Burst-based Anomaly Detection on the DNP3 Protocol*

Jeong-Han Yun[1*], Sung-Ho Jeon[2], Kyoung-Ho Kim[3], and Woo-Nyon Kim[4]

The Attached Institute of ETRI,
P.O.Box 1, Yuseong, Daejeon, 305-600, Korea
{[1*]*dolgam*, [2]*sdeva*, [3]*lovekgh*, [4]*wnkim*}*@ensec.re.kr*

Abstract

The potential effectiveness of cyber-attacks against SCADA systems could be increased because they are connected to the Internet for several purposes. The Distributed Network Protocol Version 3 (DNP3) protocol is widely used in SCADA systems as a means of communicating observed sensor state information back to a control center. Previous DNP3 security researches are based on such specifications as attack signatures and protocol-based authorization. The provision of an exact and detailed specification is a good security criterion, but the drafting of proper specifications tends to be a time-consuming and error-prone process. In general, utilities that use the DNP3 protocol repeat their own limited operations, so a whitelist-based approach is clearly suitable for network intrusion detection. A burst *is a group of consecutive packets with shorter inter-arriving time than packets arriving before or after the burst of packets. When utilities communicate on the DNP3 protocol, one transaction at the application-level is mapped to one burst. We collected and analyzed the DNP3 network traffic of a real-world SCADA system and, based on the results obtained from the analysis, produced a burst-based whitelist model for utilities using the DNP3 protocol. The proposed model can be used for intrusion detection and abnormal behaviors in the SCADA system.*

Keywords: *DNP3; SCADA; Network Anomaly Detection; Whitelist; Burst*

1 Introduction

The cyber-attack is a new and essential weapon of modern warfare. Cyber-attacks, which generally target control systems, can be used to cause social and economic disasters. The Fukushima Daiichi nuclear disaster reminds people of the absolute importance of control system safety. Control systems are targets of the highest priority for cyber terrorists, leading many countries to reinforce their investment in the area of cyber security.

There have been numerous cyber-attacks against utilities in control systems. Tom Donahue, a former senior CIA analyst, stated at the Process control Security Summit held in January 2008 that “We have information, from multiple regions outside the United States, on cyber intrusions into utilities, followed by extortion demands”. Stuxnet, a malware written to attack industrial systems manufactured by global megabrand Siemens, damaged

[*] This paper is an extended version of a paper that was presented at ISA 2013 [1].

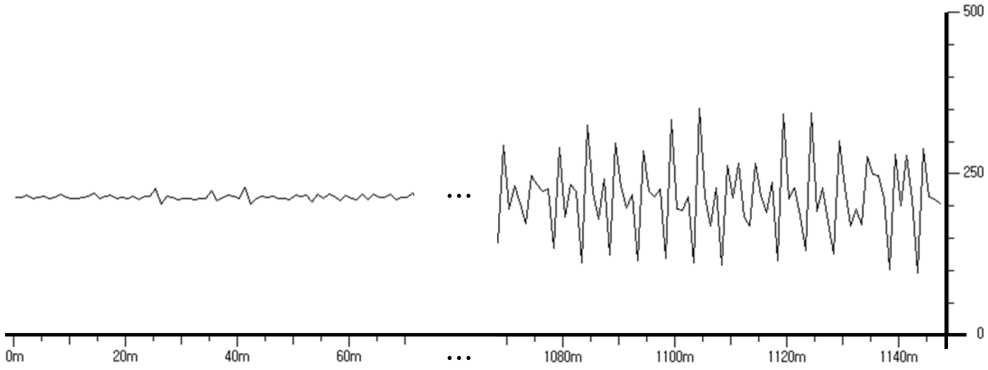


Figure 1. Packets/minute graph of a working SCADA system

physical facilities of Iran's nuclear system in 2010, while a variation of Stuxnet emerged and struck pone of the largest power plants in Iran on December 2012. Cyber-attacks have evolved and become more frequent, making the cyber security of control systems a most important and pressing issue.

The control system is managed by the Supervisory Control and Data Acquisition (SCADA) system[2], which is used to monitor and control plant or equipment in the telecommunications, water and waste control, and energy industries among others. Distributed Network Protocol Version 3 (DNP3) is used by SCADA systems to communicate between the master host and the outstation units. When SCADA systems were initially designed, cyber security problems were not taken into considerations. However, not only the introduction of new utilities such as smart meters, but also tighter integration may reveal new vulnerabilities to cyber-attack, so SCADA systems need network intrusion detection systems (IDS) for protection against network.[3]

There are two sub-approaches under the security research: host-based and network-based approaches. However, the host-based approach is difficult to apply to SCADA systems. The installation of a security agent may be impossible on some utilities in SCADA systems. Another limitation of the host-based approach for a SCADA system is the deterioration of availability and effectiveness.

