A Study on Quantification of Risk Assessment in Security Design

Yasuyuki KAWANISHI*, Yoichi HATA, Hideaki NISHIHARA, Daisuke SOUMA, and Hirotaka YOSHIDA

For further automation and efficiency improvement of production, industrial control systems have been increasingly connected to other information systems to exchange data. Under this circumstance, establishing security measures for control systems against malware (such as Stuxnet*1) is an urgent issue. Therefore, security design has been taken into consideration at the beginning of the system development. In collaboration with the National Institute of Advanced Industrial Science and Technology (AIST), we have been applying security design guidelines for automobiles to control systems, aiming to improve the efficiency of design procedures without depending on personal knowledge or experience. Focusing on the risk-assessment phase in security design consisting of multiple phases, this paper proposes a quantification method optimized for the risk assessment of control devices and systems by utilizing an existing vulnerability assessment system. We report on the security design results using the method, providing a case study on a control system equipped with a data logger as the key element.

Keywords: control system, security design, risk assessment, CWSS

1. Introduction

Recently, information and communication technology (ICT) has been increasingly applied to control systems. By applying ICT, such as information networks, general-purpose operating systems (OSs), and communication interfaces whose use has spread in information devices and systems, to control devices, a series of control processes has been streamlined and automated. These processes include the collection of sensing data generated and transmitted by field devices, analysis of data by control servers, system maintenance by maintenance personnel using human-machine interface, monitoring of an overall system in the host information network, and feedback to the control.

In the automobile industry, products have been developed to offer various services through connectivity (a system model called “connected-car”). Specifically, in-vehicle devices communicate via Ethernet, dedicated short range communications (DSRC), Wi-Fi, etc. to achieve collision avoidance between vehicles, remotely update software through communication between vehicles and the center, and so forth.

In terms of control systems and devices, it is urgently required to ensure cybersecurity in developing control devices, as evidenced by the Stuxnet attack on power plants in 2010[1] and the information security weakness of data loggers published in 2017 (CVE-2017-6048).[2] Regarding information systems and devices, a security design is a well-known process to systematically implement all the necessary security countermeasures under the cost constraints of a business, among other factors. In fact, when developing a secure product, it is required by the ISO/IEC 15408 standard[3] to implement a process in the review phase in order to derive security requirements from the system specifications before commencing product development. This process consists mainly of definition of target of evaluation (TOE),*2 threat identification, risk assessment, formulation of security objectives, and selection of security requirements.

It is imperative to establish a security design methodology that is suitable for the development of control devices and in-vehicle devices by applying the standard and embodying and optimizing the means of implementation as necessary. Various such efforts have been made in the automobile industry recently. In 2015, JASO TP15002[4] was released by the Society of Automotive Engineers of Japan, Inc. (JSAE) as the standard security design guidelines for vehicles. This design methodology is also used in Recommendation ITU-T X.1373[5] published in 2017 for remote software update between vehicles and the center.

To determine the applicability of JASO TP15002 to the security design of our automobile-related products, we applied it to a conceptual vehicle model to study the increase in efficiency.[6] Since the guidelines for the automobile industry define both specific and generally applicable procedures, we conducted a review to determine its applicability to control systems at plants, in particular.[14] Research has been steadily conducted to develop a system for reducing the labor cost by semi-automating the process from threat extraction to risk analysis and eliminating the dependence on personal skills.

This paper focuses on the risk assessment of JASO TP15002. We developed the Risk Scoring System based on CWSS (RSS-CWSS), a methodology for quantifying the risk assessment, by applying the Common Weakness Scoring System (CWSS: a scoring system for evaluating the weakness of software), as an evaluation scoring system for quantification suitable for control devices and systems, and compared the results with those of the CVSS-based Risk Scoring System (CRSS: an existing scoring system) for a review. In a case study described in this paper using a data logger, we confirmed that the evaluation values were moderately distributed in RSS-CWSS compared to CRSS, and clarified the effectiveness of threat prioritization.

