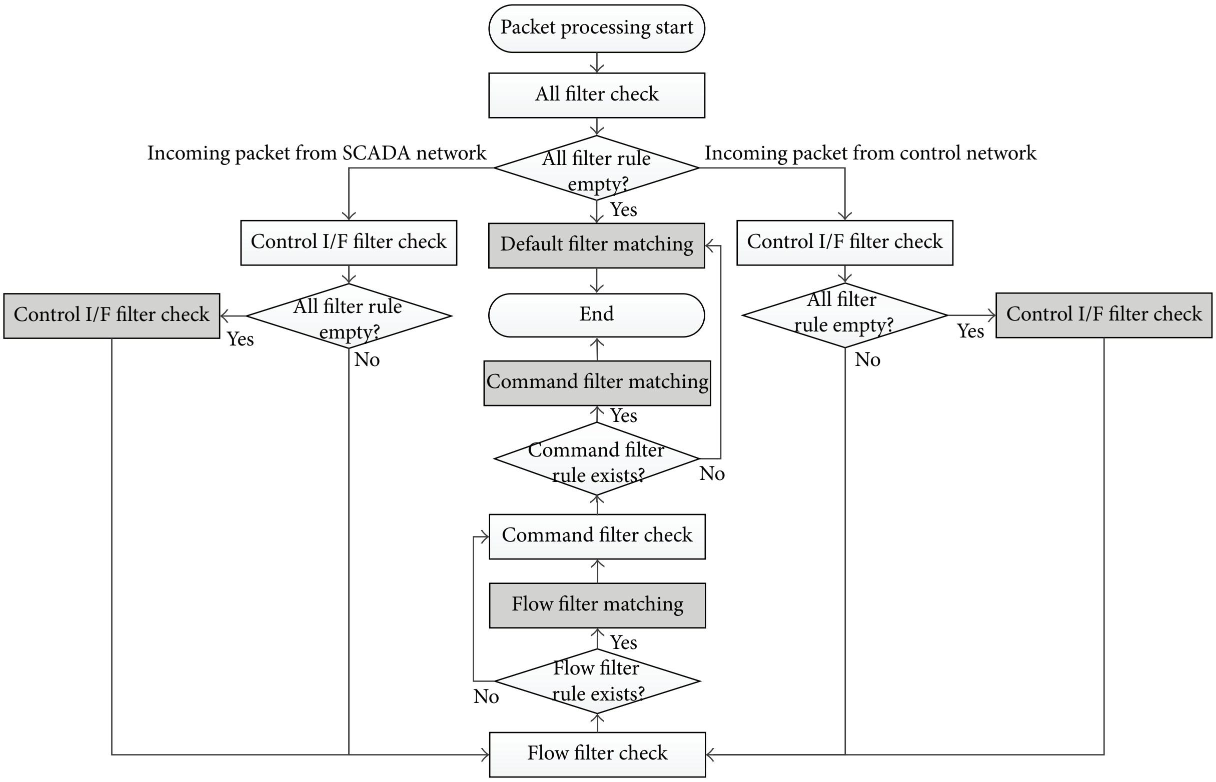

Figure 7: Overall packet processing flow through multiple access control filters.

were tested. The test result showed that the system was able to process 100% of incoming packets.

4. Relate Works

Intrusion detection/prevention system is the representative network security technique used to prevent various types of cyber-attacks. Intrusion detection/prevention technique can be divided into misuse detection method and anomaly detection method. Misuse detection method employs known attack patterns to construct signatures that are represented as rule sets. Anomaly detection method has potential to detect previously unknown attack. SCADA networks require minimal external network connections for security enhancement, except for control and monitoring purposes. Therefore, there are limitations in applying the intrusion detection/prevention system in SCADA networks because these legacy systems require signature updates from external networks and cause a high false positive ratio. Many researchers are currently engaged in developing security schemes to decrease various cyber threats and to enhance SCADA technologies [11–19]. Oman and Phillips [20] proposed comprehensive intrusion signatures for unauthorized access to SCADA devices using baseline-setting files for those devices. Morris et al. [21] introduced 50 intrusion detection rules developed to detect malicious activity on SCADA networks. The open-source IDS snort [22] was enhanced by SCADA related signatures and preprocessors for several SCADA protocols. Related works on anomaly detection approach mostly focused on traffic features derived from SCADA networks. Cheung et al. [23] introduced three model-based anomaly detection techniques. Their approach is to construct models that characterize the expected/acceptable behavior of SCADA traffic and detect attacks that cause SCADA systems to behave outside of the model. Düssel et al. [24] proposed a payload based real-time anomaly detection system. They rely on the computation of similarity between transport-layer packet payloads embedded in a geometric space. Barbosa et al. [25] propose an approach to improve the security of SCADA based on flow whitelisting. Continuing with flow based anomaly detection techniques, Siris and Papagalou [26] proposed an approach to apply network traffic monitoring technique based on the analysis of protocol headers and traffic flows.

