Journal of Engineering and Applied Sciences 13 (6): 1338-1346, 2018
ISSN: 1816-949X
© Medwell Journals, 2018

A Study on Security Vulnerabilities Assessment and Quantification in SCADA Systems

[1]Zakuan Firdaus, [1]Norziana Jamil, [1]Qais Saif Qassim, [1]Mohd Ezanee Rusli,
[2]Norhamadi Ja'affar, [2]Maslina Daud and [2]HafizahChe Hasan
[1]Center of Information Network and Security,
College of Computer Science and Information Technology, Universiti Tenaga Nasional,
Kajang, Selangor, Malaysia
[2]Department of Security Assurance, Cybersecurity Malaysia,
Seri Kembangan, Selangor, Malaysia

Abstract: Supervisory Control And Data Acquisition Systems (SCADA) monitor and control industrial and critical infrastructure functions such as electricity, oil, water and natural gas production and distribution processes. Consequently, failure in the intended operation of SCADA system results in catastrophic consequences. With the increased interconnectivity of SCADA systems and the commercial availability of cloud computing, SCADA systems have increasingly adopted Internet of Things (IoT) technologies to significantly reduce infrastructure costs and increase ease of maintenance and integration. As a result, the exposure of these systems to cyber threats has increased enormously. Therefore, there is a necessity to identify, remediate and mitigate system's security vulnerabilities to protect and prevent possible attacks. This study serves two folds; firstly, different types of vulnerabilities in SCADA systems have been identified and reviewed. Secondly, two test cases have been presented to demonstrate the severity of the identified vulnerabilities on SCADA systems. This study draws attention to the impact of threat on SCADA systems and their consequences.

Key words: SCADA, vulnerability assessment, risk assessment, vulnerability severity, attack consequences, protect

INTRODUCTION

Industrial Control System (ICS) are systems that are used to control critical infrastructure such as electric, water and wastewater, oil and natural gas, transportation, chemical, pharmaceutical and more. Supervisory Control and Data Acquisition (SCADA) systems are one of the ICS that are being used in the industries to control and monitor those critical infrastructures. Therefore, due to the wide application of SCADA systems, their security issues and weaknesses have been a primary concern. Cyber-attacks to these systems have the potential to result in catastrophic consequences in the physical domain. For example, power systems equipment could be damaged, reduced power quality could occur potentially leading to blackouts and in extreme cases, result in safety related incidents. The development of a trustworthy industrial control system requires a deeper understanding of potential impacts resulting from successful cyber attacks. Therefore, estimating feasible attack impact and identifying system vulnerabilities are major concern in SCADA implementations.

Risk and vulnerability analyses provide important knowledge of how can prevent, prepare for and manage crises. Risk is traditionally defined as the impact times the likelihood of an event (Miller and Rowe, 2012). ISO 3100: 2009 have defined likelihood as the chance that something might happen which can be defined, determined or measured objectively or subjectively and can be expressed either qualitatively or quantitatively. Once potential vulnerabilities are discovered an impact analysis should be performed to determine the risk and the consequences to the system functions.

An overview of SCADA systems: SCADA system monitors and controls critical infrastructures services, it collects data from remote locations equipment and transmits it to a central computer facility also known as a

Corresponding Author: Zakuan Firdaus, Center of Information Network and Security, College of Computer Science and Information Technology, Universiti Tenaga Nasional, Kajang, Selangor, Malaysia

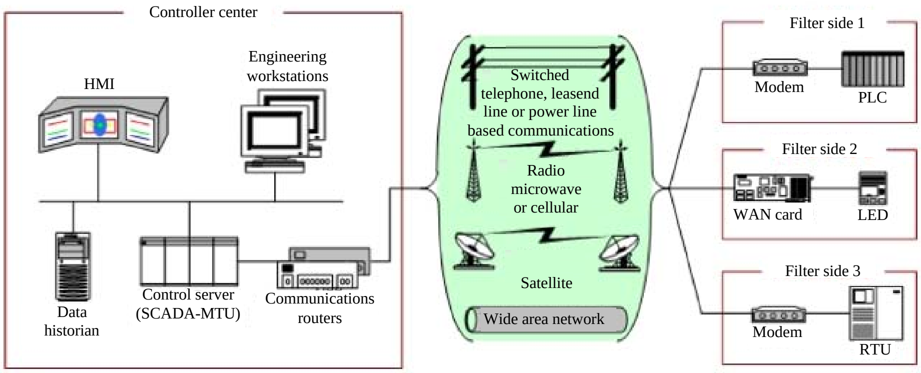

Fig. 1: Example of SCADA system general layout (Stouffer *et al*., 2011)