Network-based approach can be divided into two approaches, namely the blacklist and whitelist approaches. The blacklist approach is difficult to apply to SCADA systems. SCADA systems are isolated from the outside physically, but they remain imperfect, as mentioned earlier. This property makes the signature updating of a blacklist difficult. In addition, access points which used to update the signature of a blacklist can become new vulnerabilities. The difficulty of obtaining the attack signatures of a blacklist early on is the fundamental drawback of this approach. Thus, the whitelist network-based approach is more suitable for zero-day attacks and insider threat which target control systems.

During ten-day period, we collected the network traffic in a real-world SCADA system. It includes one master and 42 outstation using the DNP3 protocol. The DNP3 network traffic volume in that ten-day period amounted to 887MB. Figure 1 shows ppm (packets/minute) graphs of the collected DNP3 traffic. In some time zone, the traffic has a strict regularity such as self-similarity. But, as three different graphs in Figure 1 show, the regularity is not continued long time, so it hardly applied to real network traffic of a SCADA system.[4]

A *burst* is a group of consecutive packets whose inter-packet arrival time is shorter than the threshold of inter-packet arrival time[5][1]. When utilities communicate on the DNP3 protocol, one transaction at the application-level is mapped to one burst. In this paper, we are proposing a new whitelist model for utilities using the DNP3 protocol. The major difference of our whitelist model from previous research is burst-based approach. Using this approach, we can use application-level characteristics without such deep packet inspection (DPI) as packet recombination. To confirm the validity of the model proposed herein, we extracted whitelist rules for the collected network traffic based on the model and analyzed how the whitelist rules can be used to detect cyber-attacks.

The rest of the paper is organized as follows: Section 2 introduce the DNP3 protocol and the security research for the DNP3 protocol; Section 3 presents the proposed whitelist model using the burst-based approach; Section 4 presents an analysis of how to detect cyber-attacks using the proposed whitelist model; Section 5 discusses the expansion of our whitelist model; and Section 6 presents the conclusion.

2 Related Work

We will explain the overview of the DNP3 protocol and introduce previous works on SCADA network intrusion detection considering SCADA protocols.

2.1 Overview of DNP3

The DNP3 protocol was developed by Westronic, Inc. (now GE Harris) in the early 1990s [6]. The protocol defines how utilities in SCADA system communicate data and commands.

DNP3 supports three communication modes between a master and outstations: peer-to-peer transaction, broadcast transaction, unsolicited response. In a peer-to-peer transaction, a master sends a request to an outstation device, which responds with a reply. In a broadcast transaction, a master sends a request to all the outstations in the network. The outstations do not reply to the broadcast message. DNP3 allows that an outstation can provide unsolicited responses event data to masters without a request.

The DNP3 specification contains three protocol layers: the application layer, the transport layer, and the data link layer. The data link layer provides link connection services such as error detection and network addressing. The transport layer is used for the fragmentation and reassembly of the application layer fragments. The application layer structures data access requests for masters and creates data responses from outstations.

Figure 2 shows the communication process between a master and an outstation. The master can address an individual outstation or can initiate a broadcast message to all outstations. Outstations return a message (response) to requests that are addressed to them individually. If the response message is not sent within a time constraint (timeout), the communication is cancelled. The DNP3 protocol establishes the format for the master's request message by placing it into the outstation (or broadcast) address. The function code defines the interaction between masters and outstations; for example, a master may use function codes to read or write data, control operations or applications, and transfer files. The data object defines the data and its attributes between masters and outstations. a function code defining the requested action, any data to be sent, and an error-checking