5. Conclusion

SCADA systems are facing the threat of cyber-attacks due to utilizing standard open protocols and increasing connectivity to external networks. We summarize network and application protocol attacks that can occur in the SCADA networks and describe the characteristics of these attacks. Based on the survey, we have presented a multiple access control filtering approach based on whitelists for detecting abnormal traffic pattern and its system prototype. When detecting abnormal traffic, we use whitelists that can identify changes in normal characteristics during attack. Whitelists are automatically built without a need for human intervention in the analysis phase. The system can assist security administrators in identifying normal traffic by generating whitelists and decrease false positives. After the analysis phase has been completed, the proposed system inspects the traffic on communication between SCADA systems and field devices with whitelists. Our proposed system may effectively prevent unknown attacks using whitelists.

In future work, we plan to extend this work with network behavior based anomaly detection technique for detecting anomalous SCADA traffic. And then we intend to apply the approach in the other networks.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This paper is extended and improved form accepted paper of KCIC-2013/FCC-2014 conferences. And this work was supported by the IT R&D program of MSIP/KEIT [010041560, a development of anomaly detection and multilayered response technology to protect an intranet of a control system for the availability of pipeline facilities].

References

[1] K. Stouffer, J. Falco, and K. Scarfone, *Guide to Industrial Control Systems (ICS) Security*, NIST Special Publication 800. 82, 2008.

[2] B. Galloway and G. P. Hancke, "Introduction to industrial control networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 2, pp. 860–880, 2013.

[3] V. M. Igure, S. A. Laughter, and R. D. Williams, "Security issues in SCADA networks," *Computers & Security*, vol. 25, no. 7, pp. 498–506, 2006.

[4] I. D. A. Modbus, "Modbus application protocol specification v1. 1a," North Grafton Grafton, Mass, USA, 2004, http://www.modbus.org/specs.php.

[5] http://www.digitalbond.com/scadapedia/protocols/modbus-2/.

[6] http://www.metasploit.com/.

[7] http://www.tenable.com/products/nessus.

[8] https://code.google.com/p/modscan/.

[9] http://www.digitalbond.com/tools/basecamp/.

[10] http://www.digitalbond.com/tools/quickdraw/.

[11] H.-I. Kim, Y.-K. Kim, Y.-K. Kim, and J.-W. Chang, "A grid-based cloaking area creation scheme for continuous LBS queries in distributed systems," *Journal of Convergence*, vol. 4, no. 1, pp. 23–30, 2013.

[12] M. Yoon, Y.-K. Kim, and J.-W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, pp. 15–22, 2013.

[13] A. Sinha and D. Krishan Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, article 13, 2013.

[14] M. I. Malkawi, "The art of software systems development: reliability, avail-ability, maintainability, performance (RAMP)," *Human-Centric Computing and Information Sciences*, vol. 3, article 22, 2013.

[15] J. W. K. Gnanaraj, K. Ezra, and E. B. Rajsingh, "Smart card based time efficient authentication scheme for global grid computing," *Human-Centric Computing and Information Sciences*, vol. 3, article 16, 2013.

[16] H.-R. Lee, K.-Y. Chung, and K. -S. Jhang, "A study of wireless sensor network routing protocols for maintenance access hatch condition surveillance," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 237–246, 2013.

[17] K. Peng, "A secure network for mobile wireless service," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 247–258, 2013.

[18] D.-K. Kwon, K. Chung, and K. Choi, "A dynamic zigbee protocol for reducing power consumption," *Journal of Information Processing Systems*, vol. 9, no. 1, pp. 41–52, 2013.

[19] M. M. Weng, T. K. Shih, and J. C. Hung, "A personal tutoring mechanism based on the cloud environment," *Journal of Convergence*, vol. 4, pp. 37–44, 2013.

[20] P. Oman and M. Phillips, "Intrusion detection and event monitoring in SCADA networks," in *Critical Infrastructure Protection*, pp. 161–173, Springer, New York, NY, USA, 2007.

[21] T. H. Morris, B. A. Jones, R. B. Vaughn, and Y. S. Dandass, "Deterministic intrusion detection rules for MODBUS protocols," in *Proceedings of the 46th Annual Hawaii International Conference on System Sciences (HICSS '13)*, pp. 1773–1781, Wailea, Hawaii, USA, January 2013.

[22] http://www.snort.org.

[23] S. Cheung, B. Dutertre, M. Fong et al., "Using model-based intrusion detection for SCADA networks," in *Proceedings of the SCADA Security Scientific Symposium*, 2007.

[24] P. Düssel, C. Gehl, P. Laskov et al., "Cyber-critical infrastructure protection using real-time payload-based anomaly detection," in *Critical Information Infrastructures Security*, pp. 85–97, Springer, Berlin, Germany, 2010.

[25] R. R. R. Barbosa, R. Sadre, and A. Pras, "A first look into SCADA network traffic," in *Proceedings of the IEEE Network Operations and Management Symposium (NOMS '12)*, pp. 518–521, Maui, Hawaii, USA, April 2012.

[26] V. A. Siris and F. Papagalou, "Application of anomaly detection algorithms for detecting SYN flooding attacks," *Computer Communications*, vol. 29, no. 9, pp. 1433–1442, 2006.