master station. SCADA is very important because it controls critical infrastructures such as power utility, oil and gas pipeline, transportation and water waste collection (Stouffer *et al*., 2011; Robles and Kim, 2010). Typically, in SCADA systems, the information gathered from remote telemetry unit or remote Terminal Unit (RTU) is sent to the control server (MTU) which is located at the master station and will be displayed on the Human-Machine Interface (HMI) to the operator graphically or textually. Thus, the entire system can be monitored and controlled from a central location by operators in real-time. Based on the complexity and configuration of the individual system, control, operation or task of an individual system may be performed automatically or by operator commands (Stouffer *et al*., 2011). For example, an anomaly is detected in a plant and this information is then transmitted to the central host and alerts the central host that an anomaly has occurred. In addition, it also carries necessary analysis and control and displays the information to the operator. Similarly, automated or operator driven solutions or tasks can be passed back to the remote site or plant.

SCADA components: Components that form a SCADA system can be classified into four groups which are field devices, local controller, communication channel and central station as depicted in Fig. 1.

Field devices: Field devices are operating equipment such as actuators and instruments such as sensors. These devices are known as SCADA eyes, ears and hands because without these devices SCADA system is not complete (Robles and Kim, 2010). The information produced by these devices is passed to the local controller.

Local controller: local controllers or also known as local field processor are devices that communicate with field devices. Typical local controller used in SCADA system is Remote Terminal Unit (RTU) or/and Programmable Logic Control (PLC). Historically, RTUs and PLCs are distinctly different devices, yet after some time they are now practically the same (Igure *et al*., 2006). They receive input from instruments and send output to operating equipment. Local controllers control actuators and monitor the sensors; this is how they control the local processes. In some cases, local controllers may perform operation automatically without the assistance from central station because they usually have control programming stored locally. These local controllers provide input and receive output to/from central station.

Communication channel: The motivation behind the communication channel within SCADA system is to provide connectivity between field devices and local controllers and also between local controllers and central station. In traditional SCADA links, the information can be transmitted utilizing an assortment of various communication platforms, for example, Ethernet, phone line, fiber optic, radio/remote, cell, satellite, Wi-Fi and microwave.

Central station: Central station or master station acts as a central point of monitoring and control (Schneider Electric, 2012). Central stations are usually designed to house a few important components, for example, control server (MTU), communication routers, data historian (centralized database for logging process information), Human-Machine Interface (HMI) and engineering workstations. The control center collects and logs data obtained from the field sites, present the collected data on the HMI and might generate actions based upon specific and predefined events (Stouffer *et al*., 2011).

Attacks on SCADA systems: There are few attacks on SCADA system throughout the years that have been recorded. These attacks caused economic loss, physical damage to the organization process and disclosure of important and confidential data. On August 2003, nuclear regulatory has confirmed that on January 2003, a private computer network at Davis-Besse Nuclear power plant which is located in Oak Harbor, Ohio has been infected by Slammer, a Microsoft SQL Server worm. Slammer disabled the safety monitoring system for almost 5 h. It also made the plant's process computer unavailable and it took around 6 h for it to become available again (Jonathan *et al*., 2010). Another successful attack targeted Gazprom, Russia's huge gas monopoly was one of the developing number of targets hit several years ago for computer hackers. Hacker managed to move beyond the company's security and break into the system that control gas flow pipelines. The central switchboard of gas flows was for quite a while under the control of external users. The incident happened in April 1999 (Jonathan *et al*., 2010; Extreme Network, 2017).

Iranian nuclear facility at Natanz has been struck by a worm known as Stuxnet in June 2010. The worm used four vulnerabilities that are previously unknown, so, there is no time to create or distribute patches. The unknown vulnerabilities is called zero-day vulnerabilities. The worm utilizes Siemen's default passwords to access windows operating system that run WinCC and PCS7 programs. Frequency converter drives made by FararoPaya in Iran and Vacon in Finland were shut down by Stuxnet. These drives were used to power centrifuges used in the concentration of the uranium-235 isotope. Stuxnet adjusted the frequency of the electrical current to the drives making them switch between high and low speeds for which they were not designed. This switching lead the centrifuges to fail at a higher than normal rate. Recently, Ukrainian utility company has been a target of another successful attack on December 2015. It was reported that almost 225,000 customers were affected as a result of the attack. Based on the analysis from ICS, the cyber-attack was done using malware known as BlackEnergy which allowed the attacker to unauthorized access and control the HMI remotely.

