논문 18-43-07-03 The Journal of Korean Institute of Communications and Information Sciences '18-07 Vol.43 No.07
https://doi.org/10.7840/kics.2018.43.7.1084

Anomaly Detection Based on Burst Characteristics for DNP3

Gi-ung Ha◆, Dae-woon Lim*, Min-Ho Jang**, Ji-Woong Jang°

ABSTRACT

The SCADA (supervisory control and data acquisition) system has many existing security vulnerabilities because the systems are connected on network-based communications. Whereas conventional attacks concentrate on the server or master in the internet environment, direct attacks to outstations or slaves may cause significant damage in the SCADA system. If an attacker has a good knowledge of the control protocols of the SCADA system, it could attack an outstation disguised as a master. In this situation, the rule-based intrusion detection system might not be able to classify the malicious control message as intrusion because the message appears as a normal message. In this paper, an intrusion detection model based on the burst characteristics of the SCADA system with DNP3 (distributed network protocol) is proposed for outstations. Using the challenge-response authentication of the DNP3 protocol, the proposed model automatically updates a white list used to determine the control message.

Key Words : Anomaly Detection, Burst-based, DNP3, Intrusion Detection, SCADA, White List

Ⅰ. Introduction

The SCADA (supervisory control and data acquisition) system refers to the industrial control system operated in many cases on national infrastructures. The conversion of IT (information technology) and electrical grid has conceived the smart grid, which rapidly changed the electric grid environments from closed communication to open networks[1]. Accordingly, the open network-based SCADA systems have many existing security vulnerabilities, and critical attacks have been implemented against national infrastructures in many countries. In particular, if the attacker has good knowledge of the control protocols of the SCADA system, it can create confusion by counterfeiting the information from the control message without changing the message structure, which results in malfunctions of the SCADA system[2,3]. The SCADA system should be satisfied with confidentiality, integrity, availability, and non-repudiation. Fig. 1 shows the electrical system information security threats and countermeasures defined in IEC/TS 62351, which is the international standard of information security for the control and operation of electrical systems.

The IDS (intrusion detection system) has become an important research field of SCADA system information security[4]. It is well known that anomaly detection-based IDS has the advantage of detecting unknown attack types. Because the SCADA system communication shows regularity characteristics in the operation mode where the control messages are transmitted periodically, a white list-based IDS is suitable for the SCADA system[5].

※ This work was supported by the 2017 Research Fund of Ulsan College.
◆ First Author : Department of Information Security Business, KEPCO KDN, hkw342@naver.com, 정회원
° Corresponding Author : (0000-0002-0023-0733) School of Computer Information Technology, Ulsan College, uc.stasera@gmail.com, 종신회원
* Department of Information Communication Engineering, Dongguk University, daewoonlim@gmail.com, 종신회원
** (0000-0001-9195-5184) School of Electrical and Electronic Engineering, Ulsan College, mhjang@uc.ac.kr, 종신회원
논문번호 : 201806-B-156-RN, Received April 12, 2018; Revised July 4, 2018; Accepted July 20, 2018

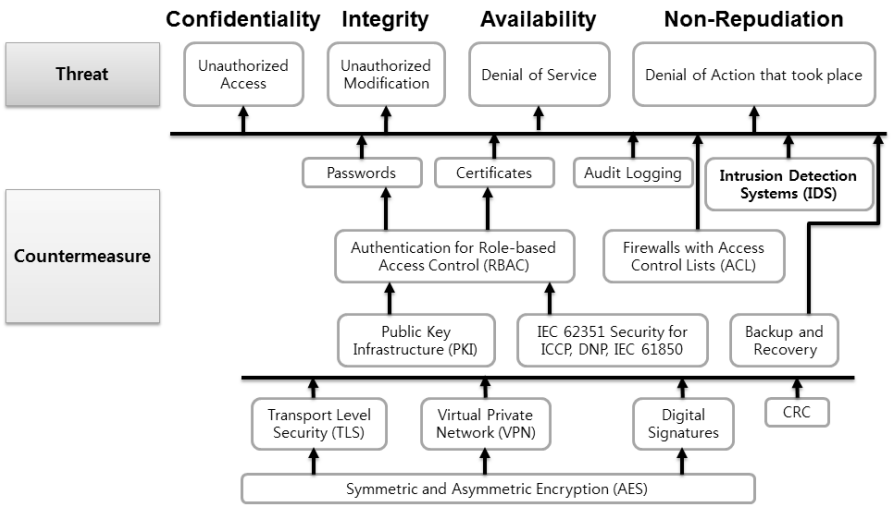

Fig. 1. Electric System Information Security Threat and Countermeasure

This paper proposes a white list-based anomaly detection scheme that exploits the burst characteristics of DNP3 (distributed network protocol) that is the IEEE 1815-2012 standard for electric power systems communication, and one of the most widely used control protocols for the SCADA system. The rest of this paper is organized as follows. After an overview of the DNP3 protocol and its secure authentication are presented in section II, the conventional intrusion detection schemes are investigated in section III. In section IV, a white list-based anomaly detection scheme is proposed for DNP3 outstations and an automatic white list update algorithm is explained. Finally, conclusion remarks are given in section V.

Ⅱ. DNP3 protocol

2.1 DNP3 overview

SCADA systems are composed of central stations and field devices called masters and outstations. DNP3 is designed for the masters to send control comments to the outstations and gather information from the outstations. DNP3 is developed as an IEEE standard primarily to ensure interoperability between national infrastructures, such as electric power, water, and gas supply[6].

Fig. 2 shows the structure of a DNP3 protocol stack. DNP3 consists of the application layer, transport function, and data link layer over serial and TCP/IP. Namely, DNP3 acts as an application layer in a TCP/IP-based environment[7]. Using the application layers services, DNP3 devices communicate with each other by sending an application message that consists of fragments. The number of fragments in a message depends on the message length. Fig. 3 shows that a fragment is composed of the application header and the plurality of pairs of object header and objects[7].