[1] http://www.ietf.org/proceedings/66/slides/ippm-10.pdf

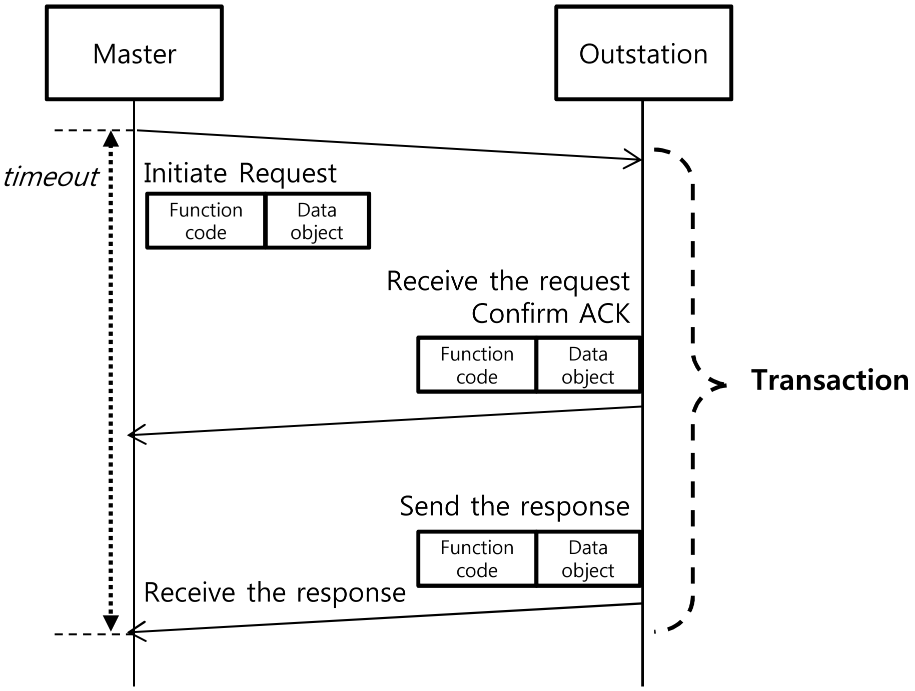


Figure 2. DNP3 communication process

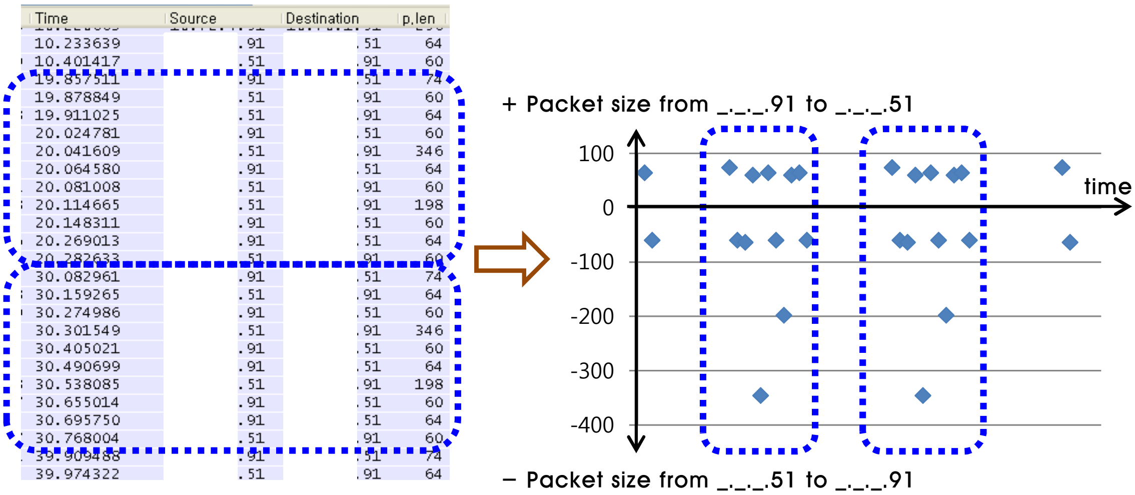
field. The application layer object header defines the properties of the data objects, and consists of the object type, qualifier, and optional range fields. The outstation's response message is also constructed using the DNP3 protocol. It contains fields confirming the action taken (if requested), any data to be returned, and an error checking field. If an error occurred in receipt of the message, or if the outstation is unable to perform the requested action, the outstation will construct an error message and send it as its response.

The DNP3 data parameters are named points, and related points are grouped together into a point type such as analog input and counters. The data object defines the data and its attributes between masters and outstations.