Chapter 2 describes the preparations for the evaluation related to this paper. Chapter 3 presents the issue identified based on Chapter 2. Chapter 4 proposes a methodology, and Chapter 5 discusses a case study on the security design of a data logger using the proposed methodology. Chapter 6 draws some conclusions.

2. Preliminary

2-1 Security design

A security design refers to the process of incorporating security requirements based on specifications before developing a product. ISO/IEC 15408 has been established as a standard, and a corresponding international authentication scheme has been established as common criteria. Regarding industrial control systems, a review has been conducted on guidelines and frameworks in IEC62443,[7] UL2900-2-2,[8] and other standards.

In the automobile field, as mentioned earlier, JASO TP15002 clearly prescribes the process of deriving security requirements (modelling from the specifications, threat extraction, risk analysis, derivation of security objectives) as in the case of the model-based risk analysis approach conducted by Lund et al.[9] This standard helps facilitate the process of visualizing attacks (including unauthorized operation and access to the communication with the vehicle control ECU from outside) and deriving countermeasures. We considered that the guidelines might be applied to systems other than vehicles; this was the motivation of our research. Figure 1 shows the process flow of JASO TP15002.

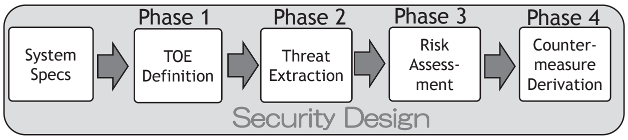

Fig. 1. Process Flow of Security Design

The work implemented in each phase is explained below.

Phase 1: Definition of TOE. We create a data flow diagram (DFD), as shown in Fig. 3, to clearly indicate the assets to be protected in TOE. We define the importance of assets, entry points for attacks from outside, and respective phases and personnel involved (e.g., system development, sales, operation).

Phase 2: Threat extraction. In this phase, we list all the threats to TOE and circumstances. Specifically, unfavorable operations of TOE are described based on the Five Ws ("Who," "When," "Where," "Why," "What") as shown in Table 1. Regarding the description method for "What", the "asset container" method described in Chapter 2-5 "Preliminary research" is used.

Phase 3: Risk assessment. In this phase, the threats extracted in Phase 2 are quantified and prioritized (Table 3). The following explanation is based on an example of evaluation by CRSS, one of the risk assessment systems referenced in JASO TP15002. CRSS is a Common Vulnerability Scoring System version 2

Table 1. Example of Threat List in JASO TP15002

#	Where	Who	When	Why	What	(At	Asset)
1	Ethernet	Outsider	at purchase	Maliciously	get firmware and analyze firmware vulnerbility	Control	Firmware
2	Ethernet	Operator	in regular use	Accidentally	cause malfunction	Network Interface	Communication function
3	Ethernet	Outsider	in regular use	Maliciously	exploit server	Network Interface	Communication function
4	Ethernet	Operator	in regular use	Accidentally	cause malfunction	Network Interface	Authentication function
5	Ethernet	Outsider	in regular use	Maliciously	cause malfunction	Network Interface	Authentication function
6	Ethernet	Operator	in regular use	Accidentally	leak information	Network Interface	Authentication information
7	Ethernet	Outsider	in regular use	Maliciously	steal information	Network Interface	Authentication information
8	Ethernet	Operator	in regular use	Accidentally	overwrite with wrong firmware	Control	Firmware
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
47	Modbus Serial	Operator	in regular use	Accidentally	set wrong data	Control	Configuration information

Table 2. Example of Metric Definition for Risk Assessment

Metric	Rank	Criteria	Value
Access Vector(AV): Distance from threat	Local(L)	Serial, ModbusSerial	0.395
	Adjacent(A)	-	0.646
	Network(N)	Ethernet	1.000
Access Complexity(AC): Number of penetrations	High(H)	3 or more	0.350
	Medium(M)	2	0.610
	Low(L)	1	0.710
Authentication(Au): Number of authentication	Multiple(M)	2 or more	0.450
	Single(S)	1	0.560
	None(N)	0	0.704