Generally, the main goal of attacking SCADA system is to harm the security properties such as confidentiality, integrity and availability. Different types of attacks may introduce harm on SCADA systems, attacks may include (but not limited to) the follow (Robles and Kim, 2010):

- Denial of Service (DoS); DoS attacks affect the system availability due to shutdown condition of the server
- File deletion on SCADA server; file manipulation attacks affect data integrity and availability of the system
- Trojan plantation: this type of attack would affect confidentiality, integrity and availability of the system due to trojan capabilities to take complete control of the system
- Log keystroke: this action is to gain the username and password that might be used to take down of the system
- Database log data modification: this action leads to loss of integrity of corporate data
- SCADA server manipulation: this action could lead to launching an attack to other system components within corporate network

SCADA system vulnerabilities: Earlier designs and implementations of SCADA systems were concern of system performance and reliability. However, recent attackers aim to compromise the integrity, confidentiality, authentication and availability of the critical infrastructure (Jonathan *et al*., 2010). In contrary to the current implementations, SCADA networks were isolated from the Web, corporate and other networks thus attackers could not penetrate the SCADA network (Jonathan *et al*., 2010). Recently, SCADA network began to embrace information technology features and liable to be connected to the outside world. SCADA network is no more a segregated network and with SCADA associated with other network particularly the internet, it is also, open to new threats. Consequently, SCADA systems become more vulnerable to cyber attacks. An assault against SCADA system could lead to physical damage, economic loss and even imperil environment and public safety. Subsequently, security of SCADA network has turned into a prime concern (Jonathan *et al*., 2010).

Generic vulnerabilities of SCADA system: A major difference in securing ICS and a typical computer system is in the ICS components that do not use standard Information Technology (IT) hardware or software. Custom ICS hardware and software have not been scrutinized like common computer products and refresh rates are typically much lower. This study lists major SCADA security weaknesses.

Multiple access point: When a SCADA system implementing interconnectivity of network which means the system is connected to corporate networks, business partner's network and/or any other networks, there will be multiple access point to any of these networks and including the SCADA network. Skilled attacker can exploit any of these connection and gain access to the SCADA physical network. A SCADA network is connected to the outside networks through a gateway, yet, it does not mean the only way out to the outside world is through that gateway. There might likewise be other unforeseen connections for example, telephone connections

(Jonathan *et al*., 2010). Hence, many of the gateways also do not include security protection mechanisms. Once the attackers reside in the network it could harm the security properties of the network. Usually attackers will gain access to the network through vulnerability on other networks such as corporate, vendors and/or customer's network. For example, the attackers use a remote access port used by vendor for maintenance to get inside the network.

The use open standards communication protocols: Most of currently implemented SCADA systems embraces Transmission Control Protocol/Internet Protocol (TCP/IP) and Ethernet communications and many have encapsulated their proprietary protocols in TCP/IP packets (Robles and Kim, 2010). Throughout the years, organizations began to transitioning from the utilization of proprietary to open international standard communication protocols due to economic and technical benefits, yet, this transitioning likewise expands vulnerabilities in the network (Jonathan *et al*., 2010). Moreover, a considerable lot of SCADA protocols does not bolster cryptography, so, if the attackers successfully barging in the network, the attackers could be eavesdropping on the network and obtain the confidential data and control commands and could later use this data to send false messages and impact the data integrity by altering the control commands.

Used of COTS hardware and software: The utilization of Commercial-On-The-Shelf (COTS)-based hardware and software is less expensive and diminish design time of the network. However, this can harm the SCADA network. COTS are not generally secure in light of the fact that they are not particularly designed for critical control systems.

Internal user careless: Attackers use insider to gain access into the network by exercising social engineering attacks. The common tactics include by convincing internal user to click on a URL in an email from a workstation that is connected to both to SCADA network and the internet. The attackers could spread malware or worms by using this tactic. Furthermore, infecting internal user laptops or removable media while the user is outside the SCADA network can also compromise the SCADA internal system when the devices get connected again to the SCADA network. Once the attacker has infiltrated the SCADA network, it will be possible for them to do any malicious attack to crash or disturb the organization process.

