A SURVEY ON THE USE OF OPEN-SOURCE FIREWALL FOR MAJOR SCADA PROTOCOLS

Hardik Maru
Student (M.Tech in Cyber Security)
Marwadi University, Rajkot, Gujarat
hardikmaru2001@gmail.com

Hepi Suthar
Assistant Professor
Marwadi University, Rajkot, Gujarat
hepisuthar@gmail.com

ABSTRACT

Supervisory Control and Data Acquisition (SCADA) system is control and monitoring system architecture used in modern industrial control systems and critical infrastructures. Many SCADA protocols have been developed to fulfill the essential requirements of SCADA systems, such as high availability, reliability, and real time response. Among those all protocols, Modbus, DNP3, and IEC 60870-5-104 (aka IEC 104) are the most used SCADA protocols. These protocols are developed to work over IP to enable the SCADA systems communication through the internet connectivity. As these protocols enable SCADA system communication from any remote location with the use of internet, it also opens the door to expose its existence and invites SCADA specific cyber-attacks. Several traffic filtering based security solutions are designed for SCADA systems, but Linux iptables based open-source firewall approach is one of the best among all. This paper presents an overview of SCADA Systems, and major three SCADA protocols with their architecture. Furthermore various SCADA specific attacks are discussed and iptables firewall is analyzed against those attacks.

Keywords: SCADA systems, SCADA security, network security, open source, firewalls, IEC 60870-5-104, Modbus, DNP3, Linux IPT ables.

1 INTRODUCTION

Mostly all the supervising, controlling, and monitoring needs of any critical infrastructure are managed by SCADA system, and therefore protecting it from any type of threat is critically important. Traditional SCADA systems has 3 major components, (A) Human Machine Interface (HMI), (B) Master Terminal Unit (MTU), (C) Remote Terminal Units (RTUs) or Programmable Logic Controllers (PLCs). Controlling and

monitoring is handled by a SCADA operator using HMI. PLCs or RTUs collect the data from physical end point devices such as sensors and actuators and send it to MTU. MTU is the heart of the system to manage core functions like communication, data collecting, processing, storing and representing.

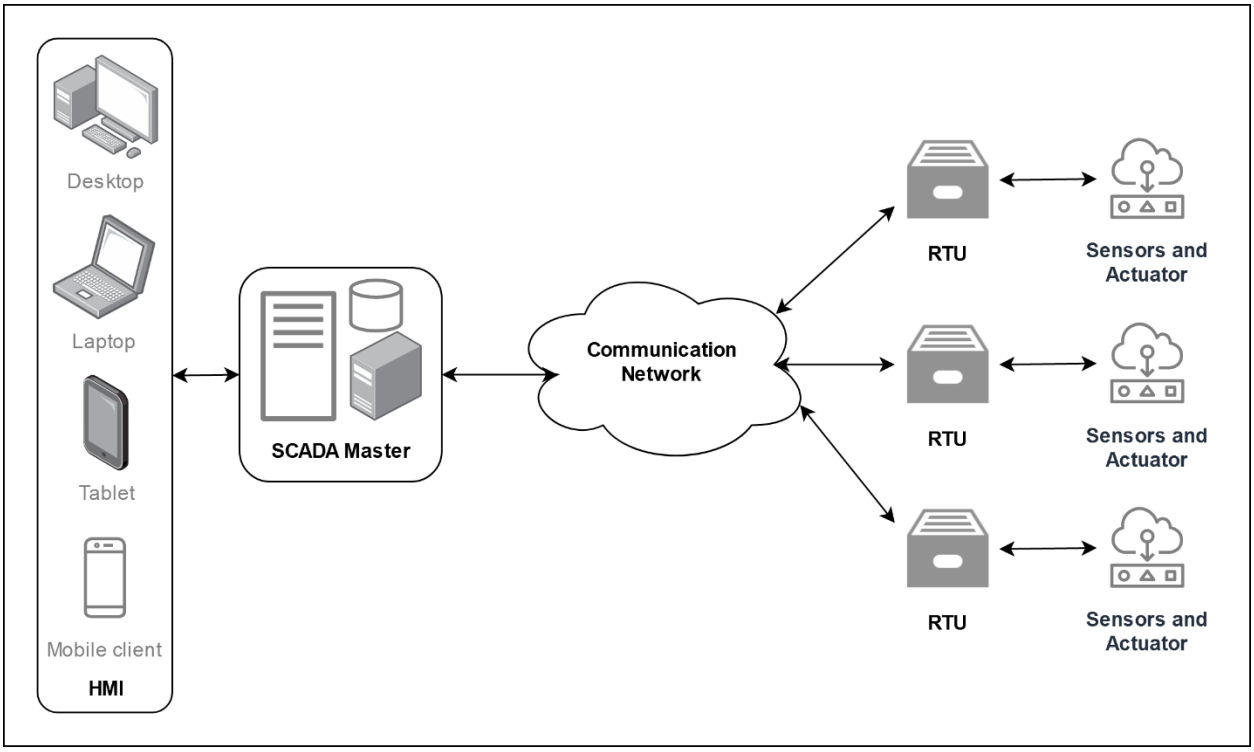
In recent decades, computing and communications have undergone considerable amount of changes. Computation is preferred on the go with a plenteous demand of mobility support in communicating [27], [28]. Due to the increasing users in wireless environment, communication paradigm also have shifted to the concept of Cognitive Radio Networks [25], [26] for better utilization of wireless spectrum. Needless to say, the advancement in handheld equipment and tremendous popularity of mobile application leads to necessity of timely analysis and security provisioning of communication environment. In specific to SCADA systems, SCADA protocols are designed to enable communication between all components of SCADA system. It transfers data and control commands between MTU and other components. Modbus, DNP3 and IEC 60870-5-104 are the three majorly used protocols in SCADA systems. Most of the protocols were initially designed to fulfill the operational requirements only. Over the time these protocols are extended to work over the internet but, it also invites various threat with this extension. Several cyber-attack incidents on SCADA are discussed in [14].