Module	Asset	Confidentialty			Integrity			Availability		
		None 0.000	Partial 0.275	Complete 0.660	None 0.000	Partial 0.275	Complete 0.660	None 0.000	Partial 0.275	Complete 0.660
Control	Configuration information	✔				✔		✔		
	Firmware			✔			✔	✔		
	PLC status	✔					✔	✔		
Network Interface	Communication function	✔					✔			✔
	PLC status	✔					✔	✔		
	Authentication function	✔					✔			✔
	Authentication information			✔			✔	✔		
Storage	PLC status	✔					✔	✔		

Table 3. Example of Prioritized Threat List

#	AV	AC	Au	AE	EF-C	EF-I	EF-A	EF	Risk Value
1	N	L	N	10.00	Complete	Complete	None	9.21	9.43
2	N	L	N	10.00	None	Complete	Complete	9.21	9.43
3	N	L	N	10.00	None	Complete	Complete	9.21	9.43
4	N	L	N	10.00	None	Complete	Complete	9.21	9.43
5	N	L	N	10.00	None	Complete	Complete	9.21	9.43
6	N	L	N	10.00	Complete	Complete	None	9.21	9.43
7	N	L	N	10.00	Complete	Complete	None	9.21	9.43
8	N	M	N	8.59	Complete	Complete	None	9.21	8.77
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
47	L	M	N	3.39	None	Partial	None	2.86	1.85

(CVSS v2)[10]-based risk assessment system. Individual threats extracted in Phase 2 are selected for respective classifications based on the metrics (Table 2). The weighted values are applied to the equation to calculate the risk values. The threats are classified based on the risk values: Level III (serious), Level II (warning), Level I (caution).

Phase 4: Derivation of security objectives. In this phase, the causes of a threat are broken down based on tree topology, and a security objective is derived from each leaf. This tree is similar to the one used in Fault Tree Analysis (FTA) for safety analysis and is referred to as an attack tree (AT).[11],[12] A matrix is finally created to indicate the correlation between threats and security objectives (Table 4). The guideline do not clearly indicate whether the tree analysis should be conducted on all the threats or only some threats whose risk values are high due to cost and time constraints. We focused on this point in our research.

Table 4. Example of Threat - Security Objective Matrix

		Security Objectives															
		Object							Environment								
		1	2	3	5	7	10	11	4	6	8	9	12	13	14	15	16
		O.Disable_Control_Command	O.Disable_Firmware_Access	O.Encrypted_Firmware	O.IP_Restriction	O.Interface_with_Password	O.Port_Disable	O.Malware_Disinfection	OE.Secure_Firmware_Development	OE.Operation_Management	OE.Setting_Data_Management	OE.System_Integration	OE.Dedicated_Use	OE.Trusted_Operator	OE.Ethics_Education	OE.Version_Management	OE.Restriction_Physical_Access
Threat	1	X	X	X					X								
	2				X	X	X			X	X	X					
	3	X			X	X	X			X	X						
	4	X			X	X	X			X	X						
	5	X			X	X	X			X	X						
	6					X		X					X	X	X		
	7	X			X	X	X	X					X		X		
	8		X							X						X	
	9	X	X	X					X						X		
	10				X	X	X				X						
	11	X			X	X	X			X					X		
	12				X		X			X	X						
	13	X			X	X	X			X							
	14									X						X	
	15	X		X					X						X		
	16	X		X					X						X		
	17	X			X		X								X		
	18	X			X		X								X		
	19	X			X		X								X		
	20									X	X						
	21									X	X						
	22	X												X	X		
	23					X					X						
	24									X							
	25	X		X					X						X		X

2-2 Vulnerability assessment

A weakness evaluation aims to quantify the weakness of an existing system. JASO TP15002 attempts to apply a weakness evaluation to a security design in the early stage of development (i.e., specifications stage). The typical vulnerability assessment methods are as follows:

CVSS v2: ITU-T Recommendation X.1521[10] formulated in 2007. This vulnerability evaluation methodology is designed for information systems in which multiple devices are connected.