2.2 Previous Security Research on DNP3

Although some security enhanced the DNP3 protocol, such as DNPSec[7] and DNP3 secure authentication[8] have been developed, most DNP3 utilities lack identity authentication, data encryption and access control. In the previous research[9, 10], the DNP3 protocol used with TCP/IP used either Transport Layer Security (TLS) or Internet Protocol Security (IPSec) as security. But they were limited in their effectiveness for DNP3 because they were not designed for the DNP3 protocol, but rather for more general applications. For example, if the masters, i.e., the computers in the control centers, are already compromised, then TLS authentication is useless as it cannot prevent the outstations, i.e., the remote computers from receiving illegitimate control commands.

Authorization specification of utilities[11] was proposed on the DNP3 protocol. It defines authorization tables between each connection of master and outstation using function code and data object type. It is good to specify allowed functions and data objects of utilities at the protocol, but it cannot recognize malicious data transfer of compromised utilities.



Time	Source	Destination	p.len
10.233639	.91	.51	64
10.401417	.51	.91	60
19.857511	.91	.51	74
19.878849	.51	.91	60
19.911025	.51	.91	64
20.024781	.91	.51	60
20.041609	.51	.91	346
20.064580	.91	.51	64
20.081008	.51	.91	60
20.114665	.51	.91	198
20.148311	.91	.51	60
20.269013	.91	.51	64
20.282633	.51	.91	60
30.082961	.91	.51	74
30.159265	.51	.91	64
30.274986	.91	.51	60
30.301549	.51	.91	346
30.405021	.91	.51	60
30.490699	.91	.51	64
30.538085	.51	.91	198
30.655014	.91	.51	60
30.695750	.91	.51	64
30.768004	.51	.91	60
39.909488	.91	.51	74
39.974322	.51	.91	64

+ Packet size from _._._.91 to _._._.51
100
0
-100
-200
-300
-400
time
− Packet size from _._._.51 to _._._.91

Figure 3. Traffic between a master and an outstation

Modbus/DNP3 state-based analysis[12] introduced simulation based approach. It specifies state transitions of target SCADA system and simulates from real-time network traffic monitoring. If the simulation result appears that the SCADA system is in dangerous state, alerts are announced. This analysis makes possible to monitor current status of target SCADA system, but the specification extraction and the real-time simulation are hard to apply to a real SCADA system.

Digital Bond's Quickdraw SCADA IDS [13] provides attack signatures for known vulnerabilities of control system such protocols as DNP3, Modbus, and ICCP. They provide these signatures as a form of Snort rule which is a free network IDS system rule. The rules are well adopted to detect attacks using known vulnerabilities, but writing proper specification could be a time-consuming and error-prone process as vulnerabilities are increasing.

3 A Burst-based Whitelist Model

Having analyzed the network traffic of real-world SCADA systems, we propose a whitelist model for master-outstation pairs using the DNP3 protocol.

3.1 Burst-based Characteristic

When a transaction such as status monitoring using a `read-response` is executed on the DNP3 protocol, the transaction consists of several packets including 'data packets' and 'acknowledge packets'. Each packet's size is determined by the DNP3 objects and the frame fragmentation rule of the DNP3 protocol. In general, the time interval between two transactions is greater than the time interval between two packets in a transaction. Therefore, the packets caused by one transaction constitute a burst.

Figure 3 shows a part of the network packets between a master and an outstation. A burst in the dotted box is caused by a transaction. The two transaction are the same operations between a master and an outstation, and that the bursts show the same characteristics based on the packet size.