To fill this gap of security, traffic filtering-based detection system is better way to detect and prevent any cyber-attack. Linux Iptables is good option to use as firewall in SCADA system. Several researches have explored and examined its capabilities against SCADA attacks. In this paper, we provide the study of SCADA systems, most used three protocols, various attacks on those protocols, and analysis of iptables rules against those attacks.

Specifically the rest of the paper is sorted out as follows. Section II gives the SCADA system and its security overview. Section III introduces major three SCADA protocols with its architecture. Section IV provides the details of various firewall and IDS security solution based researches. Section V represents common attacks on major three protocols and analyzes whether iptables rule is defined for that particular attack or not. Section VI discusses the summary of whole works. Finally, Section VII concludes this paper and giving the new direction of research in this field.

2 SCADA SYSTEM AND SECURITY OVERVIEW

Figure 1 Generic SCADA Network Architecture



Supervisory Control and Data Acquisition (SCADA) system is a control and monitoring system architecture used in modern industrial control systems and critical infrastructures (e.g. food and beverage industries, power generation plants, petroleum industries, energy sector, transportation systems, sewage plants, manufacturing industries, recycling plants, and many more). Main objectives of SCADA system are: monitor, measure, data acquisition, data communication, controlling and automation. SCADA systems consist of software and hardware units such as Master Terminal Unit (MTU), Human Machine Interface (HMI), Remote Terminal Units (RTUs), Programmable Logic Controllers (PLCs), Sensors and Actuators, and Communication Network Infrastructure. MTU is a core part of SCADA system which manages communication, representing on interfaces, data collection, data processing, and data storing. RTU collects the data from connected sensors and actuators and further sends the collected data to MTU. RTUs are facilitated with storage, so it transmits the data to MTU on received command. HMI is used for monitoring and controlling the SCADA system with the help of an interface. Communication network is a link between all components of SCADA and it can be wired or wireless. Nowadays HMIs are extended to support many devices such as desktops, laptops, tablets, mobile phones, and screens.

SCADA systems are now more vulnerable to many threats [1] as modern SCADA systems are extended from local network to public network with an increased connections. Several studies discovers many vulnerabilities and attacks on SCADA systems. In [3], the authors have used attack tree methodology to discover security vulnerabilities in SCADA systems and have identified eleven attacks. In [5], the authors have classified various SCADA systems based cyber-attacks, such as attacks based on hardware and software, and communication stack based attacks. In [8], the authors provided detailed information about four major type of attacks against SCADA system.

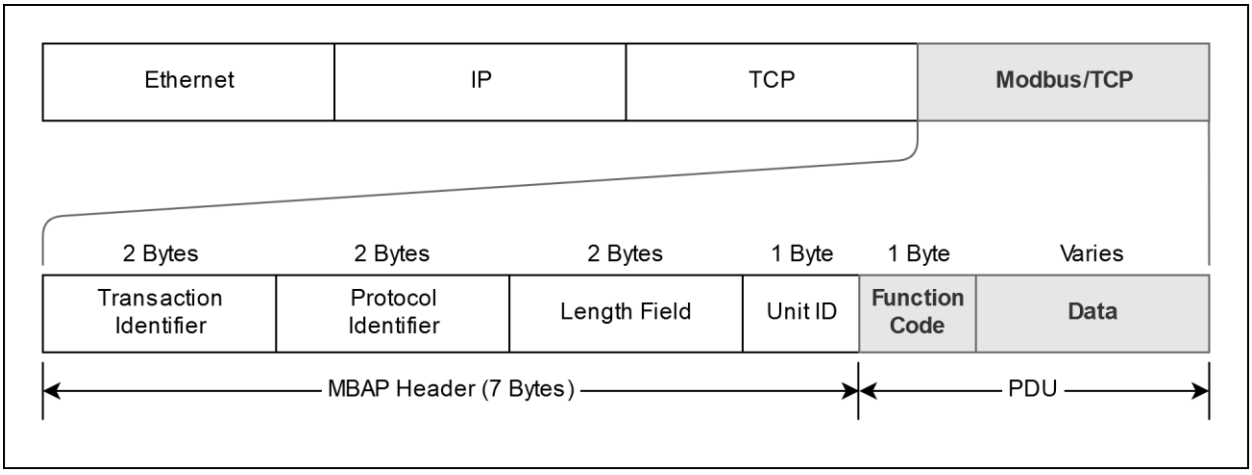
3 SCADA COMMUNICATION PROTOCOLS

SCADA communications protocols are designed to transfer data and control messages on industrial communication networks. Many SCADA protocols have been designed in recent decades, but most of these were initially designed where network security was not considered as a problem [3]. Because of it, many SCADA protocols are lacking when it comes to security, which leads to make the critical infrastructure vulnerable to threats.