CWSS: ITU-T Recommendation X.1525[13] released in 2015. This evaluation methodology aims to quantify the weakness (as its name suggests) in a system more extensively.

2-3 CWSS metrics

CWSS defines 16 metrics in total from three viewpoints: base finding, attack surface, and environment (Table 5).

Table 5. CWSS Metrics[13]

	Metric	Description
Base Finding	Technical impact (TI)	The potential result that can be produced by the weakness.
	Acquired privilege (AP)	The type of privileges that are obtained by an attacker.
	Acquired privilege layer (AL)	The operational layer to which the attacker gains privileges.
	Internal control effectiveness (IC)	The ability of the control to render the weakness unable to be exploited by an attacker.
	Finding confidence (FC)	The confidence that the reported issue is a weakness that can be utilized by an attacker.
Attack Surface	Required privilege (RP)	The type of privileges that an attacker must already have in order to reach the code/functionality that contains the weakness.
	Required privilege layer (RL)	The operational layer to which the attacker must have privileges.
	Access vector (AV)	The channel through which an attacker must communicate to reach.
	Authentication strength (AS)	The strength of the authentication routine.
	Level of interaction (IN)	The actions that are required by the human victim(s) to enable a successful attack.
	Deployment scope (SC)	Whether the weakness is present in all deployable instances of the software, or limited.
Environment	Business impact (BI)	The potential impact to the business or mission.
	Likelihood of discovery (DI)	The likelihood that an attacker can discover the weakness.
	Likelihood of exploit (EX)	The likelihood that an attacker would be able to successfully exploit it.
	External control effectiveness (EC)	The capability of controls or mitigations outside of the software.
	Prevalence (P)	How frequently this type of weakness appears in software.

2-4 Risk assessment in a security design

Risk assessment in a security design refers to the process of extracting threats from the system specifications before the product release and prioritizing the threats based on objective quantification. JASO TP15002 indicates CRSS (a risk assessment system to which CVSS v2, a vulnerability scoring system, is applied) mentioned above as an example. It was found that the vulnerability scoring system originally designed to be used after product release can also be used for risk assessment before product release.

2-5 Preliminary research

At the International ERCIM/EWICS/ARTEMIS Workshop on "Dependable Smart Embedded and Cyber-physical Systems and Systems-of-Systems" in 2017 (DECSoS 2017),[6] we proposed reducing the workload of risk analysis by extracting issues when JASO TP15002 was applied to TOE in a vehicle. Specifically, the proposal was intended to conduct a risk analysis before performing work that required large number of man-hours to derive countermeasures for threats and to reduce the number of target threats to an appropriate level for the cost. To this end, we considered which of the Five Ws (that determine a threat) should be focused on to achieve quantification. To reach a solution, we broke down “What” into “Asset” and “At” (subject to intrusion) in combination with the use of “Where.” Thus, we proposed a methodology to quantify risks while preventing omissions of threats given the importance of the attack route (“Where” to “At”) and “Asset” attacked. The relationship between “Where,” “At,” and “Asset” can be likened to a container that stores assets and its opening (see Fig. 2). Thus, we named this methodology the “asset container” method.