One octet function code identifies the purpose of the application layer message, such as read, write, operate, freeze, authenticate, and so on. A total of

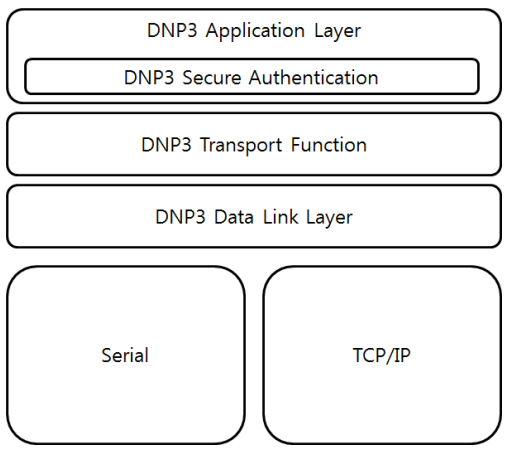

Fig. 2. DNP3 Stack Structure

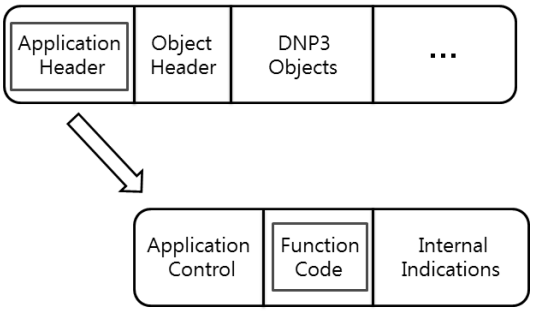

Fig. 3. Fragment Application Header Structure

37 function codes are defined in the latest standard. Fig. 4 shows the object header of the fragment[7].

When the application header alone cannot re ect the purpose of the message perfectly, object headers and objects are supported to provide supplementary information for the completion of the application message. The object header has object type field, qualifier field and range field. The object type field consists of a group octet and a variation octet. The object group specifies the data type or values in a master request or outstation response. The object variation species the data format. Thus, the structure of an object and the type of data are identified uniquely by the object group and variation.

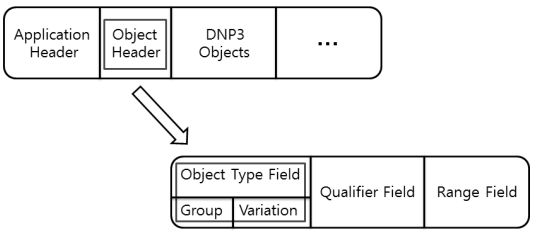

Fig. 4. Fragment Object Header Structure

2.2 DNP3 secure authentication

The DNP3 standard provides a secure authentication protocol mechanism at the application layer to ensure that an outstation communicates with an authorized master and vice versa. The DNP3 secure authentication protocol mechanism provides a challenge-response mode and an aggressive mode. Fig. 5 shows a successful example of the challenge-response mode[7].

DNP3 ASDU (application service data unit) is

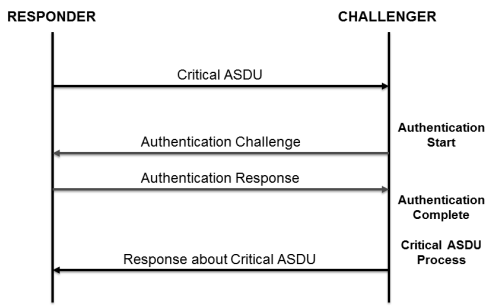

Fig. 5. Example of Successful Challenge

classified into two categories, noncritical and critical, according to the function code in the fragment. For example, the read function is defined as non-critical and the write function as critical. A challenger is the one that wants to certificate the other party, and a responder is the one that sends a critical ASDU. A challenger starts an authentication procedure by sending the authentication challenge message only when it receives a critical ASDU. If a responder receives an authentication challenge message, it replies with an authentication response that contains the HMAC value calculated with the authentication challenge message and the challenged critical ASDU. Finally, the challenger authenticates the responder if the received HMAC value from the responder is identical to the value generated by the challenger. After finishing the authentication procedure, the challenger sends a reply message to the responder for the critical ASDU.

Fig. 6 shows a successful example of the aggressive mode. In the aggressive mode, it is assumed that the responder receives a virtual challenge from the challenger, although the challenger does not actually send a challenge. Then, the responder sends a critical ASDU that contains

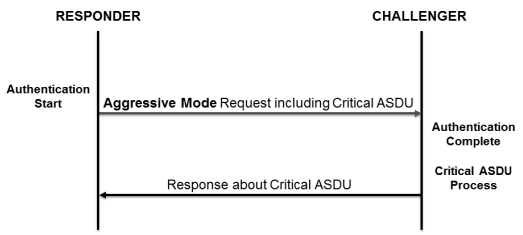

Fig. 6. Example of Successful Aggressive Mode

the HMAC value. By omitting the challenge message and sending a critical ASDU with HMAC, the aggressive mode reduces communication overhead compared with the challenge response mode.

There are three types of symmetric keys in DNP3: session, update, and authority. The session key is used to generate the HMAC value in the authentication procedure and it is changed periodically. The master generates a session key and encrypts it using an update key to send the encrypted session key to the outstation. The authority key is used in order to change the update key[7].

2.3 Study of conventional DNP3 intrusion detection scheme

In this paper, we analyze separately the conventional research on intrusion detection scheme for DNP3 through three types: attack signature-based, authorization-based, and burst-based.

The attack signature-based intrusion detection scheme was developed by a certain security professional company of industrial control systems in the United States. It is a widely used commercial technique in the SCADA system. The attack signature-based scheme works with rule-based intrusion detection systems by considering the characteristics of the industrial control protocol, and characteristically consists of two parts: the SCADA preprocessor and SCADA detection engine. The preprocessor is composed of ve plug-ins that decode the packets and process the evaluation for allowing the decoded field; the detection engine identifies a situation that has a probability of attacks. There are 16 patterns used in the SCADA detection engine related with DNP3[8].