Technical details of three major SCADA protocols are provided in the following subsections. This information enables the readers to understand the protocol overview, its architecture, various commands, and vulnerabilities/attacks on it.

3.1 Modbus

Figure 2 Modbus/TCP Protocol Architecture



Modbus/TCP is designed for Ethernet communication. It is an extension of Modbus/RTU protocol, which is a serial communication protocol designed by Modicon to use with PLCs of it. It uses request-response communication model where a device known as Modbus master is requesting or writing the information

and devices known as Modbus slaves supplies the information or acknowledge the execution state. There is one master and up to 247 slaves in one standard Modbus network. Each slave is uniquely assigned with slave address from 1 to 247.

A Modbus/TCP packet contains Modbus Application Protocol (MBAP) header of 7 bytes and Protocol Data Unit (PDU) with variable size. MBAP consists transaction and protocol identifier along with the length of packet, and slave identifier. While PDU consists two fields Function Code (FC) and Data Field which contains the actual Modbus command. FC is the 1 byte information which instruct the slave device which task to perform. Data field contains a detailed information of respective FC defined in 1st byte of PDU. This information could be Read/Write access method, data type, number of registers/coils, starting and ending address of registers/coils, data to write, sub-function code, device states, and etc.

Some Modbus function codes are publically standardized, which are [21]:

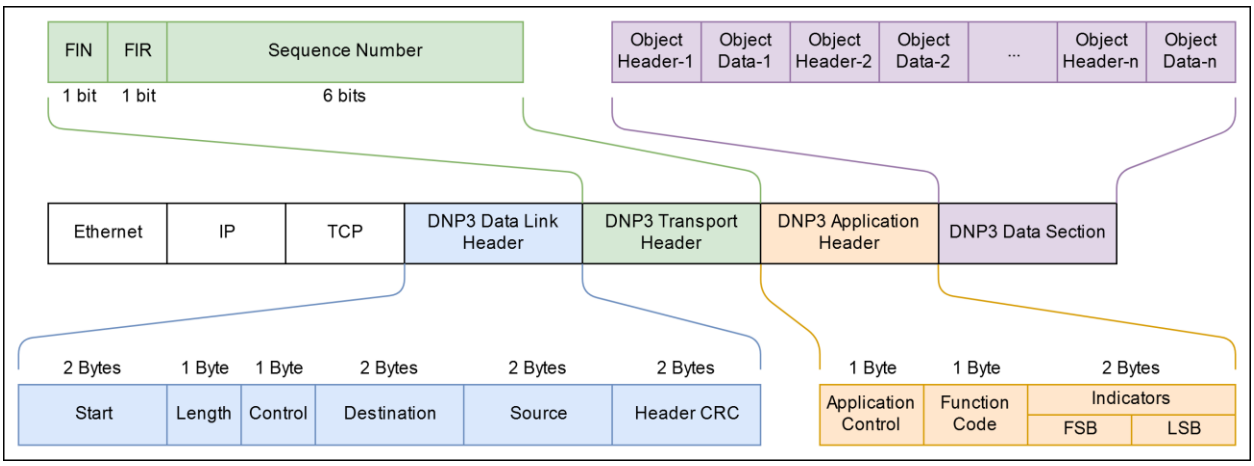
Table: 1 Standard Modbus Function Codes

Function Code	Hex Value	Action
01	0x01	Read Coils
02	0x02	Read Discrete Inputs
03	0x03	Read Holding Registers
04	0x04	Read Input Registers
05	0x05	Write Single Coil
06	0x06	Write Single Register
07	0x07	Read Exception Status
08	0x08	Diagnostics
11	0x0B	Get Communication Event Counter
12	0x0C	Get Communication Event Log
15	0x0F	Write Multiple Coils
16	0x10	Write Multiple registers
17	0x11	Report Slave ID
20	0x14	Read File Record
21	0x15	Write File Record
22	0x16	Mask Write Register
23	0x17	Read/Write Multiple registers
24	0x18	Read FIFO Queue
43	0x2B	Encapsulated Interface Transport
43/13	0x2B/0x0D	CANopen General Reference Request and Response PDU
43/14	0x2B/0x0E	Read Device Identification
65-72, 100-110	-	Reserved for User Defined Function Codes

3.2 DNP3

DNP3 is a group of telecommunications protocols that defines communication between SCADA components such as Master unit, RTUs, Intelligent Electronic Devices (IEDs) and other outstation devices. It is an open source protocol with many important features which makes it interoperable, robust, and one of the most efficient protocol in SCADA systems. It transmits data reliably in sequence of relatively small packets. It supports 4 types of communication mode, one-to-one, multi-slave, multi-master, and hierarchical [22]. In one-to-one, only one master station manage one slave. In multi-slave, one master station manages multiple slaves. In multi-master, one slave has been managed by multiple masters. In hierarchical, master station manages a slave master station as well along with other slaves.

Figure 3 DNP3 Protocol Architecture



A DNP3 message is divided into 4 main parts, (A) Data Link Header is of 10 bytes, which consists starting address (2 Bytes), length of message (1 Byte), a control field which contains data to manage flow of message (1 Byte), destination address where message needs to reach (2 Bytes), source address from where the message was originated (2 Bytes), and cyclic redundancy check code (2 Bytes). (B) Transport Header is of 1 byte, which consists FIR and FIN bits of 1 bit to indicate start and end of a sequence of frames, and sequence number (6 bits) denotes the frame sequence number. It can be any from 0 to 63 for initial frame and increments for each frame comes after initial and number rollover from 63 to 0. (C) Application Header is of 4 Bytes, which consists application control (1 Byte) to control flow of communication, function code (1 Byte) indicates the action to be performed, and indicators (2 Bytes) are used in reply message to pass useful information from outstation device to master station. Reply message can be confirmation, response, or an unsolicited response. (D) Data Section is of variable size and contains data objects with their header.