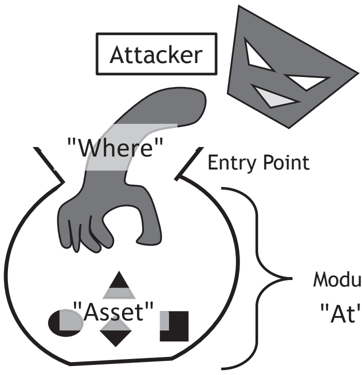

Fig. 2. Concept of “Asset Container” Method[6]

This proposed methodology has the following advantages over a method to evaluate risks from the viewpoint of attackers and attack scenarios:

- Eliminates the dependence on personal skills (evaluator’s findings).
- Prevents the omission of threats by considering only combinations derived from the specifications of devices and systems.
- Reduces the man-hours required for risk analysis by enabling unique judgments based on three metrics (number of viewpoints smaller than Five Ws).

It should be noted that in-vehicle devices and devices in a control system have much in common (e.g., the assets which require protection include control functions, priority is placed on integrity and availability). At the Computer Security Symposium 2017 (CSS2017),[14] we reported the results of a case study using a data logger in which JASO TP15002 was applied to an industrial control system. Specifically, we created a TOE that incorporated a data logger as shown in Fig. 3 and extracted 47 threats.

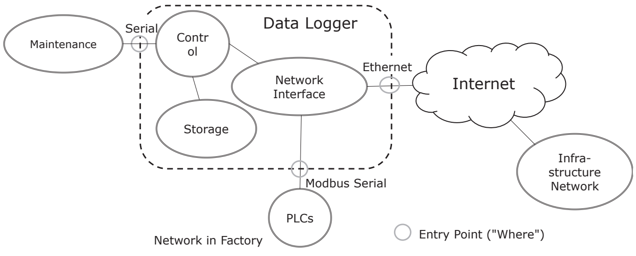

Fig. 3. TOE incorporated Data Logger[14]

3. Issues

As discussed in the previous chapter, JASO TP15002 can be applied to control systems. However, CRSS, a risk quantification system, conforms to CVSS v2, an old standard. So a further review of the literature[14] was necessary to determine the applicability to control devices.

In this research, we developed a new means of quantification using the same TOE as a substitute for CRSS, and made a comparison and conducted a review. In the comparison phase, we checked the difference in workload in the process of selecting security objectives in Phase 4. Based on our evaluation standard, a quantification system with fewer threats subject to attack tree analysis before all the effective security objectives were identified was considered to be superior.

When CRSS was used for risk quantification, the risk values tended not to be distributed properly. Multiple threats were ranked on the same level, making prioritization difficult. Specifically, in the risk quantification described in the literature,[14] there were seven threats with the risk value ranked first, six threats with the risk value ranked 14th, and eight threats with the risk value ranked 20th. As a result, the threats could not be identified properly. The number of threats that required analysis in Phase 4 was almost half of all the threats. Thus, there was room for improvement.

4. Newly Considered Quantification Method for Risk Assessment

We conducted an experiment to determine whether the CWSS metrics can be applied as a new quantification methodology to solve the issues mentioned above.

4-1 RSS-CWSS

We defined Risk scoring system based on CWSS (RSS-CWSS) as a risk assessment system using 10 metrics in Table 5 (with six in Table 6 excluded).

Table 6. CWSS Metrics[4] Used as Fixed Value

Metric	Code	Value	Description
AL	A(Application)	1.00	The attacker acquires all privileges.
IC	N(None)	1.00	No controls exist.
FC	T(Proven True)	1.00	The vulnerability is reachable by the attacker.
IN	A(Automated)	1.00	No human interaction is required.
SC	R(Rare)	0.50	Only present in rare platforms.
P	W(Widespread)	1.00	The influence of the attack spreads widely.

4-2 CWSS equation

For the CWSS risk values that served as the dataset for RSS-CWSS, values between 0 and 100 were recorded and were multiplied by weight variables for each metric based on the equation below.[13]

Risk value = BaseFindingSubscore × AttackSurfaceSubscore × EnvironmentSubscore

BaseFindingSubscore =

$$[(10.0 \times TI + 5.0 \times (AP + AL) + 5.0 \times FC) \times f(TI) \times IC)] \times 4.0$$

If TI = 0.0, f(TI) = 0.0. For others, f(TI) = 1.0.