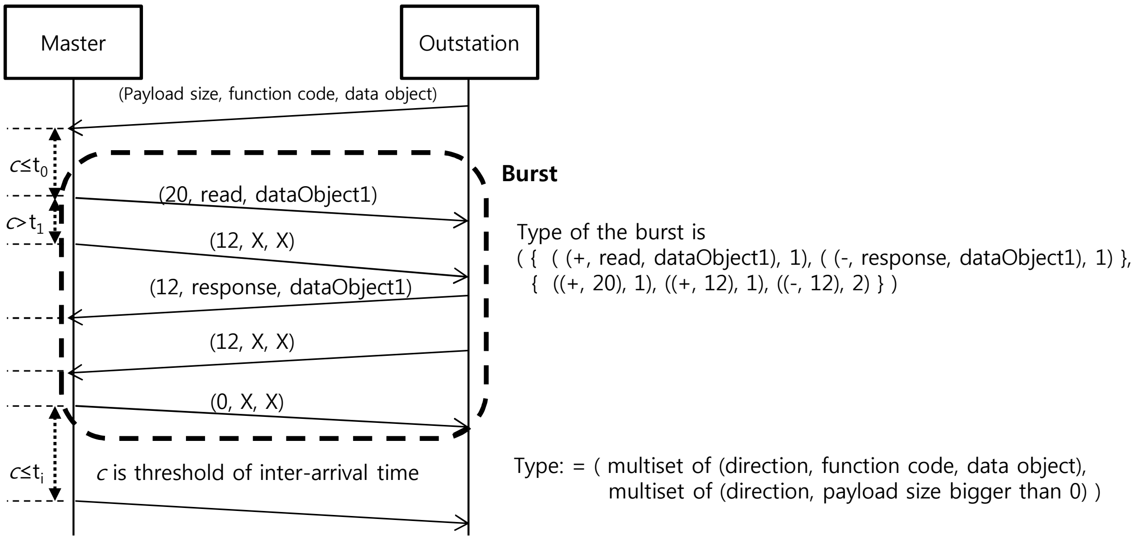

Master
Outstation
(Payload size, function code, data object)
c≤t0
c>t1
(20, read, dataObject1)
(12, X, X)
(12, response, dataObject1)
(12, X, X)
(0, X, X)
Burst
Type of the burst is
({ ((+, read, dataObject1), 1), ((-, response, dataObject1), 1) },
{ ((+, 20), 1), ((+, 12), 1), ((-, 12), 2) })
c≤ti
c is threshold of inter-arrival time
Type: = (multiset of (direction, function code, data object),
multiset of (direction, payload size bigger than 0))

Figure 4. An example of a burst and its type

We can extract several characteristics of a burst. In this paper, we define the type of burst as a 2-tuple (multiset[2] of (direction, function code, data object)) and (multiset of (direction, payload size bigger than 0)). The direction is + if a packet is sent from a master to an outstation, and the direction is - otherwise. Figure 4 shows an example of a burst and its type.

(direction, function code, data object) is an authorized function between a master and an outstation. The multiset of (direction, function code, data object) shows the executed functions in a transaction. The multiset of (direction, payload size bigger than 0) shows the packet fragmentation of a transaction. We do not consider packets without a payload because packets without a payload are generated by the TCP/IP protocol rather than the DNP3 protocol. We use multiset of the information instead of a sequence because the time order of packets may often be changed by trivial causes, such as the switch port mirroring condition.

3.2 Whitelist Model

From the results of the analysis of the collected network traffic, we defined our whitelist mode for the master-outstations, as shown in Figure 5. Each rule represents the allowed types of bursts between a master and an outstation.

In this paper, a `Master` is distinguished by IP and service port, while an `Outstation` is distinguished by IP. `BurstTypes` is a set of `BurstType`. `BurstType` means type of a burst that is used between a master and an outstation. A `Burst` consists of `MControls` and `MPayload1`. `MControls` is the multiset of (direction, function code, data object) pairs that are included in the target burst. `MPayload1` is the multiset of (direction, payload sizes bigger than zero). `ThresholdTime` is the threshold of the inter-packet arrival time.

[2] The number of times an element belongs to the multiset is the multiplicity of that member. For example, a set {a, a, b, b, b, c} is represented by {(a, 2), (b, 3), (c, 1)} in the multiset.

```
         Whitelist  :=  Set of Rule
              Rule  :=  (Master, Outstation, Bursts)
            Master  :=  (IP, Port)
        Outstation  :=  IP
        BurstTypes  :=  Set of BurstType
         BurstType  :=  (MControls, MPayload1)
         MControls  :=  Multiset of Control
         MPayload1  :=  Multiset of (Direction, pay1)
           Control  :=  (Direction, function_code, data_object)
         Direction  :=  + | -
 ThresholdInterval  :=  time
```