Some well-known public function codes of DNP3 are as below [22]:

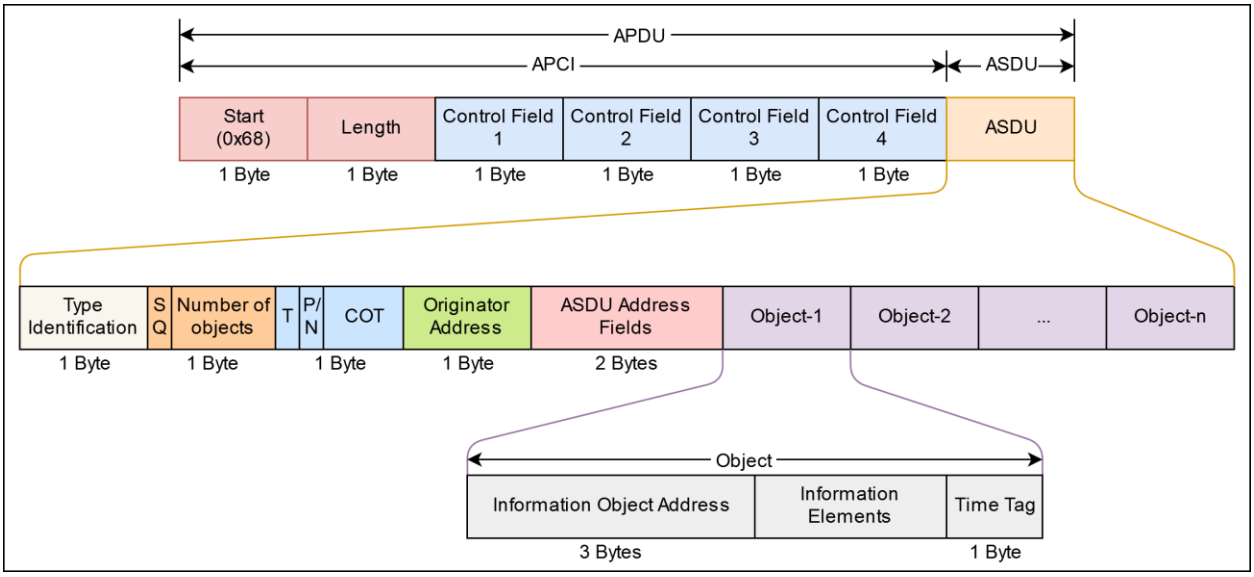
Table: 2 DNP3 Function Codes

Function Code	Hex Value	Action
01	0x01	Read
02	0x02	Write
03	0x03	Select
04	0x04	Operate
05	0x05	Direct Operate
06	0x06	Direct Operate, No Ack
07	0x07	Immediate Freeze
08	0x08	Immediate Freeze, No Ack
09	0x09	Freeze and Clear
10	0x0A	Freeze and Clear, No Ack
13	0x0D	Cold Restart
14	0x0E	Warm Restart
20	0x14	Enable Unsolicited Messages
21	0x15	Disable Unsolicited Messages
22	0x16	Assign Class
23	0x17	Delay Measurement
129	0x81	Response
130	0x82	Unsolicited Response

3.3 IEC 104 (IEC 60870-5-104)

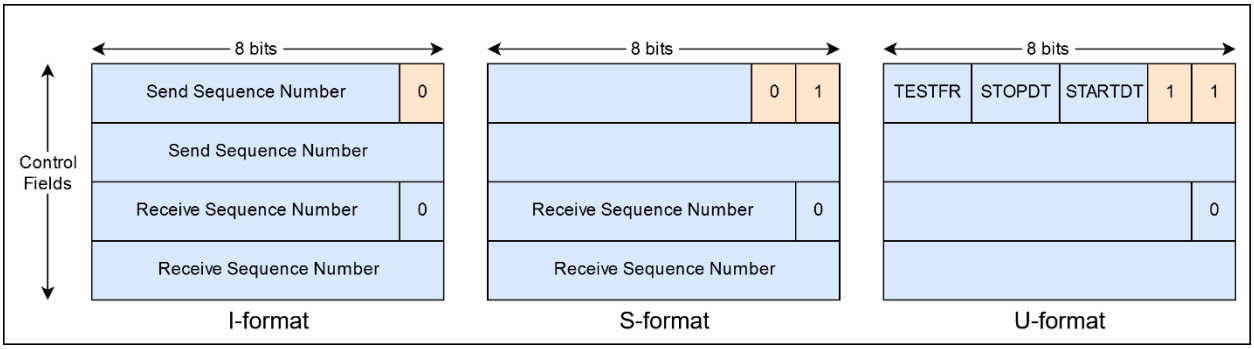
IEC 60870 standards are defined by the International Electrotechnical Commission (IEC) for SCADA systems in electrical and power systems. Part 5 of these standards consist transmission protocols for transmitting telecontrol messages between master station and outstation over standard TCP/IP network. IEC 60870-5-104 (IEC 104) was developed in 2000 and facilitate IEC 60870-5-101 with network access using standard transport profiles. It is a standard for SCADA systems with TCP/IP based communication network for monitoring and controlling geographically pervasive processes.