AttackSurfaceSubscore =

$$[20.0 \times (RP + RL + AV + SC) + 15.0 \times IN + 5.0 \times AS]/100.0$$

EnvironmentSubscore =

$$[(10.0 \times BI + 3.0 \times DI + 4.0 \times EX + 3.0 \times P) \times f(BI) \times EC)]/20.0$$

If BI = 0.0, f(BI) = 0.0. For others, f(BI) = 1.0.

5. Review results

5-1 Results of introducing RSS-CWSS

In the case study, we applied the systems and compared the results.

While CRSS used six metrics, RSS-CWSS used 10 metrics. The increase in the number of metrics improved the distribution of risk values. For example, the CRSS evaluation results showed as many as seven threats whose risk value was ranked top (Table 7). In RSS-CWSS, the results were distributed in five clusters.

Table 7. Comparison of Risk Values

Threat No.	CRSS	RSS-CWSS
13	9.43	75.8
36		33.7
31		23.3
35		
37		18.1
30		13.2
34		

Threats were analyzed in descending order of the risk scores. Security objectives were extracted in Phase 4. And the security objectives extracted for the first time against the threats were plotted and connected by a line in Fig. 4.

As shown in Fig. 4, all the security objectives were identified for the 25th threat in CRSS and the 14th threat in RSS-CWSS. Based on the evaluation standard discussed in Chapter 3, RSS-CWSS was superior to CRSS.

We confirmed that RSS-CWSS, which properly selected the risk quantification metrics and their number, could properly distribute the risk values in the risk analysis phase, effectively narrow down the threats, and reduce the workload in the subsequent phase.

5-2 Discussion on the results

Regarding the differences in the results obtained from different risk analysis quantification systems, we considered based on the review results that the differences were simply attributable to the number of metrics. In actual applications, it is also important to consider whether the content of definition of metrics is appropriate for the system subject to risk assessment. For example, RSS-CWSS has both technical impact (TI) and business impact (BI) as metrics of impact caused by an attack on assets (see Table 5). In RSS-CWSS, differences are likely to occur in risk values even in the case of a data logger whose attack routes are limited. In other words, it is important to use a risk assessment system that has many metrics to clearly differentiate the system configuration elements (i.e., only having many metrics may not necessarily help) and classify the weight variables of metrics depending on the current condition of the system while avoiding bias.

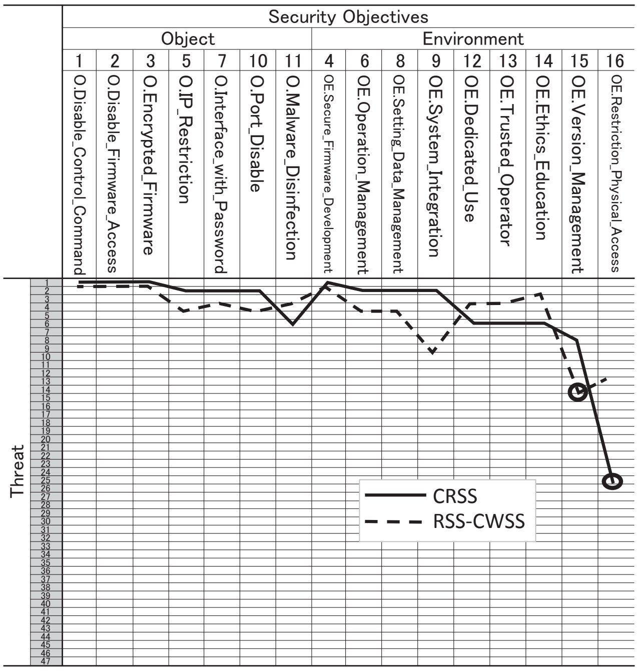

Fig. 4. Comparison of Threat Analysis Amount Required to Extract All Security Objectives

