Identifying and Scoring Vulnerability in SCADA Environments

Parves Kamal, Abdullah Abuhussein
Department of Information Systems
St. Cloud State University
St. Cloud, MN
{pkamal, aabuhussein}@stcloudstate.edu

Sajjan Shiva
Computer Science Department
The University of Memphis
Memphis, USA
sshiva@memphis.edu

***Abstract*—Supervisory Control and Data Acquisition (SCADA) systems form a critical component to industries such as national power grids, manufacturing automation, nuclear power production and more. By interacting with control machines and providing real-time support to monitor, gather, and record data, SCADA systems show major impact in industrial environments. Along with the uncountable benefits of SCADA systems, inconceivable risks have raised. Moreover, SCADA operators, production staff and sometimes systems experts have no or little knowledge when applying security due diligence. In this paper, we systematically review SCADA security based on different aspects (i.e. SCADA components, vulnerability, severity, impact, etc.). Our goal is to provide an all-inclusive reference for future SCADA users and researchers. We also use a time-based heuristic approach to evaluate vulnerabilities and show the importance of the evaluation. We aim to establish a fundamental level of security due diligence to ensure SCADA risks are well-comprehended and managed.**

Keywords—Supervisory Control and Data Acquisition (SCADA) security; critical infrastructure security; SCADA; risk assessment; vulnerability scoring

I. INTRODUCTION

In the past, Supervisory Control and Data Acquisition (SCADA) systems were merely used in oil, gas, and power distribution systems. Today SCADA systems are almost everywhere; in telecommunication, pharmaceutical and manufacturing industries [1]. Smooth operation of a SCADA system is vital, not only for the smooth operation of these business sectors, but also for the environment and human life as any disturbance could cause catastrophic damage [2].

Since the interconnection of SCADA systems to the internet, SCADA systems have become an easy target for the attacker. This is attributed to the wider attack surface and more attack vectors in addition to the lack of security features in place. Over the last decade, a significant bulk of the reported attacks was SCADA pertinent. Among these attacks are 1) the shutdown of Browns Ferry Nuclear Plant in Alabama due to a DDoS attack [3], 2) intrusion in water treatment facilities in Harrisburg, Pennsylvania [4], and 3) the shutdown of the train signal system by a virus at CSX Corp, which managed to stop many of the train lines in the eastern part of the USA [5]. Perhaps the most notorious of all was the Stuxnet malware worm attack which quaked the world as major global energy companies fell victim of it [6]. Many more attacks are being reported on SCADA systems regularly with a recent report of more than 50 new attacks which are similar to Stuxnet [7].

The recent increase of attacks on SCADA environments is a result of the combination of multiple causes such as the unprotected devices, and communication protocols that were not built with security in mind. In addition, the transition from wired to wireless communication as well as radio communication technologies added salt to injury as it added a new sphere of vulnerabilities for the cyber attackers to exploit. Although securing wired and wireless networks have been extensively studied and their security solutions have advanced well since then. But, use of different communication protocols (e.g. modbus, profibus, DNP3, etc.) and different network architectures for SCADA brings in additional vulnerabilities. In this paper, we present the state-of-the-art in SCADA systems security by implementing a taxonomy for SCADA security. We classify security issues of SCADA environments based on their presence on SCADA components, vulnerabilities exploited, and potential attacks. We further classify attacks based on different aspects (i.e. target, severity, impact, medium, motivation, and type). Additionally, we exemplify each attack by a real-world incident from literature. Finally, we demonstrate how to use the classification to gauge a vulnerability score. We aim to improve the security of SCADA environments by enabling SCADA stakeholders to better comprehend and evaluate vulnerabilities in SCADA environments. This paper is structured as follows. Section 2 shortly introduces related work. Section 3 presents SCADA anatomy and security concerns of each SCADA component. A taxonomy of security in SCADA and an example of how to trace taxonomy is discussed in Sections 4 and 5 correspondingly. In Sections 6 and 7, we explain the Vulnerability Scoring approach and demonstrate an example. We discuss future work and conclusion in Section 8.

II. RELATED WORK

Several researchers have identified, classified, assessed and illustrated SCADA environments in multiple ways [8]-[14]. Various methods to identify the risk of SCADA systems have evolved. Hierarchical holographic modeling (HHM) [37], [38] was used to identify the risk of all conceivable risk sources of SCADA system in railroad sector [39]. Risk filtering, ranking, and management method (RFRM) use HHM model to identify risks and then ranks them in order to prioritize them [40]. Inoperability input–output modeling (IIM) quantifies the

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

economic loss in large-scale infrastructure like SCADA system to over IP (Internet Protocol) network [41]. Nozick et al. focused on capturing the uncertainty in network links in critical systems using Markov and semi-Markov processes [42].

This work complements the bulk of efforts devoted to survey SCADA environments security. We have carried out an in-depth analysis of the recent issues related to the security and privacy of SCADA environments reported in published literature between 2000 and 2017. We have exemplified each vulnerability from well-known vulnerability databases (e.g. ICS-CERT [43], CVE [44], CVSS [45], vuldb [46], etc.). We have classified SCADA security terminology to enable better understanding. We have extrapolated published SCADA security incidents and showed how they map to our classification. Finally, we use a time-based heuristic approach to evaluate vulnerabilities and show the importance of the assessment.