Figure 4 IEC 104 Protocol Architecture



IEC 104 can be of fixed length and variable length. Fixed length just contains APCI (Application Protocol Control Information) in APDU (Application Protocol Data Unit), while variable length have APCI and ASDU (Application Service Data Unit) in APDU. APCI starts with Start field (1 Byte) with fixed value 0x68 followed by length of APDU (1 Byte), and four CF (control fields) (1 Byte each). There are 3 types of APCI frame (A) I-format (information transfer format) where last bit of CF1 is 0, (B) S-format (numbered supervisory functions) where last bits of CF1 are 01, (C) U-format (unnumbered control functions) where last bits of CF1 are 11. Control fields are elaborated in below figure 5. ASDU contains type identification field of 1 Byte, Structure Qualifier (SQ) bit specifies the addressing of information objects or elements, number of objects defines the number of objects or elements ASDU contains, T bit indicates ASDU is generated for test conditions, P/N bit is used for positive or negative confirmation, cause of transmission (COT) is six-bit code that control the message routing and interpretation of information when it reach the destination, originator address (ORG) of 1 Byte is used to identify controlling station in case of more than one else there is no originator address, ASDU address of 2 Bytes is also called as common address which is associated with the information objects in ASDU. Each information object contains information object address (IOA) which act as a destination address when it is used in a control direction and as a source address when it is used in monitor direction.

Figure 5 IEC 104 Protocol APCI Frames



Some common command types of IEC 104 are [22]:

Table: 3 IEC 104 Common Command Types

Command Type	Reference	Description
45	C_SC_NA_1	Single command
46	C_DC_NA_1	Double command
47	C_RC_NA_1	Regulating step command
58	C_SC_TA_1	Single command with time tag CP56Time2a
59	C_DC_TA_1	Double command with time tag CP56Time2a
60	C_RC_TA_1	Regulating step command with time tag CP56Time2a
48	C_SE_NA_1	Setpoint command, normalized value
49	C_SE_NB_1	Setpoint command, scaled value
50	C_SE_NC_1	Setpoint command, short floating point value
61	C_SE_TA_1	Setpoint command, normalized value with time tag CP56Time2a
62	C_SE_TB_1	Setpoint command, scaled value with time tag CP56Time2a
63	C_SE_TC_1	Setpoint command, short floating point value with time tag CP56Time2a
103	C_CS_NA_1	Clock synchronization command
105	C_RP_NC_1	Reset process command
107	C_TS_TA_1	Test command with time tag CP56Time2a
101	C_CI_NA_1	Counter interrogation command
102	C_RD_NA_1	Read command

4 FIREWALL/IDS FOR SCADA SYSTEMS

In this section, we discussed various researches based on filtering solutions for all three major protocols of SCADA systems. Several work uses Linux iptables while some other uses different approaches. It includes the information about the work and their limitations.

In [16], critical state-based filtering system, the authors have introduced an innovative state analysis based filtering system for SCADA systems. They designed a firewall architecture for the Modbus protocol and DNP3 protocol based SCADA systems with aim to detect off-sequenced command of complex process and block it. This filtering mechanism can secure the SCADA systems only against specifically crafted attack which uses set of commands to disturb the process. While all other classes of attacks can still affect the SCADA systems. Early warning system for the critical state is really helpful, but it cannot be used as solo firewall. However this approach very helpful for enhancing the SCADA firewalls.

In [17], [14] and [15], the authors have identified the potential of the open source Linux iptables based firewall solution for network security and SCADA system security. Some of the common network based attacks were simulated by authors in [17] and tested to examine the capabilities of iptables. Many open source firewall solutions are being used for network security, but use of it in SCADA system were not properly investigated. So, in other two researches, the authors used iptables as a firewall solution in the SCADA systems. For dynamic packet inspection of data, the authors have created iptables rules by utilizing the advance features of iptables. Rules have been defined, tested and validated for its ability to detect various simulated attacks only on Modbus protocol, and DNP3 protocol based SCADA systems. However, rules represented in these papers are for only few attacks, while some more rules needs to be developed for other common attacks on Modbus and DNP3 protocols. Furthermore no work has been accomplished to determine the capabilities of iptables against IEC 104 protocol based SCADA systems.

In [13], SCADAWall model is developed and presented by the authors. SCADAWall consists 3 algorithms, (A) CPI (Comprehensive Packet Inspection), (B) PIPEA (Proprietary Industrial Protocol Extension Algorithm), and (C) OSDA (Out of Sequence Detection Algorithm). A CPI uses the iptables, but extends the dynamic packet inspection technique. It checks the data field as well along with the header to ensure that only trusted payload and packets accepted. A PIPEA enables the SCADAWall users to add any new proprietary protocol and create rules for it. An OSDA is defined to resolve the issue of off-sequenced command like we discussed above for [16]. This model is specifically developed and tested against Modbus protocol based SCADA system.