The authorization-based intrusion detection scheme, proposed by T. Mander, is a technique designed based on the available actions of the DNP3 data link layer and the DNP3 application layer. The authorization-based scheme can have three cyber security states: Idle, Frame security, and Data security. The Idle state is the state where there is no security processing because there is no current data transfer; the Frame security state is the state where the security procedure is processed at the data link layer. Lastly, the Data security state is the state where the security procedure is processed at the application layer. Each state is changed flexibly based on the conditions that correspond to the situation and the security procedure. The security procedure checks for authorization based on the available data fields in the message about the user and the environment[9].

The burst-based intrusion detection scheme, proposed by J. Yun, is an anomaly detection method that introduces the burst concepts used in existing communication areas to detect intrusion. Burst in

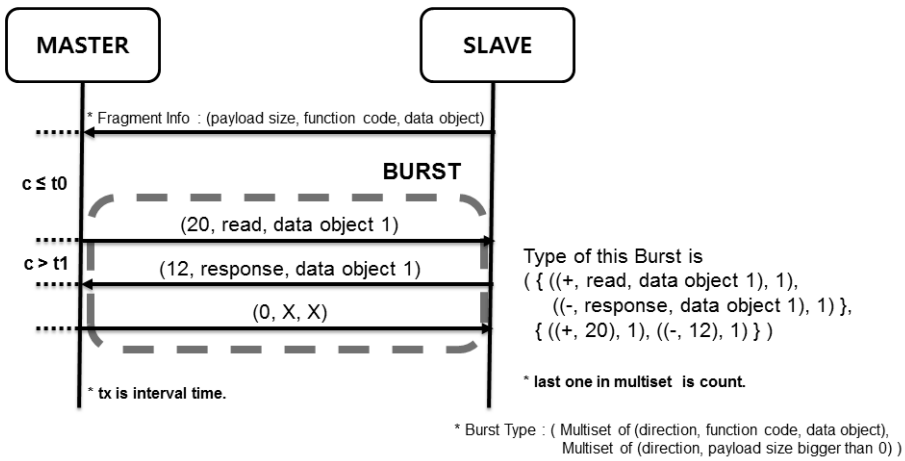

Fig. 7. Example of Burst Type

Table 1. White List Model Based on Burst Type (where ':=' means 'is' or 'is defined')

White List	:=	Set of Rule
Rule	:=	(Master, Slave, Burst Types)
Master	:=	(IP, Port)
Slave	:=	(IP)
Burst Types	:=	Set of Burst Type
Burst Type	:=	(MControls, MPayloads)
MControls	:=	Set of Control
MPayloads	:=	Set of (Direction, Payload)
Control	:=	(Direction, Function Code, Data Object)
Direction	:=	+ | -
Threshold Interval	:=	Time

DNP3 means a set of fragments received at time intervals shorter than any threshold c. The burst-based approach defines burst in two forms: one that represents the direction, function code, and data object, and another that represents the direction and payload size greater than zero. Fig 7 shows an example of burst type[10].

In Fig. 7, a burst can be represented in two different formats, which together are called a burst type. Here, c is the threshold used as the baseline to classify the burst. The upper burst type represents the direction, function code, and data object, and the bottom represents the direction and payload size greater than zero. In the burst-based approach, the upper burst type is called Control, whereas the bottom is called a Payload. Each burst has a burst type composed of several controls and payloads. Table 1 indicates the white list model expressed based on the burst type described above [10].

The burst-based intrusion detection scheme was used to perform a cyber attack detection experiment based on the white list model. Actual traffic was collected for ten days, and a model was made based on that data. As a result of the analysis, the environment uses 11 burst types and four Control types; the criterion of the threshold value for the burst configuration is set to 0.2 sec. After making the model, J. Yun conducted experiments on abnormal control commands, attack of data transfer, and traffic flooding attack, and confirmed that all these attacks can be detected effectively[10].

Ⅲ. Comparison and limitations of conventional intrusion detection schemes

3.1 Comparison of conventional DNP3 intrusion detection scheme

Table 2 lists a comparison of the conventional intrusion detection scheme for DNP3 with respect to the cyber attack in the burst-based intrusion detection approach. As can be seen from Table 2, the intrusion detection method based on the attack signature can detect control command attacks using DNP3 vulnerability, but it has a limitation in detecting traffic flooding and data transfer attacks. The detection method based on the attack signature is vulnerable to other types of attacks because the pattern is configured based on DNP3 vulnerability. The next scheme is the intrusion detection scheme based on authority. This scheme is similar to the

Table 2. Comparison of Conventional Intrusion Detection Scheme Based on Cyber Attacks

Cyber Attack	Attack Signature Approach	Authorization Approach	Burst Approach
Data Transfer	Undetectable	Undetectable	Detectable
Control using DNP3 Vulnerability	Detectable	Detectable	Detectable
Traffic Flooding	Part Detectable	Part Detectable	Detectable

Table 3. Comparison of Conventional Intrusion Detection Scheme Based on Attack Types

Attack Type	Attack Signature Approach	Authorization Approach	Burst Approach
Attack in Signature	Detectable	Detectable	Detectable
Attack Not in Signature	Undetectable	Detectable	Detectable
Attack using Allowed Control	Undetectable	Undetectable	Detectable

scheme based on the attack signature. Because it is configured based on user authority, it has difficulties detecting attacks configured with permitted content. The last scheme is the intrusion detection scheme based on burst. This scheme can detect not only control command attacks using DNP3 vulnerabilities, but also all attacks, including traffic flooding and data transfer attacks. The burst-based scheme can effectively detect various attacks because the attacks generate bursts not in the white list.

After sorting and classifying the cyber attacks in the burst-based intrusion detection according to attack types, a comparison based on attack type is demonstrated in Table 3. As in Table 2, the intrusion detection scheme based on attack signature can detect attacks that exist in the pattern, but it has a limitation in detecting attacks not in the pattern. The intrusion detection scheme based on authority can detect attacks regardless of whether they are found in the pattern, but it is vulnerable to attacks that use authorized control commands. The last scheme is the burst-based intrusion detection scheme. This scheme can detect all three types of attacks because such attacks generate a burst not in the white list.