III. Anatomy and Security of SCADA

Any complex SCADA environment can be reduced to simplest components that are connected through communication protocols. The four components we use to classify SCADA environments are presented in Fig. 1. In the following sub-sections, we will go over the SCADA components and explore the security threats in each.

A. Data Storage (Historian)

Within the SCADA network data storage is required for future or ongoing analytics. Data acquired from the SCADA network is used to adjust the current processes and decipher if the current process is within specification. A data historian is a software that can run on the supervisory computer or on a dedicated machine. Historians are database systems that store real-time data from the SCADA network. Data security is of major concern in SCADA systems and hence we classify and identify the attacks related to data storage as follows:

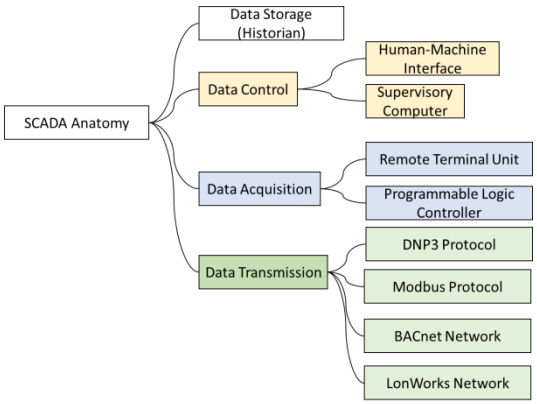

Fig. 1. SCADA devices.

- **Buffer Overflow:** Historian systems such as the King View HistorySvr were at risk of Buffer Overflow errors. Released in 2011, King View Manufacture Well Intech released a vulnerability notice that their main system could be remotely attacked via TCP port 777, causing a heap-based buffer overflow [15].
- **SQL Injection:** The core structure of a historian system is a database. Many data historians came equipped with a web interface, whether it be for administration or usage. These websites could be used to execute un-sanitized Structured Query Language (SQL) inputs causing damage to the historian or the entire system.
- **Cross-Site Scripting:** Many historian systems now use some sort of web interface. Either for remote administration or for a system UI. These sites are susceptible to web page vulnerabilities. Released in 2011, Billy Rios and Terry McCorkle made a report on a cross-site scripting vulnerability to the GE Intelligent Historian Web administrator [16]. The vulnerability was a result of a lack of server-side validation parameters. This attack was able to be remotely exploited due to many organizations allowing the web interface to be outward facing.
- **Memory Corruption:** This is an attack that can be executed internally or remotely on a data historian depending on access. In 2012, a Zero-Day Initiative report alerted that the General Electric (GE) Intelligent historian could be remotely attacked via port 14000/TCP which caused the archive to crash. This, in turn, could allow for more code to be executed and loss of data [17]. This is a single example of a memory corruption attack but many others are possible.
- **Denial of Service (DoS):** Schneider Electric's Vijeo Historian during late 2011 had multiple vulnerabilities released by Kuang-Chun Hung. Topping the list was a DoS vulnerability that was caused by a linked third-party ActiveX control [18]. For the attack to happen, the historian would have to voluntarily interact. This could have been avoided by not using a third-party source within the critical software, such as ActiveX, which has slowly made its way out due to security concerns.
- **Directory Traversal:** Following the same release as above, Video was also hit with a directory transversal vulnerability [19]. An attacker that had gained unauthorized access to the network could openly read files through HTTP requests without prior authentication or social engineering.

B. Data Control

To control data, SCADA collects and send control commands to the field connected devices. SCADA systems use supervisory computers, which are installed with a unique Human–Machine Interface (HMI), to be responsible for communicating with the field connection controllers (i.e. Remote Terminal Units (RTU) and programmable logic controllers (PLC)) and include the HMI software running on operator workstations. Data control is a critical SCADA component, so we further classify and identify attacks related to it based on the following sub-components:

1) Supervisory Computer:

This is the core of the SCADA systems. In smaller SCADA systems, the supervisory computer may be composed of a

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

single personal Computer (PC) in which the HMI is a part. However, in larger SCADA systems, the master station may include several HMI components hosted on a single or multiple client computers or multiple servers for higher quality data acquisition, distributed software applications, and multiple disaster recovery sites. Much like typical computers, supervisory computers can be easily attacked. The following are potential supervisory computer security threats.

- **Physical Attacks:** Like other ordinary computers, the supervisory computers regularly face physical attacks because of human errors or intruders. According to the "2014 Cyber Security Intelligence Index" from IBM, 95 percent of all security incidents involved human error. Different from other attacks, physical attacks usually happen when the employees ignore the security practices or the Safety Manual of the companies [20].
- **Denial of Service (DoS):** This is one of the most common attacks on computers. Due to the outdated operating system, supervisory computers or the networks connecting them may become unavailable to intended users causing temporarily or indefinitely disrupting services. Defensive responses to DoS attacks typically involve the use of a combination of attack detection, traffic classification, and response tools like Intrusion prevention systems (IPS), which are effective if the attacks have signatures associated with them [21].