In [18], [19], and [20], the authors have presented various approaches such as anomaly detection, rule-based IDS and stateful IDS with the use of DPI (Deep Packet Inspection). Anomaly detection based approach is

built on Bro platform with capability of detecting any kind of malicious threats, even a zero-day threats. Authors have tested this approach on IEC 104 SCADA protocol with just three different attacks and represented the results of it. There are many other attacks which needs to be tested with this approach. Also authors have used Bro tool to build the proposed IDS system, but additional efforts are needed in writing parser to convert the network data into Bro compatible format. A rule-based IDS approach is implemented using snort rules, with the use of a DPI (Deep Packet Inspection) method. It uses signature-based approach to detect the known attacks, and model-based approach to detect the unknown attacks. Several attacks were tailored specifically for IEC 104 protocol based SCADA system, tested against both rule-based approaches and detection, and the result is represented by the authors. According to our analysis, this approach is the best security solution among all three different approaches. The stateful IDS approach also uses the DPI method and specifically designed, implemented, and validated for IEC 104 based SCADA systems. However the proposed approach is limited to 8 different alarm states, mainly representing timer overtime state. Furthermore, network based or protocol based attacks cannot be detected or prevented using this approach. From all these three IDS approaches, no one investigated the use of open source Linux iptables rules to prevent the attacks on SCADA systems.

In [24], the authors have studied and analyzed various firewall systems for Smart Grid (SG) paradigm. Authors provided overview of seven different firewall solutions and concluded that most of the paper examined Modbus and DNP3 protocols only, while SCADA protocols like IEC 61850 and IEC 60870 still need more work.

From all these different solutions, our analysis determines that open source Linux iptables is really good approach for SCADA security. However till now, only Modbus and DNP3 protocols based only few attacks are examined. While capability of iptables against IEC 104 protocol based attacks is totally unexplored.

5 COMMON ATTACKS AND IPTABLES RULES

As SCADA systems are controlling critical infrastructures, an attacks on SCADA systems can damage the system or disrupt the critical operations. Further it can lead to hazardous damages to the environment, monetary losses, and most dangerous is human losses. In this section, we discussed attacks identified on Modbus, DNP3, and IEC 104 SCADA protocols and their corresponding iptables rules.

5.1 Attacks on Modbus Protocol [2], [3], [5], [6], [14]

Table: 4 Attacks on Modbus Protocol

No.	Attack Goal	Methodology	IPTables Rule Defined?
(M1)	**Gain SCADA system access**	Access to wireless PCN, Third party access, Access to remote field sites, or Use of SCADA transmission media.	No
(M2)	**Identify Modbus device**	FC (Function Code) 43, and Sub FC 14 is used for reading device identification.	Yes
(M3)	**Disrupt master-slave communication**	Accepting communication/command from an unauthorized IPs.	No
(M4)	**Disable/Compromise Master/Slave**	Accepting operation commands from an unauthorized IPs.	No
(M5)	**Unauthorized read/write data**	Accepting read/write commands from an unauthorized IPs.	Yes
(M6)	**Clear counters and diagnostic registers**	FC 08, and Sub FC 10 is used for clearing counters and diagnostic registers.	Yes
(M7)	**Remote restart**	FC 08, and Sub FC 01 is used for restarting the Modbus device remotely.	Yes
(M8)	**Force PLC into listen-only mode**	FC 08, and Sub FC 04 is used to put PLC into listen-only mode.	Yes
(M9)	**Report server information**	Attacker can use FC 17 to enumerate PLCs.	Yes
(M10)	**Clear overrun counters and diagnostic flags**	FC 08, and Sub FC 20 is used for clearing overrun counters and diagnostic flags.	No
(M11)	**Broadcast message spoofing**	Attacker sends faked broadcast messages.	No
(M12)	**Direct slave control**	By identity spoofing, attacker access the slave device.	No
(M13)	**Passive reconnaissance**	Passively sniffing network traffic.	No
(M14)	**Response delay**	Delaying the response from slave devices to the master.	No
(M15)	**Man in the middle attack**	Access to SCADA network and put device between master and outstation device to sniff and modify the messages.	No

5.2 Attacks on DNP3 protocol [2], [4], [15]

Table: 5 Attacks on DNP3 Protocol

No.	Attack Goal	Methodology	IPTables Rule Defined?
(D1)	**Gain SCADA system access**	Access to wireless PCN, Third party access, Access to remote field sites, or Use of SCADA transmission media.	No
(D2)	**Passive reconnaissance**	Passively sniffing network traffic.	No
(D3)	**Baseline response replay**	Attacker sends spoofed message as a response to master and as a command to an outstation devices.	No
(D4)	**Man in the middle attack**	Access to SCADA network and put device between master and an outstation device to sniff and modify the messages.	No
(D5)	**Transport sequence modification**	Attacker sends spoofed message in fragmented message sequence.	No
(D6)	**Outstation write attack**	FC 2 is used to writes data on an outstation device.	No
(D7)	**Clear objects attack**	FC 9, and 10 are used to freeze and clear the data objects.	Yes
(D8)	**Outstation data reset**	FC 15 is used to reinitialize the data objects on outstation.	No
(D9)	**Configuration capture attack**	Fifth bit in second byte of the IIN is set in the message informs master to resend the configuration file again to an outstation.	No
(D10)	**Length overflow attack**	Incorrect value is set in the length field.	No
(D11)	**DFC flag attack**	Attacker sets DFC flag to indicate an outstation as busy.	No
(D12)	**Reset function attack**	FC 1 is used to reset the user process on the outstation device.	No
(D13)	**Unavailable function attack**	FC 14 or 15 is used to make the outstation device unavailable to the master.	No
(D14)	**Destination address alteration**	Attacker alter the destination address field to affect the communication.	No
(D15)	**Fragmented message interruption**	FIR and FIN flags are set in wrong fragmented message to disrupt communication.	No
(D16)	**Outstation application termination attack**	FC 18 is used by attacker to terminate the applications running on an outstation.	Yes