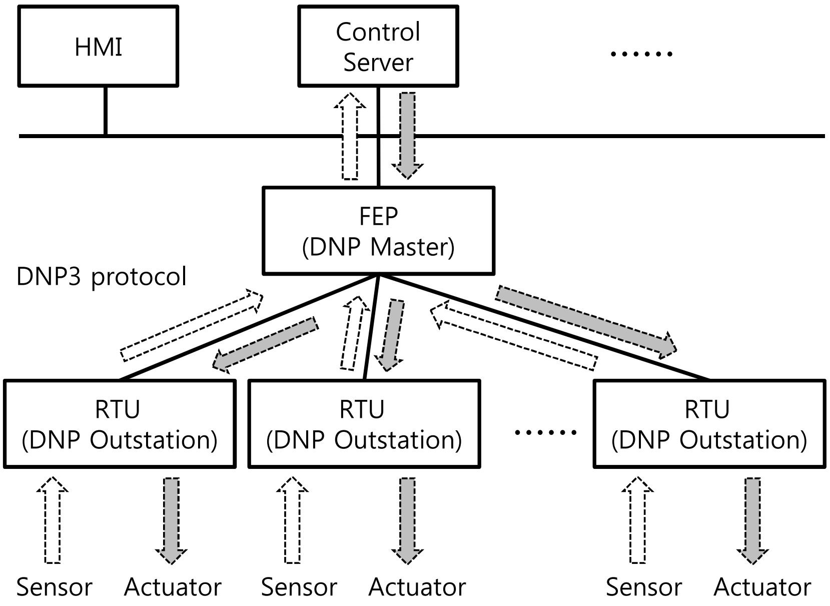


Figure 6. Target SCADA system using the DNP3 protocol

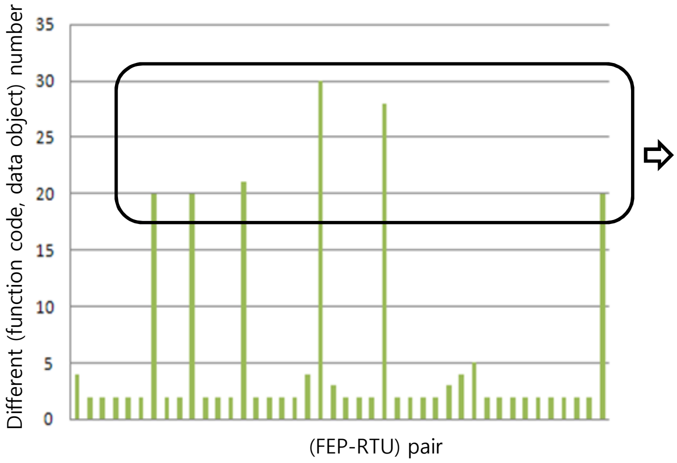
4 Experiment based on Cyber Attacks

Using on the collected network traffic, we introduce how to detect two attack models which can occur in SCADA system and which cannot be detected using the previous research.

4.1 Applying Our Whitelist Model to Real Network Traffic

Above mentioned, we collected DNP3 network traffic in a real-world SCADA system for ten days. There are 42 master-outstation pairs. which includes one master and 42 outstations using the DNP3 protocol. The SCADA system follows the topology of Figure 6. Usually, the master periodically sends the *read* function code to all outstations to monitor current status data, and each outstation sends its specific data objects[3] using the *response* function code.

[3] In our collected network traffic, an outstation sends one to three data objects.



RTU	Total	*Used more than five*
RTU # 7	20	10
RTU # 10	20	10
RTU # 14	21	9
RTU # 20	30	11
RTU # 25	28	11
RTU # 42	20	9

Figure 7. (function code, data object) pairs used in master-outstations during ten days

We counted the different (function code, data object) that each pair of a master and an outstation uses during ten days in Figure 7. Most master-outstations used less than five different (function code, data object).

Six pairs of master-outstations more than 20 different (function code, data object). But, only half of them are used more than five times during the ten days. Although a specific (function code, data object) is allowed between a master and an outstation, the (function code, data object) has to be monitored carefully if it is used for special situation.

In our experiment we chose *0.2 seconds* for the `ThresholdTime` of the collected network traffic. In applying this whitelist model to the network traffic collected in seven days, a pair of a master and an outstation used eleven different types of bursts and four different (direction, function code, data object) on average. Each master-outstation pair has its own types of bursts[4].

To check the coverage the extracted rules, we extracted whitelist rules from last three days of network traffic. The extracted whitelist rules from seven days of network traffic include all the whitelist rules extracted from last three days of network traffic.

4.2 Abnormal Control/Data Transfer

Burst that pretend to be normal mutation bursts in allowed connection give rise to new attack model. As an example of this attack, Stuxnet infects any computers and networks, although it does not damage any computers and networks that are not its target. Previous researches cannot detect malware propagation if a malware is attached to allowed data object communication and its signature is not ready.

When a malware is propagated through network, a malware causes abnormal type of bursts. If a malware is attached to normal data, it causes bigger bursts than usual. Therefore, type of bursts can detect abnormal data transfer.