3.2 Limit of conventional Burst-based intrusion detection scheme

The conventional burst-based intrusion detection scheme can detect several cyber attacks compared with other intrusion detection schemes, but it has limitations in detecting attacks using allowed bursts. In other words, this method cannot detect those attacks that generate a type of burst present in the white list. Fig. 8 shows a scenario of an attack that uses an allowed burst.

As shown in Fig. 8, if it is possible for an attacker to generate a Burst <1'> with the same shape as Burst <1>; consequently, the attacker can attempt to attack without violating the white list. Therefore, by sending a burst with the attack commands that the attacker wants to transmit, it is possible to transmit the same attack commands that the attacker wants to send. Using the threshold reference of the burst configuration, attackers can obtain information on the burst type used in the target system. Using such information, attackers can cause a malfunction in the target system by sending

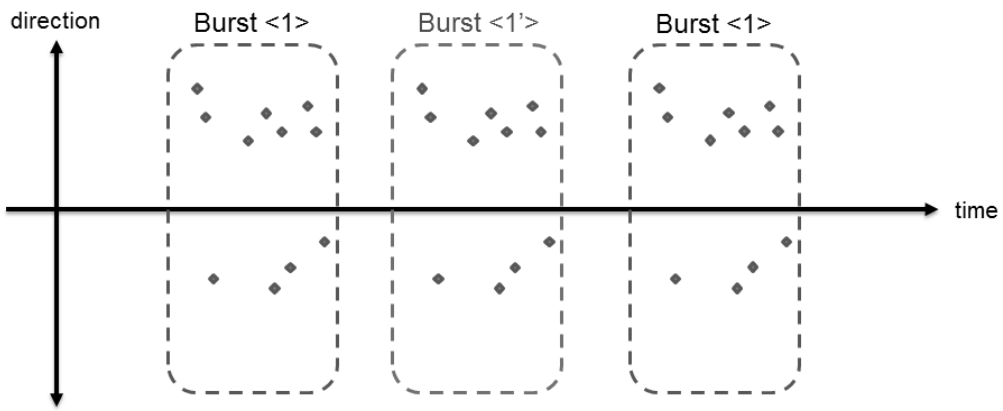

Fig. 8. Attack Using Allowed Burst

a burst that contains attack commands.

Ⅳ. Intrusion detection scheme for DNP3

In this paper, the slave specialized intrusion detection scheme is constructed based on the improved burst-based intrusion detection scheme. The authentication and white list automation used in the intrusion detection scheme is applied to the proposed system.

4.1 Improved Burst-based intrusion detection scheme

As described in the previous section, the existing burst-based intrusion detection approach is vulnerable to attacks that use allowed bursts. Therefore, in this paper, we propose a method for detecting such attacks by adding the concept of interval burst information to the conventional scheme. Fig. 9 shows the intrusion detection scenario using the burst interval information.

In Fig 9, the attacker attempts to attack by generating a Burst <1'> in the same form of Burst <1>. The conventional scheme cannot detect such types of attacks, but the scheme that uses the interval information can do so. In Fig 9, the interval information for Burst <1> is 10 sec. If the attacker attempts to attack using Burst <1'>, the interval information is changed to 2 sec. Through this, we can detect abnormal symptoms in the system. Therefore, by adding burst interval information to the conventional white list, we can detect the attacks using allowed bursts.

The improved burst-based white list model that

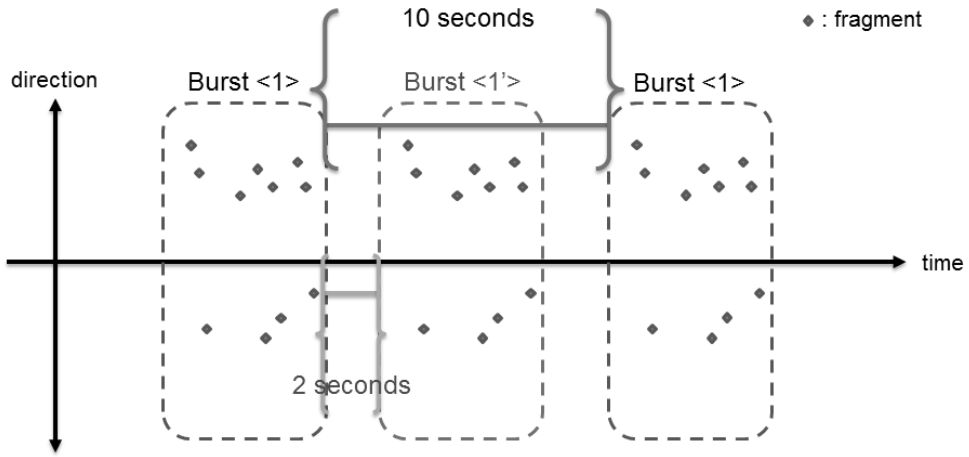

Fig. 9. Detection Using the Burst Interval

Table 4. Burst White List Model Incude Burst Interval (where ':=' means 'is' or 'is defined')

White List	:=	Set of Rule
Rule	:=	(Master, Slave, Burst Types)
Master	:=	(IP, Port)
Slave	:=	(IP)
Burst Types	:=	Set of Burst Type
Burst Type	:=	(MControls, MPayloads, **Burst Interval**)
Burst Interval	:=	**Time**
MControls	:=	Set of Control
MPayloads	:=	Set of (Direction, Payload)
Control	:=	(Direction, Function Code, Data Object)
Direction	:=	+ | -
Threshold Interval	:=	Time

includes burst interval information is listed in Table 4.

In Table 4, burst interval information is added to the conventional white list model, and the items displayed in bold font correspond to this. The interval information is required for each burst type, and this is expressed by time as a threshold. Furthermore, the master and slave fields can be determined to be used by purpose. In this paper, the slave field can be omitted because the target is constructed as a slave-specific intrusion detection system.

4.2 Intrusion detection scheme that applies authentication

In this section, we propose an improved intrusion detection scheme that applies authentication. The structure of proposed slave intrusion detection system is shown in Fig 10.