Specific security vulnerabilities of SCADA system: SCADA system as highlighted in Fig. 2 has been divided into several layers according to ISA99 security model. The network has been divided into 6 layers which is.

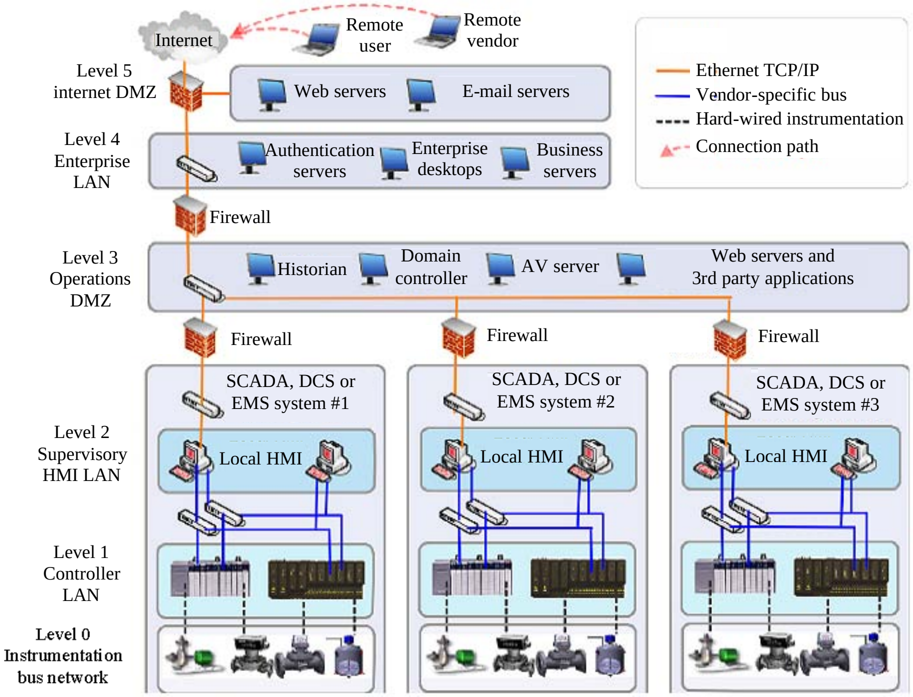

Fig. 2: Generic SCADA system (Acunetix, 2016)

Layer 0 (bus network): This layer is known as instrumentation bus network where operating equipment, instrumentation reside. This layer rarely contains any vulnerabilities, thus it is not a target for attackers.

Layer 1 (controller lan): This layer is known as a controller LAN which controls the communication between layer 0 (Bus Network) and Layer 2 (Supervisory LAN). PLC and RTU reside in this layer. It is also not a very popular target for an attacker.

Layer 2 (supervisory HMI LAN): This layer is where HMI resides and depending on what type of operating system the HMI and other devices at this layer use. Vulnerabilities type identified at this layer are related to operating systems and applications vulnerabilities (Miller and Rowe, 2012; Acunetix, 2016).

Layer 3 (operational DMZ): This layer typically is designed as an interface to Layer 2 (HMI Supervisory LAN) to acquire data and to share data with enterprise IT applications that reside in Layer 4 (Enterprise LAN). Almost all SCADA vendors use a design that places data historians, web servers, reporting systems and other back-end servers in an area that is both accessible from the SCADA networks as well as the enterprise IT networks (Miller and Rowe, 2012). The DMZ network layer is the most connected area in the SCADA system. Consequently, it has been identified to be the area that contains the highest number of vulnerabilities (Miller and Rowe, 2012; First.org., 2016). DMZ network layer is considered as the last line defense before any traffic hit the SCADA and industrial process control system. If this layer is compromised, SCADA system can easily be compromised as well, since, all the applications in this layer often are all authorized and trusted by the SCADA system (Miller and Rowe, 2012; First.org., 2016).

Layer 4 (enterprise LAN): This layer is an enterprise or corporate network which is designed to obtain information from operation DMZ. This layer typically contains the same vulnerabilities as operation DMZ. This layer is among a popular target to be compromised because it is easier to gain access to this layer. This is mainly due to the fact that the enterprise network is often connected to other outside networks, especially, the internet.

Layer 5 (internet DMZ): Layer 5 is internet DMZ. This layer also typically contains the same vulnerabilities as Layer 3 and 4.