Cyber-attack using buffer overflow and vulnerable commands is also a kind of abnormal data transfer. In our experiment, we checked how to detect 16 DNP3 vulnerabilities attacks of Table 1 provided by Digital Bond [13]. Our extracted whitelist rules detected all the

[4] We expect that types of bursts may be used for authentication of master-outstations.

SID	DNP3 Attack
1111201/11112011	Disable Unsolicited Responses
1111202/11112021	Non-DNP3 Communication on a DNP3 Port
1111203	Unsolicited Response Storm
1111204/11112041	Cold Restart from Authorized Client
1111205/11112051	Cold Restart from Unauthorized Client
1111206/11112061	Unauthorized Read Request to a PLC
1111207	Unauthorized Write Request to a PLC
1111208	Unauthorized Miscellaneous Request to a PLC
1111209/11112091	Stop Application
1111210/11112101	Warm Restart
1111211	Broadcast Request from an Authorized Client
1111212	Broadcast Request from an Unauthorized Client
1111213	Points List Scan
1111214	Function Code Scan
11112151	Time Change Attempt
11112161	Failed Checksum Error

Table 1. DNP3 vulnerabilities attacks

attacks because the attack traffics make bursts that their types are not included in the whitelist rules.

4.3 Traffic Flooding Attack

'Traffic flooding attack' refers to the sending of too numerous packets or requests for a victim to handle. It is the most typical DoS attack but threatening attack. In the SCADA system, utilities having restrict resource such as PLCs (Programmable Logic Controller) are weak against this attack.

This attack causes that many packets are arrived in a short time. The dense packet interval time causes bigger bursts than usual. Therefore, the burst used to cause a traffic flooding attack are distinguished by their similar arrival time and the biased direction.

In our experiment, we checked two kinds of traffic flooding attack. One is to try many meaningless operations; e.g., SYN flooding attack. We generated the attack traffic by an attack simulation system[5], and this attack is detected by our extracted whitelist rules.

The other is to try many allowed operations such as monitoring data requests. To simulate this attack, we modified packet interval times of the collected network traffic shorter than usual. We used the modified traffic to simulate a traffic flooding attack using allowed operations. Previous work based on authorization[11] cannot detect this attack because all the function codes and data objects are authorized. But, in the viewpoint of bursts, a burst includes several transactions due to shorter packet interval time than threshold time. Therefore, the attack traffic was detected by our whitelist rules.

[5] `http://www.breakingpointsystems.com/`

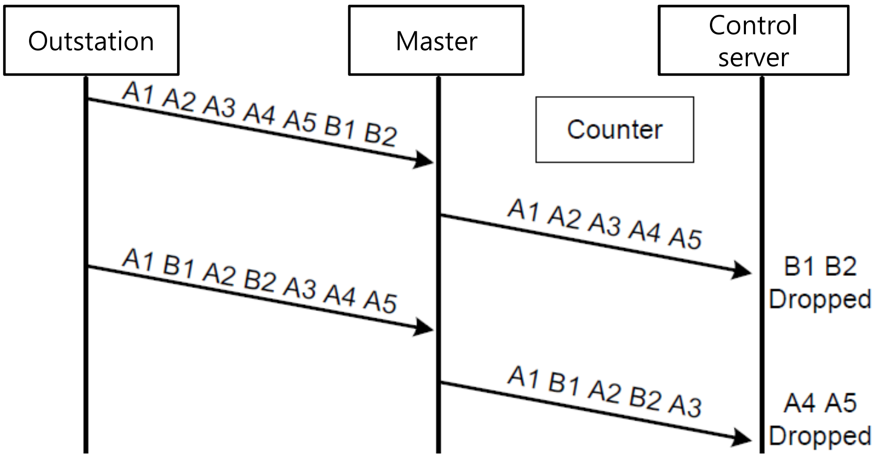


Figure 8. Message drop caused by an event overflow attack[14]

5 Discussion

In this section, we discuss the possibility to detect other cyber-attacks which are not considered in our experiment.

5.1 Man-in-the-Middle Attack

A 'Man-in-the-Middle (MITM) attack' is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection, when in fact the entire conversation is being controlled by the attacker[6]. The MITM attack is not only used to steal or modify information, but it is also used to hide affected system's behaviors.

A MITM attack slows down packet delivery time due to the relay of attacker between victims. Delayed packet arrival time makes unusual type of bursts, such as consisting of fewer packets compared with the bursts of the whitelist rules.