In Fig. 10, when a fragment is received, it is stored continuously in the burst buffer. If the burst is set by a threshold value, the burst algorithm is executed through white list rules. If the algorithm is executed successfully, it is treated as normal. Otherwise, it is treated as abnormal, and both the alarm and authentication are processed. Authentication is processed for non-critical ASDU excluded from the execution target of the existing environment.

To apply the non-critical process described in this intrusion detection system, the concept of sub-burst is also required. A single burst can configure the sub-burst when the fragment is added. The conventional burst-based intrusion detection scheme does not execute the intrusion detection algorithm until the burst is configured completely. Therefore, in order to apply authentication, adding the sub-burst concept to the process intrusion detection algorithm is required for the received fragment before the burst is completely configured. If the abnormal fragment of the detection algorithm is non-critical ASDU, the authentication process is executed.

Fig. 11 shows the slave intrusion detection algorithm that applies authentication. The moment that the fragment is received, it is determined to be the fragment of the burst internal if the interval is less than the threshold value. Otherwise, it is determined to be burst external.

At this moment, the total burst algorithm is executed if the fragment is determined to be external. After that, both internal and external bursts are checked for the payload size to be greater than zero, and the sub-burst algorithm is executed by storing the fragment to the current burst buffer.

4.3 Intrusion detection scheme that applies white list automation

The proposed intrusion detection system in this paper applies white list automation. Fig. 12 shows the structure of the slave intrusion detection system that applies white list automation.

In Fig. 12, the moment that the burst is set based

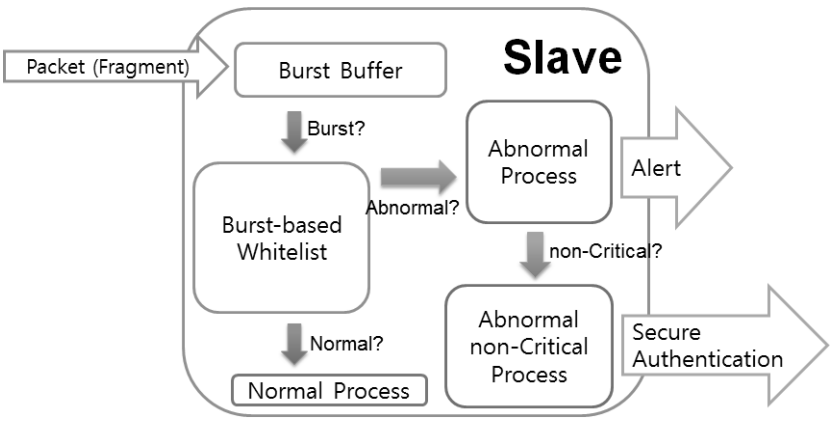

Fig. 10. Structure of Slave Intrusion Detection System that Applies Authentication

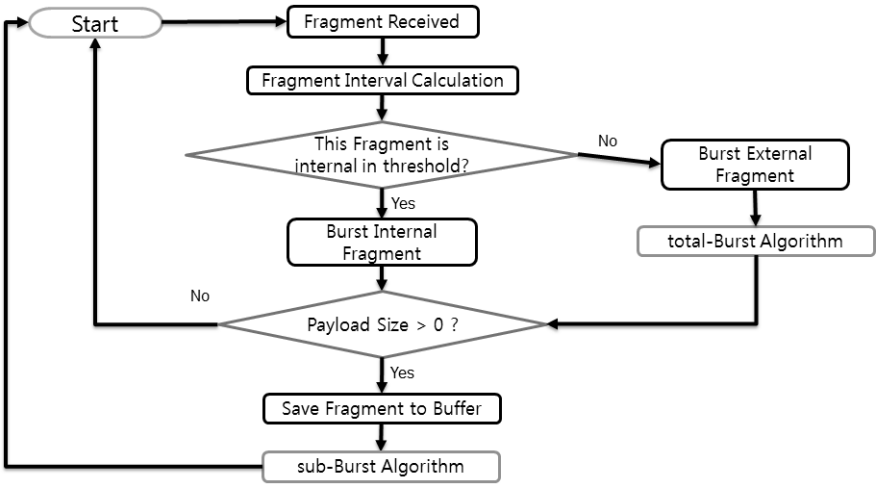

Fig. 11. Algorithm for Slave Intrusion Detection Scheme that Applies Authentication

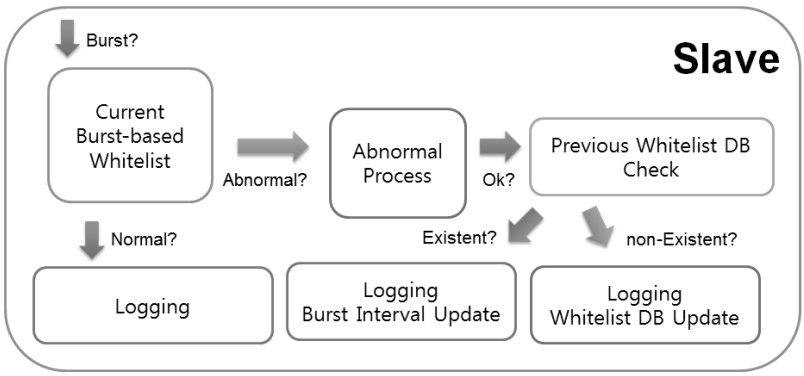

Fig. 12. Structure of Slave Intrusion Detection System that applies White List Automation

on the threshold value, the burst algorithm is executed according to the white list rules. In the case of normal algorithm execution, the system runs logging tasks. Otherwise, the system runs authentication. If authentication is executed successfully, the system changes the state to OK and executes the check process to determine whether the current burst type exists in the white list database. If the burst type does exist in the database, the logging operation and burst interval update are processed. Otherwise, the system performs the logging operation and adds the current burst type to the white list database.

Furthermore, the white list rule is updated according to the corresponding white list update interval. If the current time corresponds to the update cycle, the current white list rule is updated based on the current stored log information, and the stored log information is initialized.

Fig. 13 shows the algorithm for the intrusion detection scheme that applies white list automation.