MATERIALS AND METHODS

Vulnerability severity assessment methodology: The vulnerability severity assessment that is widely and commonly used is known as Common Vulnerability Scoring System (CVSS). As it name suggests, CVSS is used to access the severity of computer system security vulnerability. CVSS scores are calculated based on a formula that depend on several metrics that related to ease of exploit and the impact of exploit. CVSS allow vulnerabilities to be prioritized based on the score obtained. Scores in CVSS range from 0-10 with being the most severe or critical.

CVSS measure 3 areas of concern which is called Base metric, temporal metric and environment metric. National Infrastructure Advisory Council (NIAC) is the organization that founded CVSS. NIAC selected the Forum of Incident Response and Security Teams (FIRST) to become the custodian of CVSS for future development.

The CVSS assessment measures three areas of concern: base metrics for qualities intrinsic to vulnerability; temporal metrics for characteristics that evolve over the lifetime of vulnerability; environmental metrics for vulnerabilities that depend on a particular implementation or environment. A numerical score is generated for each of these metric groups.

Base metric: Base metric is the basic characteristic of vulnerabilities and this metric is mandatory. The explanation of Base Metric's values is given in Table 1.

Temporal metric: Temporal metric reflects characteristic of a vulnerability that may change over time. The explanation of Temporal metric's values is given in Table 2.

Environment metric: These metrics enable the analyst to customize the CVSS score depending on the importance of the affected IT asset to a user's organization, measured in terms of complementary/alternative security controls in place, confidentiality, integrity and availability. The explanation of environment metric's values is given in Table 3.

Table 1: Base metric explanation

Metric	Explanations	Possible values
Access vector	The more remote vulnerability can be exploited the higher the rating	Network: Can be exploit remotely Adjacent: Can be exploit remotely but limited to the same physical or logical network Local: Not bound to the network stack and the attacker's path is via. read/write/execute capabilities Physical: The vulnerable component must be touch physically to be exploited
Attack complexity	How difficult for an attacker to launch an attack	Low: Specialized condition does not exist and an attacker can expect repeatable success against the vulnerable component High: Specialized condition exist that make attacker must spend some amount of effort to exploit the vulnerable component
Privileges required	Level of privileges an attacker must possess before successfully exploiting the vulnerability	None: Does not need any privileges to successfully exploited the vulnerable component Low: Require basic privileges before can successfully exploit the vulnerable component High: Require administrative privileges before can successfully exploit
User interaction	Required user or other than attacker to participate in the successful attack	None: The vulnerable system can be exploited without interaction from any user Required: Successful exploitation of this vulnerability requires a user to take some action before the vulnerability can be exploited
Scope	Whether the vulnerable component may affect other component govern by other authorities or not	Unchanged: The successfully exploited vulnerable component may not impact other resources govern by other authorities Change: The successfully exploited vulnerable component may
Confidentiality, integrity, availability	Degrees of loss of the security properties	None: There is no loss of confidentiality, integrity and availability Low: There is some loss of confidentiality, integrity and availability High: There is total loss of confidentiality, integrity and availability

Table 2: Temporal metric explanation

Metric	Explanations	Possible values
Exploit code maturity	The existence of exploit code and its level of maturity	Not defined: This will not affect the score High: Functional autonomous code exists Functional: Functional exploit code is available Proof-of-concept: Proof-of-concept exploit code is available or an attack demo is not practical for most systems Unproven: No exploit code available, or is only theoretical
Remediation level	Level of remediation available	Not defined: This will not affect the score Unavailable: No remediation available or impossible to apply Workaround: Unofficial, non-vendor solution available Temporary: There is an official but temporary fix not practical for most systems Official: Complete vendor solution
Report confidence	The confidence that one has in the description of vulnerability	Not Defined: This will not affect the score Unknown: There is a report of impact of the vulnerability but the cause of the vulnerability is unknown Reasonable: Significant details are published, but researchers either do not have full confidence in the root cause or do not have access to source code to fully confirm all of the interactions that may lead to the result Confirmed: Detailed reports exist or functional reproduction is possible (functional exploits may provide this)