(D17)	**Disable unsolicited responses attack**	FC 21 is used by attacker to stop unsolicited response update from an outstation to master.	Yes
(D18)	**Warm restart attack**	FC 14 is used to restart the communication in the outstation. Continuous stream of this attack can lead to DoS attack as well.	Yes
(D19)	**Cold restart attack**	FC 13 is used to restart the outstation device.	Yes
(D20)	**Broadcast message spoofing**	Attacker sends faked broadcast messages.	Yes

5.3 Attacks on IEC 104 protocol [19], [18], [7], [9], [23]

Table: 6 Attacks on IEC 104 Protocol

No.	Attack Goal	Methodology	IPTables Rule Defined?
(I1)	**Gain SCADA system access**	Access to wireless PCN, Third party access, Access to remote field sites, or Use of SCADA transmission media.	No
(I2)	**IEC/104 port communication**	Establish spoofed connection or hijack the established connection between client and server.	No
(I3)	**Spontaneous messages storm**	Attacker sends huge amount of false spontaneous messages.	No
(I4)	**Unauthorized read command**	Unauthorized client sends command to read the field device.	No
(I5)	**Unauthorized interrogation commands**	Unauthorized client sends interrogation command against server.	No
(I6)	**Remote control commands or remote adjustment commands**	Unauthorized client sends control or adjustment command.	No
(I7)	**Reset process command**	Unauthorized client sends command with type identification 69H to reset the process of server.	No
(I8)	**Broadcast request**	Attacker sends faked broadcast messages.	No
(I9)	**Buffer overflow**	Incorrect packet length.	No
(I10)	**Network reconnaissance**	Port scanning from known and unknown hosts	No
(I11)	**Man in the middle attack**	Access to SCADA network and put device between master and an outstation device to sniff and modify the messages.	No

(I12)	**Single command attack**	Unauthorized client sends a single command to execute.	No
(I13)	**Modification and injection attack**	Command is modified or injected in SCADA system using MiTM.	No

6 DISCUSSION

Several papers have examined the SCADA security issues with detailed information of major protocols used in SCADA systems, attacks on those protocols, attack impacts, and use of different methodology as a countermeasure. In [1] the authors provide technical details of various SCADA protocols along with their corresponding packet structure. Among all those protocols, Modbus, DNP3 and IEC 60870-5-104 (aka IEC 104) are the most widely used protocols in SCADA systems. Different vulnerabilities and attacks on above three major protocols have been identified by the authors in [2], [3], [4], [5], [6], [7], [8], [9], and [10]. Moreover, in [11] the authors have implemented a secure Modbus protocol with the help of cryptography, in [12] the authors have presented a security framework for DNP3 protocol. In [16], [18], [19] and [20] the authors have presents various firewall/intrusion detection system (IDS) solutions with different approaches. In [17], the authors have used iptables as a firewall for network based attacks. Furthermore in [13], [14], and [15] the authors implements Linux iptables as a firewall for SCADA systems. Although lot of research work has been accomplished in direction of firewall/IDS for SCADA System, but most of them are for Modbus protocol and DNP3 protocol and only few for IEC 104 protocol. Also we did not find any paper that examines or evaluates Linux iptables on IEC 104 protocol.

7 CONCLUSION AND FUTURE PLANS

This paper presented the review of SCADA systems and three major protocols used in SCADA network communication. We have analyzed various traffic filtering based security solutions and found open-source Linux iptables are good and effective solution to secure SCADA systems. We have analyzed several attacks on all these three protocols and determined whether an iptables based rules are defined for those attacks or not. Our evaluation shows that for Modbus and DNP3 protocols, iptables rules are defined for only few attacks and lacking for many of the attacks. For IEC 104 protocol, iptables based approach is totally unexplored and no rule is defined for any of the attacks.

In the future plans,

- We will investigate iptables based firewall system against SCADA systems which uses IEC 104 protocol.
- We will develop rules for attacks of Modbus and DNP3 protocols where it is lacking.