In Fig. 13, if an interval of the fragment exceeds the threshold, it is determined that the fragment is burst external. Otherwise, the white list rule is updated according to the stored log information if the current time is within the update interval. For the case where the fragment is the burst external, the total-burst algorithm is executed, and then the logging operation for a normal burst is performed. If the authentication is not processed successfully, the alarm process is executed. On the other hand, if authentication is processed successfully, this scheme

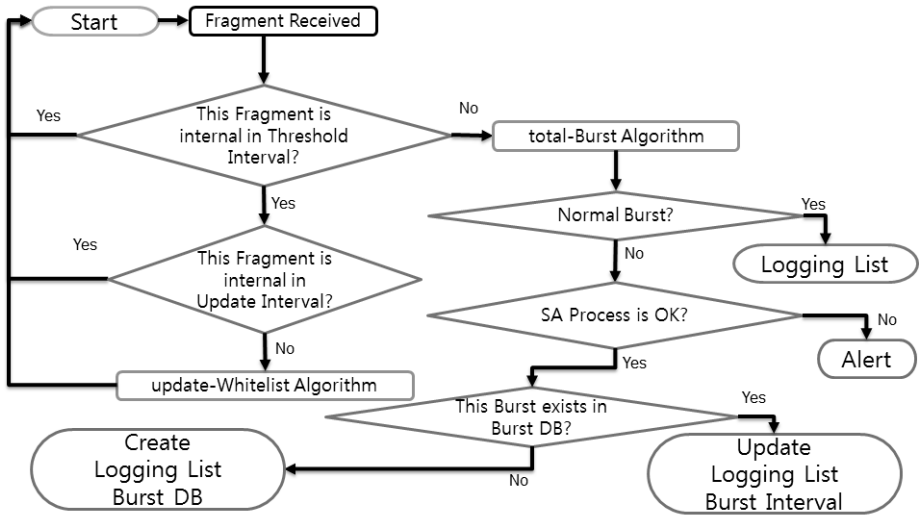

Fig. 13. Algorithm of Intrusion Detection Scheme that applies White List Automation

checks the white list database to determine whether the current configured burst exists in the list. If there is a current configured burst in the database, the scheme performs the logging work and updates the burst interval information. Otherwise, the scheme performs the logging work and adds the configured burst information of the current buffer to database.

4.4 Experiment results and analysis

In this paper, in order to experiment with the proposed intrusion detection system, we configure

Table 5. Results from Proposed Intrusion Detection System Operation

<First Packet>	
Raw Traffic Part	
Time	21:46:50.923
From	Master
To	Slave
Meaning	Application Header, Read Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 7
Raw Data	c7 01 3c 02 06 3c 03 06 3c 04 06
Burst Analysis Part	
Burst Number	2
Interval Burst Time	00:00:05.063
Burst Detection Type	Normal
mControl Burst	((1, 1, <60,2>, <60,3>, <60,4>), 1)
mPayload Burst	((1, 9), 1)
Fragment Detection Type	Normal
Interval Packet Time	00:00:05.062
<Second Packet>	
Raw Traffic Part	
Time	21:44:35.064
From	Master
To	Slave
Meaning	Application Header, Authentication Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 3
Raw Data	c3 20 78 04 07 01 64 00
Burst Analysis Part	
Fragment Detection Type	Normal
Interval Packet Time	00:00:00.063

Table 6. Results for Detection of Attack that uses Controls Not Allowed

<First Packet>	
Raw Traffic Part	
Time	14:45:07.162
From	Master
To	Slave
Meaning	Application Header, Read Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 10
Raw Data	ca 01 32 01 07 01
Burst Analysis Part	
Fragment Detection Type	Abnormal
mControl Fragment	((1, 1, <50,1>), 1)
mPayload Fragmant	((1, 4), 1)
Interval Packet Time	00:00:00.125
<Second Packet>	
Raw Traffic Part	
Time	14:45:07.162
From	Slave
To	Master
Meaning	Outstation Authentication Event, state=IDLE, event=CRITICAL RCVD
	Application Header, Authentication Response, FIR(1) FIN(1) CON(0) UNS(0) SEQ 10
Raw Data	ca 83 00 00 78 01 5b 01 0c 00 04 00 00 00 00 00 04 01 27 99 ae 3b
<Third Packet>	
Raw Traffic Part	
Time	14:45:07.287
From	Master
To	Slave
Meaning	Application Header, Authentication Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 10
Raw Data	ca 20 78 02 5b 01 16 00 04 00 00 00 01 00 e2 2d 1a 3a 7f 49 95 78 c3 a0 ea 87 b7 72 5b 00

the DNP3 communication environment using a commercial DNP3 source code library by Triangle Microworks's products[11]. Master and slave communicate with each other one-to-one in this experiment environment. We construct a specialized intrusion detection system based on the improved burst-based scheme on the slave side. Authentication is applied to this intrusion detection system and white list automation is not applied in this experiment. We used authentication algorithm that recommended by DNP3 Standard.

As a result of configuring the communication experiment environment based on a conventional experiment environment, there are nine types of burst and ten types of control in the experiment environment. The threshold of the burst configuration is set to 0.3 sec. Because the experiment environment includes communication initiation, the number of used controls is larger than in the conventional experiment environment. We set the threshold to 0.1 sec longer than the conventional environment because there is a time delay required for Secure Authentication. In addition, because an interval error in communication occurs when the burst interval information is applied to the intrusion detection scheme, we set an acceptable threshold for the burst interval information to 0.3 sec.

After implementing the proposed intrusion detection system by configuring the experiment environment, the results of operating in normal

Table 7. Results for Detection of Attack that uses Allowed Controls

<First Packet>	
Raw Traffic Part	
Time	14:50:11.775
From	Master
To	Slave
Meaning	Application Header, Read Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 14
Raw Data	cc 01 3c 02 06 3c 03 06 3c 04 06
Burst Analysis Part	
Burst Detection Type	Abnormal (Burst Violation)
mControl Burst	((1, 3, <12,1>), 1) ((1, 32,<120,2>), 1) ((1, 1,<60,2>,<60,3>,<60,4>), 1)
mPayload Burst	((1, 18), 1) ((1, 28), 1) ((1, 9), 1)
<Second Packet>	
Raw Traffic Part	
Time	14:50:46.963
From	Master
To	Slave
Meaning	Application Header, Authentication Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 3
Raw Data	c3 20 78 04 07 01 64 00
Burst Analysis Part	
Burst Detection Type	Abnormal (Burst Violation)
mControl Burst	((1, 3,<12,1>), 1) ((1, 32,<120,2>), 1)
mPayload Burst	((1, 18), 1) ((1, 28), 1)

network traffic are shown in Table 5.