Table 3: Environment metric explanation

Metric values	Explanations
Not defined	Assigning this value to the metric will not influence the score It is a signal to the equation to skip this metric
Low	Loss of (Confidentiality-Integrity-Availability) is likely to have only a limited adverse effect on the organization or individuals associated with the organization (e.g., employees, customers)
Medium	Loss of (Confidentiality-Integrity-Availability) is likely to have a serious adverse effect on the organization or individuals associated with the organization (e.g., employees, customers)
High	Loss of (Confidentiality-Integrity-Availability) is likely to have a catastrophic adverse effect on the organization or individuals associated with the organization (e.g., employees, customers)

RESULTS AND DISCUSSION

SCADA vulnerability severity rating: Level 3 and 4 are layers where most of the vulnerabilities are found (Miller and Rowe, 2012). Therefore, most of the severity ratings were focusing at these layers. To compute severity for vulnerabilities, CVSS requires several metrics or characteristics of vulnerable component such as remote access configuration (serves as an access vector), level of difficulty for an attacker to launch an attack towards vulnerable component (serves as an attack complexity), level of privilege an attacker must possess before successfully exploiting the vulnerability (serves as privileges required) and several other metrics. However, the vulnerabilities discussed in this study are generic and do not belong to any specific SCADA system. Thus, the researcher cannot get the exact metric to compute vulnerability severity rating accurately.

In our study, we developed two case studies to compute the vulnerability severity in general SCADA system. The required metric values are based on the generally defined values for SCADA system. We designed the first case study to have only base metric, a mandatory metric to calculate the severity. No temporal metric is defined in the first case study. On the other hand, we used base and temporal metric in the second case study in other words, the value for base metrics in the second case study is similar to the value assigned to base metric in the first case study. Result of these case studies are recorded and analyzed. Afterwards, the vulnerabilities are prioritized. The prioritization enables ones to plan for proper vulnerability treatment if need be.

Case study 1: For the first case study, we restrict the metric to only base metric to represent a very minimal consideration. In layer 3 and 4, web server platform and database server are two commonly used and therefore, chosen to be the elaborated further in this case study. We define the value of every metric for these vulnerabilities based on different findings in the literatures. The severity rating for the web server platform vulnerability is shown in Table 4 and its explanation is as follows.

Table 4: Case study 1: severity rating of web server platform vulnerability

Base metric	Values
Access vector	Network
Attack complexity	Low
Privileges required	Low
User interaction	None
Scope	Changed
Confidentiality impact	High
Integrity impact	High
Availability impact	High
Severity score	9.9 (Critical)

Table 5: Case study 1: severity rating of database server vulnerability

Base metric	Values
Access vector	Network
Attack complexity	High
Privileges required	Low
User interaction	None
Scope	Changed
Confidentiality impact	High
Integrity impact	High
Availability impact	High
Severity score	8.5 (High)

Access vector: This vulnerability is bound to the network stack. Since, an attacker can remotely exploit web server over the network, the access vector should be the 'Network'. This is a nature to web server and can't easily be changed. However, all other base metrics are conceivable depending on the configuration of vulnerable components (Robert, 2002; Acunetix, 2016).

Attack complexity: The web server does not have any specified access conditions, i.e., for an attacker to exploit the vulnerability of a web server, he would not face much difficulty. For example, if a web server uses a default configuration, unutilized services will cause more ports to be left opened and thus, leaving more doors for attackers to exploit. This will set the attack complexity vector to 'Low' (Sommestad *et al*., 2013).

Privileges required: The privilege is set too 'Low' because attackers do not need to have an administrator privileges before he can exploit the web server.

User interaction: User interaction is defined as 'None', which means that the attacker can launch the attack on his own without any help from other resources. For example, no social engineering is needed to trick legitimate users such that the users have to click a link that, without user's knowledge, contains malicious script before he can exploit the web server.

Scope: Scope metric has 'Changed' value. This means that the attack will possibly affect other components or resources that are governed by other authorities. For example, if the attacker successfully exploits the web server, the database server will also be affected.

Confidentiality, integrity, availability impact: Confidentiality, integrity and availability are set too 'High' because the attack will cause a serious impact to these three security objectives. For example, the attack will lead to the disclosure of sensitive information. Also, attackers can modify the information and delete some of them.

From the defined metric values above, the severity score was computed at 9.9 which is then classified as 'Critical'. This is mainly due to the insecure configuration on the web server that leads to more opportunities for an attacker to easily exploit the web server and subsequently cause damage to the web server and other resources. This type of vulnerability should be given a high priority for vulnerability treatment.

The next vulnerability is database server. Its base metrics is shown in Table 5 and the severity rating for this vulnerability as follows.