In SCADA system, each utility communicates data within a strict time constraint such as timeout of the DNP3 protocol. Previous work does not reflect the time interval aspect. but our whitelist model strictly specifies the interval time limit of packets so as to divide bursts. Thus, our whitelist model, which considers the time interval, can detect MITM attacks. In addition, this detection ability can be used for performance monitoring in the viewpoint of request-response timing constraint.

5.2 Event Overflow Attack

In this paper, we only consider specific types or values, and not their sequence. Perhaps considering these sequences could provide a clue to detecting abnormal behaviors in control systems. One of the vulnerabilities with consideration of the sequence of message types is the event buffer flooding attack [14].

[6] `http://en.wikipedia.org/wiki/Man-in-the-middle_attack`

Most commercial DNP3 masters simply store all received data in the event buffer. Every new event occupies a new buffer space; if the buffer is full, then the event is discarded. The attack tricks a compromised DNP3 outstation (or an attack on the network successfully pretending to be a DNP3 outstation) into sending so many unsolicited events that the buffer is filled, and events from the uncompromised outstations are lost. To detect this type of attack, we have to check the sequence of messages types - which are received and sent not only by message types. Also, consideration of the other feature's sequence can provide a clue for detecting other attacks.

6 Conclusion

In SCADA system, a utility repeatedly performs its own limited operations. One operation generates a burst, which is a group of consecutive packets with shorter inter-packet gaps than packets arriving before or after the burst of packets. So, a burst can reflect the operation's characteristic.

In this paper, we have proposed a burst-based whitelist model for DNP3 network traffic between a master and an outstation. Our burst-based approach can represent the characteristics of application-level operations and inter-packet arrival time. To confirm the validity of our whitelist model, we extracted the whitelist rules of 42 working master-outstation pairs and analyzed how the rules can be used to detect cyber-attacks in a SCADA system. In addition, we discuss the possibility of providing additional features based on the burst-based approach.

References

[1] Yun, J., Jeon, S., Kim, K., and Kim, W.: A Burst-based Whitelist Model for DNP3 Communication in the SCADA system. The 7th International Conference on Information Security and Assurance (ISA 2013). Cebu, Philippines, (2013)

[2] Daneels, A. and Salter, W.: What is SCADA. International Conference on Accelerator and Large Experimental Physics Control Systems, pp. 339–343, (1999)

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

[4] Barbosa, R.R.R., Sadre, R., and Pras, A.: Difficulties in Modeling SCADA Traffic: A Comparative Analysis: 13th International Conference Passive and Active Measurement (PAM). In: Lecture Notes in Computer Science, vol. 7192, Springer, Verlag (2012)

[5] Shakkottai, S. and Brownlee, N. and Claffy, KC: A study of burstiness in tcp flows. Passive and Active Network Measurement, pp. 13–26, Springer, (2005)

[6] Clarke, G. and Reynders, D.: Practical modern SCADA protocols: DNP3, 60870.5 and related systems. Newnes, (2004)

[7] Majdalawieh, M., Parisi-Presicce, F., and Wijesekera, D.: DNPSec, Distributed Network Protocol Version 3 (DNP3) Security Framework: Advances in Computer, Information, and Systems Sciences, and Engineering, pp. 227–234 (2006)

[8] DNP Users Group: dnp3 specification, secure authentication, supplement to volumn 2: `http://www.dnp.org/Modules/Library/Document.aspx`

[9] Dierks, T.: The transport layer security (TLS) protocol version 1.2, (2008)

[10] Doraswamy, N. and Harkins, D.: IPSec: the new security standard for the Internet, intranets, and virtual private networks, Prentice Hall, (2003)

[11] Mander, T., Cheung, R., and Nabhani, F.: Power system DNP3 data object security using data sets. Computers and Security. vol. 29, no. 4, pp. 487–500. Springer, (2010)

[12] Fovino, I.N. and Carcano, A. and De Lacheze Murel, T. and Trombetta, A. and Masera, M. Modbus/DNP3 state-based intrusion detection system. Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on, pp. 729–736, IEEE, (2010)

[13] Quickdraw SCADA IDS, `http://www.digitalbond.com/tools/quickdraw/`

[14] Jin, D., Nicol, D.M., Yan, G.: An event buffer flooding attack in DNP3 controlled SCADA systems. In: Proceedings of the 2011 Winter Simulation Conference (WSC), pp. 2614-2626. IEEE Press, (2011)