In Table 5, the upper part is the case where the burst external fragment is received, and the lower part is the case where the burst internal fragment is received. The upper part determines that the burst is external because the reception interval of the current fragment is longer than 5 sec. Therefore, it runs the total burst algorithm by targeting the configured buffer, and determines whether the burst is normal or abnormal by executing the results of the algorithm. Then, the buffer external is configured with the current fragment to initial the data by initiating the buffer, and the internal then sets the current fragment to the buffer of the last data. Then, the algorithm runs for the common buffer configured as both internal and external, and the fragment is determined by the results.

The experiment results from the intrusion detection scenario for the attack that uses commands not allowed is shown in Table 6.

The top side of Table 6 is the case where the fragment of the burst internal has been received. The currently received fragment is composed of the Read Function Code, No. 50 Group Number, and No. 1 Variation Number. This is a command to read time information. However, because this command is not used in the experiment environment, it determines the burst rules when processing the algorithm. Because the Function Code is non-critical ASDU, the authentication mechanism is executed after abnormal symptoms are determined during execution of the sub-burst algorithm. The top side of the table corresponds to this case described above.

Table 7 shows the results of the scenario for detecting attacks using allowed commands. Because Select and Operate work collaboratively in the configured experiment environment, the command is always transmitted continuously, and configure the same burst. However, in Table 7, Select operates alone. Therefore, it is determined as an abnormal

Table 8. Result about the Detection for Attack using Allowed Burst

<First Packet>	
Raw Traffic Part	
Time	14:57:08.608
From	Master
To	Slave
Meaning	Application Header, Read Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 11
Raw Data	cb 01 3c 02 06 3c 03 06 3c 04 06
Burst Analysis Part	
Burst Number	2
Interval Burst Time	00:00:03.063
Burst Detection Type	Abnormal (Interval Violation)
mControl Burst	((1, 1, <60,2>,<60,3>, <60,4>), 1)
mPayload Burst	((1, 9), 1)
<Second Packet>	
Raw Traffic Part	
Time	15:24:21.004
From	Master
To	Slave
Meaning	Application Header, Authentication Request, FIR(1) FIN(1) CON(0) UNS(0) SEQ 15
Raw Data	cf 01 3c 02 06 3c 03 06 3c 04 06
Burst Analysis Part	
Burst Number	4
Interval Burst Time	00:00:03.062
Burst Detection Type	Abnormal (Interval Violation)
mControl Burst	((1, 1,<60,2>,<60,3>,<60,4>), 1)((1, 3,<12,1>), 1) ((1, 32,<120,2>), 2)((1, 4,<12,1>), 1)
mPayload Burst	((1, 9), 1) ((1, 16), 2) ((1, 28), 2)

symptom because burst is configured to a different type from the white list rules.

The results for the detection scenario of attacks that use allowed bursts is shown in Table 8.

The top side of Table 8 is the case where a buffer similar to the burst type that corresponds to <2> in the white list rules is set. The bottom side of the table is the case where a buffer similar to the burst type that corresponds to <4> is set. In the experiment environment, <2> is composed of 5-sec burst interval information, and <4> is composed of 20-sec burst interval information. However, the attacker sends <2> and <4> with the same burst interval of 3 sec because the attacker is attempting to attack without considering the burst interval information. Therefore, this determines anomaly, and the system executes the alarm because it is in violation of the white list rules.

The proposed scheme in this paper has a difference in many respects from conventional schemes. Table 9 lists the comparison between the proposed scheme and the conventional scheme from the perspective of the attack type.

In Table 9, the conventional approach is vulnerable to attack using the allowed burst. However, the proposed approach can detect the attack using allowed bursts because it checks for intrusion based on the burst interval information.

Table 10 shows a comparison of the proposed and conventional scheme from the perspective of the behavior.

In Table 10, the conventional approach processes

Table 9. Comparison of Conventional Scheme and Proposed Scheme on Attack Type

Attack Type	Conventional Burst Approach	Proposed Burst Approach
Attack using Not Allowed Control	Detectable	Detectable
Attack using Allowed Control	Detectable	Detectable
Attack using Allowed Burst	Undetectable	Detectable

Table 10. Comparison of Conventional Scheme and Proposed Scheme on Behavior at the time of intrusion

Behavior at The Time of Intrusion	Conventional Burst Approach	Proposed Burst Approach
Alert	Supported	Supported
Secure Authentication	Unsupported	Supported

an alarm only if abnormal symptoms occur. However, the proposed approach can correspond immediately because it performs a sub-burst algorithm when penetration is suspected. Furthermore, the proposed process performs not only an alarm, but also an authentication if the suspected fragment is non-critical ASDU. Therefore, it can provide more complementary security.

Ⅴ. Conclusion

In this paper, we proposed an intrusion detection scheme based on the burst characteristics of the DNP3 environment, and the proposed scheme can detect attacks using allowed bursts not detected by the conventional burst-based intrusion detection scheme.

By adding the sub-burst concept, the implemented intrusion detection system based on the proposed scheme can execute the intrusion detection algorithm before the burst is configured. As a result, the proposed scheme can execute authentication for non-critical ASDU that violates rules. We also proposed a model where white list rules can be generated automatically and updated. Finally, we verified the proposed scheme by implementing an intrusion detection system.

Because we implement the intrusion detection system only applying authentication in this paper, we will verify the scheme using white list update automation by implementation.

In the future, we will continuously investigate the research related to the specialized behaviors of the intrusion detection system using the specialized behavior that performs changes in the session key if authentication-related risks occur.