6. Conclusion

We introduced a security design procedure for control systems and reported the results of a review regarding the processes before and after risk assessment and improvement of quantification methodology in order to reduce workload and eliminate the dependence on personal skills. We developed RSS-CWSS as a risk assessment methodology that applied CWSS, a weakness evaluation methodology, for control systems. We conducted a follow-up review up to the phase of deriving security objectives, and confirmed that RSS-CWSS can reduce the number of analyses and effectively narrow down the threats by selecting appropriate metrics in the risk quantification and moderately distributing the risk values.

Technical Terms

＊1 Stuxnet: A malware (malicious software) that was used to attack Iranian nuclear facilities in 2010. Stuxnet and its variants caused significant damage to industrial control systems.

＊2 Target of evaluation (TOE): In the security design, the design target is defined as a model.

References

(1) S. Karnouskos: “Stuxnet Worm Impact on Industrial Cyber-Physical System Security.” In: “37th Annual Conference of the IEEE Industrial Electronics Society (IECON 2011), Melbourne, Australia,” (November 2011)

(2) NIST, CVE-2017-6048 Detail
available at https://nvd.nist.gov/vuln/detail CVE-2017-6048

(3) ISO/IEC 15408-1:2009 Information technology Security techniques Evaluation criteria for IT security Part 1: Introduction and general model

(4) JASO, “TP15002: Guideline for Automotive Information Security Analysis” (2015)

(5) ITU-T X.1373: Secure software update capability for intelligent transportation system communication devices

(6) Y. Kawanishi, H. Nishihara, D. Souma and H. Yoshida, ”Detailed analysis of security evaluation of automotive systems based on JASO TP15002,” DECSoS: Dependable Smart Embedded Cyber-physical Systems and Systems-of-Systems, LNCS 10489, 2017, Springer

(7) IEC TS 62443-1-1:2009 Industrial communication networks - Network and system security - Part 1-1: Terminology, concepts and models

(8) UL 2900-2-2: Outline of Investigation for Software Cybersecurity for Network-Connectable Products, Part 2-2: Particular Requirements for Industrial Control Systems (2016)

(9) M. S. Lund, B. Solhaug and K. Stolen, Model-Driven Risk Analysis, the CoRAS Approach, Springer-Verlag Berlin Heidelberg (2011)

(10) ITU-T X.1521 (04/2011): Cybersecurity information exchange, Vulnerability/state exchange, Common vulnerability scoring system

(11) A. Roy, D. S. Kim, and K. S. Trivedi, “Cyber security analysis using attack countermeasure trees.” Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research. ACM (2010)

(12) A. Roy, D. S. Kim, and K. S. Trivedi: ACT: Towards unifying the constructs of attack and defense trees, Security and Communication Networks, 2011:3:1-15

(13) ITU-T X.1525: Cybersecurity information exchange, Vulnerability/state exchange, Common weakness scoring system

(14) Y. Kawanishi, H. Nishihara, D. Souma, H. Yoshida, and Y. Hata, “A study on security design for industrial control system including data logger,” CSS2017 (in Japanese)

Contributors

The lead author is indicated by an asterisk (＊).

Y. KAWANISHI*

- Assistant Manager, Cyber-security R&D Office

Y. HATA

- General Manager, Cyber-security R&D Office

H. NISHIHARA

- Doctor of Science
  National Institute of Advanced Industrial Science and Technology
  The SEI-AIST Cyber Security Collaborative Research Laboratory

D. SOUMA

- Ph.D in Information Science
  National Institute of Advanced Industrial Science and Technology
  The SEI-AIST Cyber Security Collaborative Research Laboratory

H. YOSHIDA

- Doctor of Engineering (Ph.D)
  National Institute of Advanced Industrial Science and Technology
  The SEI-AIST Cyber Security Collaborative Research Laboratory