REFERENCES

1. Francia, G. A. III., Francia, X. P., Pruitt, A. M.: Towards an In-depth Understanding of Deep Packet Inspection Using a Suite of Industrial Control Systems Protocol Packets. In: Journal of Cybersecurity Education, Research and Practice, vol. 2016: no. 2, article 2 (2016).
2. Drias, Z., Serhrouchni, A., Vogel, O.: Taxonomy of attacks on industrial control protocols. In: International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS) (2015).
3. Byres, E. J., Franz, M., Miller, D.: The Use of Attack Trees in Assessing Vulnerabilities in SCADA Systems. In: IEEE Conf. International Infrastructure Survivability Workshop (IISW'04), Institute of Electrical and Electronics Engineers. Lisbon, (2004).
4. East, S., Butts, J., Papa, M., Shenoi, S.: A Taxonomy of Attacks on the DNP3 Protocol. In: International Conference on Critical Infrastructure Protection, pp. 67-81. Springer, Berlin, Heidelberg (2009).
5. Zhu, B., Joseph, A., Sastry, S.: A taxonomy of cyber-attacks on SCADA systems. In: Proc. IEEE Int. Conf. Internet Things, Int. Conf. 4th Int. Conf. Cyber, Phys. Soc. Comput., pp. 380–388 (2011).
6. Huitsing, P., Chandia, R., Papa, M., Shenoi, S.: Attack taxonomies for the Modbus protocols. In: International journal of critical infrastructure protection, volume 1, pages: 37-44 (2008).
7. Grammatikis, P. R., Sarigiannidis, P., Giannoulakis, I., Kafetzakis, E., Panaousis, E.: Attacking IEC-60870-5-104 SCADA Systems. In: IEEE World Congress on Services (SERVICES) (2019).
8. Morris, T. H., Gao, W.: Industrial Control System Cyber Attacks. In: 1st International Symposium for ICS & SCADA Cyber Security Research 2013 (ICS-CSR 2013) (ICSCSR) (2013).
9. Parcharidis, M. D.: Simulation of cyber-attacks against SCADA systems (2018).
10. Bin, Z., Cheah: Testing and Exploring Vulnerabilities of the Applications Implementing IEC 60870-5-104 Protocol. In: Institute for telematikk, Sweden (2008).
11. Fovino, I. N., Carcano, A., Masera, M., Trombetta, A.: Design and Implementation of a SecureModbus Protocol. In: Third Annual IFIP WG 11.10 International Conference on Critical Infrastructure Protection. Hanover, New Hampshire, USA (2009).
12. Majdalawieh, M., Parisi-Presicce, F., Wijesekera, D.: DNPSec: Distributed Network Protocol Version 3 (DNP3) Security Framework. In: K. Elleithy et al. (eds.), Advances in Computer, Information, and Systems Sciences, and Engineering, 227–234. Springer, Dordrecht (2007).
13. Li, D., Guo, H., Zhou, J., Zhou, L., Wong, J. W.: SCADAWall: A CPI-Enabled FirewallModel for SCADA Security. In: Computers & Security, volume-80, pages 134-154 (2018).
14. Nivethan, J., Papa, M.: On the use of open-source firewalls in ICS/SCADA systems. In: Information Security Journal:A Global Perspective, Volume 25 Issue 1-3, Pages 83-93, Taylor & Francis, Inc. Bristol, PA, USA (2016).

15. Nivethan, J., Papa, M.: A Linux-based firewall for the DNP3 protocol. In: Technologies for Homeland Security (HST), IEEE Symposium on, pp. 1-5. IEEE (2016).
16. Fovino, I. N., Coletta, A., Carcano, A., Masera, M.: Critical state-based filtering system for securing SCADA network protocols. In: IEEE Transactions on industrial electronics, vol. 59, no. 10, pp. 3943-3950 (2012).
17. Mihalos, M. G., Nalmpantis, S. I., Ovaliadis, K.: Design and Implementation of Firewall Security Policies using Linux Iptables. In: Journal of Engineering Science and Technology Review 12 (1) 80 - 86 (2019).
18. Udd, R., Asplund, M., Nadjm-Tehrani, S., Kazemtabrizi, M., Ekstedt, M.: Exploiting Bro for Intrusion Detection in a SCADA System. In: CPSS '16 Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security, Pages: 44-51 (2016).
19. Yang, Y., McLaughlin, K., Littler, T., Sezer, S., Pranggono, B., Wang, H. F.: Intrusion Detection System for IEC 60870-5-104 based SCADA networks. In: IEEE Power & Energy Society General Meeting (2013).
20. Yang, Y., McLaughlin, K., Sezer, S., Yuan, Y. B., Huang, W.: Stateful intrusion detection for IEC 60870-5-104 SCADA security. In: IEEE PES General Meeting | Conference & Exposition (2014).
21. The Modbus Organization. "Modbus Application Protocol Specification v1.1b3".
22. Clarke, G., Reynolds, D.: Practical Modern SCADA Protocols: DNP3, IEC 60870.5 and Related Systems. Newnes, Oxford, United Kingdom (2004).
23. Maynard, P., McLaughlin, K., Haberler, B.: Towards Understanding Man-In-The-Middle Attacks on IEC 60870-5-104 SCADA Networks. In: Queen's University Belfast (2014).
24. Radoglou-Grammatikis, P., Sarigiannidis, P., Liatifis, T., Apostolakos, T., Oikonomou, S.: An Overview of the Firewall Systems in the Smart Grid Paradigm. In: Global Information Infrastructure and Networking Symposium (GIIS), Thessaloniki, Greece, 2018, pp. 1-4 (2018).
25. Dutta, N., Sarma, H. K. D., Polkowski, Z.: Cluster based routing in cognitive radio Adhoc networks: reconnoitering SINR and ETT impact on clustering. In: Com. Com., (Elsevier), pp. 10-20, vol. 115 (2018).
26. Dutta, N., Sarma, H. K. D.: A probability based stable routing for cognitive radio Adhoc networks. In: Wire. Net., (Springer), vol. 23(1), pp. 65-78 (2017).
27. Dutta, N., Misra, I. S.: Multilayer hierarchical model for mobility management in IPv6: a mathematical exploration. In: Wire. Pers. Comm. (Springer), vol.78 (2), pp.1413-1439 (2014).
28. Dutta, N., Misra, I. S.: Mathematical modelling of HMIPv6 based network architecture in search of an optimal Performance. In: IEEE 15th ADCOM, pp. 599-605, Guwahati, India (2007).