For future studies, in order to perform closer and more realistic work, research that uses actual traffic is required. In addition, research is required that not only designs a model based on a predetermined period, but the model also considers special individual situations.

References

[1] G.-U. Ha, N.-H. Min, H.-S. Hwang, and C.-H. Lim, “Proposal of information security management system of special control network,” *J. KIISC*, vol. 24, no. 3, pp. 54-66, Jun. 2014.

[2] O. Khaled, A. Marin, F. Almenares, P. Arias, and D. Diaz, “Analysis of secure TCP/IP profile in 61850 based substation automation system for smart grids,” *Int. J. Distrib. Sensor Netw.*, vol. 2016, Article ID 5793183, p. 11, 2016. doi:10.1155/2016/5793183

[3] A. Shahzad, K. P. Udagepola, Y.-K. Lee, S. Park, and M. Lee, “The sensors connectivity within SCADA automation environment and new trends for security development during multicasting routing transmission,” *Int. J. Distrib. Sensor Netw.*, vol. 2015, Article ID 738687, p. 15, 2015. doi:10.1155/2015/738687

[4] IEC, *IEC/TS 62351-1:2007*(E), pp. 15-17,

2007.

[5] P. Koh, H.-J. Choi, S.-R. Kim, H.-M. Kwon, and H.-K. Kim, "Intrusion detection methodology for SCADA system environment based on traffic self-similarity property," *J. KIISC*, vol. 22, no. 2, pp. 267-281, Apr. 2012.

[6] J. Park, N. Min, G.-U. Ha, G. Yu, and K.-Y. Song, "Introduction of mechanism for secure authentication in the power control system," *J. KIISC*, vol. 24, no. 3, pp. 44-53, Jun. 2014.

[7] IEEE Power and Energy Society, IEEE Std 1815:2012, 2012.

[8] Digital Bond Inc. Quickdraw SCADA IDS from http://www.digitalbond.com/tools/quickdraw.

[9] T. Mander, F. Nabhani, L. Wang, and R. Cheung, "Data object based security for DNP3 over TCP/IP for increased utility commercial aspects security," in *Proc. Power Eng. Soc. Gen. Meeting*, pp. 1-8, Tampa, FL, Jun. 2007.

[10] J.-H. Yun, S.-H. Jeon, K.-H. Kim, and W.-N. Kim, "Burst-based anomaly detection on the DNP3 protocol," *Int. J. Control and Automat.*, vol. 6, no. 2, pp. 313-324, Apr. 2013.

[11] Triangle Microworks, Inc. Protocol Source Code Libraries DNP3 from http://www.triangle microworks.com/products/source-code-libraries /dnp-scl-pages/overview.

[12] M. Jang, G. Lee, S. K. Kim, B. Min, W. Kim, and J. Seo, "Testing vulnerabilities of DNP3," *J. Secur. Eng.*, vol. 7, no. 1, pp. 15-28, Feb. 2010.

[13] B. Zhu, A. Joseph, and S. Sastry, "A taxonomy of cyber attacks on SCADA systems," in *Proc. IEEE ITHINGSCPSCOM '11*, pp. 380-388, Dalian, China, Oct. 2011.

[14] I. Garitano, R. Uribeetxeberria, and U. Zurutuza, "A review of scada anomaly detection systems," in *SOCO 2011, AISC,* vol. 87, pp. 357-366, 2011.

[15] A. Almalawi, X. Yu, Z. Tari, A. Fahad, and I. Khalil, "An unsupervised anomaly-based detection approach for integrity attacks on SCADA systems," *Comput. Secur.*, vol. 46, pp. 94-110, Oct. 2014.

[16] A. Nicholson, S. Webber, S. Dyer, T. Patel, and H. Janicke, "SCADA security in the light of Cyber-Warfare," *Comput. Secur.*, vol. 31, no. 4, pp. 418-436, Jun. 2012.

[17] T. Mander, R. Cheung, and F. Nabhani, "Power system DNP3 data object security using data sets," *Comput. Secur.*, vol. 29, no. 4, pp. 487-500, Jun. 2010.

하 기 웅 (Gi-ung Ha)

2012년 2월 : 동아대학교 컴퓨터공학과 학사

2015년 2월 : 동국대학교 정보보호학과 석사

2015년 6월~현재 : 한전 KDN 정보보안사업처 주임

<관심분야> 정보보호, 제어시스템보안

임 대 운 (Dae-woon Lim)

1994년 8월 : KAIST 전기및전자공학과 학사

1997년 2월 : KAIST 전기및전자공학과 석사

2002년 8월 : 서울대학교 전기컴퓨터공학부 박사

1995년 9월~2002년 8월 : LS산전 중앙연구소 선임연구원

2006년 9월~현재 : 동국대학교 정보통신공학과 부교수

<관심분야> 암호학, 제어시스템보안

장 민 호 (Min-Ho Jang)

2002년 8월 : 연세대학교 전기전자공학부 공학사
2004년 8월 : 서울대학교 전기컴퓨터공학부 공학석사
2009년 2월 : 서울대학교 전기컴퓨터공학부 공학박사
2009년 3월~2011년 8월 : 삼성전자 DMC연구소 책임연구원
2011년 9월~현재 : 울산과학대학교 전기전자공학부 부교수
<관심분야> 이동통신시스템, 오류정정부호, OFDM, 암호학, 정보보호, 제어시스템보안

장 지 웅 (Ji-Woong Jang)

2000년 2월 : 서울대학교 전기공학부 공학사
2002년 2월 : 서울대학교 전기컴퓨터공학부 공학석사
2006년 2월 : 서울대학교 전기컴퓨터공학부 공학박사
2006년 3월~2008년 6월 : 삼성전자 책임연구원
2008년 8월~2009년 7월 : Post Doctor in UCSD, USA
2009년 8월~2012년 8월 : LG전자 책임연구원
2012년 9월~현재 : 울산과학대학교 컴퓨터정보학부 부교수
<관심분야> 시퀀스, 암호학, 오류정정부호, 디지털통신, 정보보호, 제어시스템보안