Access vector: The access vector for this vulnerability is depending on the database server configuration. However, usually a database server is configured to have a remote access capability. Therefore, the value for this

Table 6: Case study 2: severity rating of web server platform vulnerability

Temporal metric	Values
Exploit code maturity	Proof-of-concept
Remediation level	Temporary fix
Report confidence	Reasonable
Severity score	9.2 (Critical)

Table 7: Case study 2: severity rating of database server vulnerability

Temporal metric	Values
Exploit code maturity	High
Remediation level	Official fix
Report confidence	Confirmed
Severity score	7.8 (High)

metric would be 'Network'. Local value is also possible when the user requires local access to the database server (Robert, 2002; Sommestad *et al*., 2013).

Attack complexity: Attack complexity is set to 'High' because in this case study we assumed that the attackers conduct comprehensive investigation to locate the database.

Privileges required: A privilege required is defined as 'Low' because the attacker must at least possesses basic user privileges to access the database.

User interaction: User interaction is set to 'None'.

Scope: Scope is defined as 'Changed'. This means a successful attack will give an impact to other resources that are governed by other authorities. Compromised database may also give impact to HMI and worsen the physical operation of SCADA network (Robert, 2002).

Confidentiality, integrity, availability impact: Confidentiality, integrity and availability are defined as 'High' because the attack will cause a serious impact to these security objectives. For example, an attacker is able to disclose, modify and even delete sensitive information inside the database that might lead to catastrophic failure in SCADA network.

From the metric values defined as above, the severity is classified as 'High'. This is due to insecure configuration of the database server. Also in this case, when the attack complexity is 'High', the severity rating for this vulnerability is lowered. This type of vulnerability should also be given an attention for vulnerability treatment.

Case study 2: In this case study, the same base metric values for both vulnerabilities in the first case study were used. However, there is an additional metric used, i.e., temporal metric. Temporal metric reflects characteristics of vulnerabilities that may change over time. For example, when there exists an exploit code, remediation and report confidence are sometimes available and sometimes not. Table 6 shows the severity rating for web server platform vulnerability where the temporal metric are as follows.

Exploit code maturity: Exploit code maturity metric for an individual vulnerability can range from 'Unproven' to 'High'. The methods to attack web server are well known and one of them is shellshock. Shellshock is defined as proof-of-concept (Ten *et al*., 2008), therefore, the metric for exploit code maturity is set to proof-of-concept.

Remediation level: Remediation level is depending on the organization. The metric value of the vulnerability can range from 'Unavailable' to 'Official Fix'. In this case study, we recommended a temporary fix for this vulnerability, i.e., web server vendor releases or recommends a temporary mitigation while the official fix is being developed.

Report confidence: Report confidence is set to 'Reasonable'. Based on the metric defined as above, the severity score is obtained at 9.2 which implies that the vulnerability is critical. This is because this vulnerability is still in a category of 'Critical' despite temporal metric values are added in. Therefore, an organization needs to consider further treatment for this vulnerability. Afterwards, the severity rating for database server and its temporal metric explanation are given in Table 7.

Exploit code maturity: Exploit code maturity metric for an individual vulnerability can range from unproven to high. Database server is a very popular target because they contain a lot of sensitive and valuable information of a SCADA system. There are many well-known attacks to compromise a database server. There is one tool that is very popular used to attack a database server called Sqlmap. It is an automated tool used to detect and exploit database server vulnerabilities and taking over of database servers. Therefore, exploit code maturity metric is set to high.

Remediation level: We assume that, remediation level is set to official fix which means the database server vendor has provided official patch for this vulnerability.

Report confidence: Report confidence is set to reasonable. This vulnerability is rated as 'High' with a scoring at 7.8. It can be interpreted that database server vulnerability is less threatening as compared to web server vulnerability. However, the score also implies that database server vulnerability has to be treated sooner. It can be seen that the final severity score is lower than that in the first case study. This is due to the added metrics, i.e., temporal metric values used in this case study. It

Table 8: Vulnerability prioritization from case study 1 and 2

Priority	Base metric	Base+Temporal metric
1	Web server platform Vulnerability (9.9/critical)	Web server platform Vulnerability (9.2/critical)
2	Database server Vulnerability (8.5/high)	Database server Vulnerability (7.8/high)

indicates that the availability of remediation, report confidence status and the maturity of exploit code are also important because they affect the final score.

Vulnerabilities prioritization: From these two case studies, we prioritize the vulnerabilities based on the scores of the severity rating in case study 1 and 2. The prioritization of vulnerabilities from both case studies is shown in Table 8.

Possible affected components: When a web server is compromised, the possibility of a database server to be affected is also high. This is because they are usually designed to research with each other. Typically, database server is connected to HMI. Operator usually uses HMI to control and monitor the remote location based on the information logged by historian (database) (Stouffer *et al*., 2011). In other words, a possibility that HMI would also be affected if the database is compromised is also high. As a consequence, this could lead to a situation where physical control of SCADA system is possible.

CONCLUSION

In this study, we discuss a formal method to assess vulnerability in a SCADA system. However, due to the fact that the vulnerabilities vary in different situation, we developed two case studies that represent different level of vulnerability in SCADA system. The results of the study indicate that vulnerabilities identification is essential and its severity rating is deem required for an organization to prioritize which vulnerability to undergo for treatment to ensure that SCADA network is kept secure at a highest level possible. The study also shows that the vulnerability severity might vary depending on the case studied. The case studies that we developed might help ones to understand the vulnerabilities and its characteristics before their severity can be rated. From the scores obtained from vulnerabilities severity rating, an organization knows what to do and which vulnerabilities must be mitigated first in order to secure their SCADA network.

REFERENCES

Acunetix, 2016. Web server security and database security. Acunetix Ltd., London, UK. http://www.acunetix.com/websitesecurity/webserver-security/

Extreme Network, 2017. Security vulnerability in Apache web server-Struts2 bundled with ridgeline 4.0. Extreme Networks, San Jose, California, USA. https://gtacknowledge.extremenetworks.com/articles/Solution/Security-vulnerability-in-Apache-web-server-Struts2-bundled-with-Ridgeline-4-0

First.org., 2016. Common vulnerability scoring system v3.0: Specification document. Manchester, New Hampshire, USA. https://www.first.org/cvss/specification-document.

Igure, V.M., S.A. Laughter and R.D. Williams, 2006. Security issues in SCADA networks. Comput. Secur., 25: 498-506.

Jonathan, P., CISSP, CAP. and PCIP., 2010. The dirty underbelly of SCADA and smart meters. Red Tiger Security, Houston, Texas. https://media.blackhat.com/bh-us-10/whitepapers/Pollet_Cummins/BlackHat-USA-2010-Pollet-Cummings-RTS-Electricity-for-Free-wp.pdf.

Miller, B. and D. Rowe, 2012. A survey SCADA of and critical infrastructure incidents. Proceedings of the 1st Annual Conference on Research in Information Technology, October 11-13, 2012, ACM, Calgary, Alberta, Canada, ISBN: 978-1-4503-1643-9, pp: 51-56.

Robert, N.C., 2002. Port 80: Apache HTTP daemon exploit: In support of the cyber defense initiative GCIH practical assignment v2.1, option 2. GIAC, London. https://www.giac.org/paper/gcih/361/port-80-apache-http-daemon-exploit/103818.

Robles, J. and T. Kim, 2010. Architecture of wireless supervisory control and data acquisition system. Adv. Comput. Intell. Man Mach. Syst. Cybern. Venezuela, 2: 241-244.

Robles, R.J., M.K. Choi, E.S. Cho, S.S. Kim and G.C. Park *et al*., 2008. Vulnerabilities in SCADA and critical infrastructure systems. Intl. J. Future Gener. Commun. Networking, 1: 99-104.

Schneider Electric, 2012. SCADA systems, telemetry and remote SCADA solutions. Rueil-Malmaison, France. http://www.schneider-electric.com/solutions/ww/EN/med/20340568/application/pdf/1485_se-whitepaper-letter-scadaoverview-v005.pdf.

Sommestad, T., M. Ekstedt and H. Holm, 2013. The cyber security modeling language: A tool for assessing the vulnerability of enterprise system architectures. IEEE. Syst. J., 7: 363-373.

Stouffer, K., J. Falco and K. Scarfone, 2011. Guide to Industrial Control Systems (ICS) security. MBA Thesis, US Department of Commerce, National Institute of Standards and Technology, Gaithersburg, Maryland, USA.

Ten, C.W., C.C. Liu and G. Manimaran, 2008. Vulnerability assessment of cybersecurity for SCADA systems. IEEE. Trans. Power Syst., 23: 1836-1846.