2) HMI (Human-Machine Interface)

HMI systems are the operating window of supervisory systems. They present plant information to the operating personnel graphically in the form of mimic graphs displaying schematic representations of the plant being controlled, system alarm, and event logging pages. The HMI is linked to the SCADA supervisory computer to provide live data to create the graphs. In many installations, the HMI is the graphical user interface for the operator, collects all data from external devices, creates reports, performs alarming, sends notifications, etc. HMIs are vulnerable to security threats like other software. The primary threats are as follows [22]:

- **Memory Corruption:** Memory corruption occurs in HMI when the content of a memory location is unintentionally modified due to programming errors causing a memory safety violation. MICROSYS PROMOTIC Memory Corruption is an example of memory corruption vulnerability [47]. MICROSYS PROMOTIC [48] is a Microsoft Windows-based SCADA HMI software programming suite. MICROSYS has produced a new version to mitigate this vulnerability. More details related to memory corruption in SCADA is available in [49].
- **Buffer Overflow:** In September 2012, a buffer overflow error on HMI was released [50]. It's reported that a specially crafted packet sent to the PLC's HMI listening service triggered a remotely exploitable buffer overflow condition. In November 2016, a critical buffer overflow vulnerability has been identified in HMI known as UCanCode [51]. This vulnerability affects one of the UCanCode functions. No countermeasures were released [52].
- **Account Hijacking:** HMIs are prone to many vulnerabilities like insecure default among others. Under normal circumstances, people tend not to pay attention to defaults set by the system which in turn brings risks of Hijacking by hackers. HMI may lose its functionality to provide live data to produce graphs or produce modified graphs. The sensitive data collected by HMIs is also a target for hackers. Siemens SIMATIC STEP 7 DLL is an HMI hijacking example [53].
- **SQL Injection:** SQL injection is a code injection technique that allows attackers to spoof identity, tamper with existing data collected by HMI causing repudiation issues such as voiding transactions, or allowing the complete disclosure of all data on the system including sensitive data, destroy the data or make it otherwise unavailable. In December 2016, National vulnerability database (NVD) [71] released SQL injection warning about a vulnerability in Ecava IntegraXor [54] that allows remote attackers to execute arbitrary SQL commands via unspecified vectors [55].
- **Running on System-level:** There are many levels of access control that operating system grants to users in order to protect system files and functions from being altered accidentally or intentionally. System level allows any person to run the applications with an administrator privilege. It could be very dangerous if HMI running on system level is hacked. It will become the tunnel of sending malware to the system which will be controlled by hackers as the administrators. A Successful exploitation of such vulnerability was reported in July 2016. This vulnerability allows an authenticated user on the system to modify the configuration of the CIMPLICITY service of GE Proficy HMI SCADA and launch any executable on the system as a service [56].
- **Denial of Service (DoS):** In November 2016, a vulnerability was found in HMI UCanCode that affects unknown function. The report in [57] mentioned that vulnerability is triggered when manipulation with an unknown input causing a denial of service vulnerability. No published countermeasure was found other than replacing the product.

C. Data Acquisition

Data acquisition in SCADA environments is the processes of collecting information to document or analyze some phenomenon. Data Acquisition begins at the RTU or PLC level which involves parameter readings by sensors that are transmitted to the SCADA supervisory system. Frequent attacks on Data acquisition devices like RTU and PLC makes it one of the important devices to protect in SCADA systems. In this section, we present security threats in SCADA environments using both RTU and PLC.

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

1) RTU (Remote Terminal Unit)

RTU is defined as a communication device within the SCADA system and is located at the remote substation. It gathers data from field devices in memory until the information is requested. RTU relies on common standards protocols (Modbus, IEC 60870-5-101/103/104, DNP3, IEC 60870-6-ICCP, IEC 61850, etc.) used in SCADA systems to communicate and control the lower level devices which make them prone to vulnerabilities and therefore security threats as follows:

- **Message modification:** If an RTU supports an insecure SCADA protocol that can be straightforwardly attacked and used to control or damage the connected objects, it will be almost impossible for a lower level prevention mechanism to protect the RTU since it has no way of differentiating between authentic and unauthentic SCADA communications. SubSTATION Server Telegyr 8979 Master Vulnerabilities in July 2014 is an example of this type [58]. By sending specially crafted invalid RTU messages to the Telegyr 8979 master, a buffer overflow can occur, resulting in a denial of service (DoS).
- **Spoofing:** In December 2015, a vulnerability was discovered in 1000 CCU and RTU GMS devices. These devices are products Pacom [59]. The vulnerability allows remote attackers to spoof the controller-to-base data stream by leveraging improper use of cryptography [60].
- **Sniffing:** A sniffing attack is when the attacker tries to gain access to unauthorized data. Lack of authentication mechanism makes it possible for the attacker to read all sorts of RTU information (e.g. status, location, vendor, software, etc.) of SCADA devices like sensors, actuators [24].
- **Insider threats:** Disgruntled insiders have been the main source of computer crime since they have knowledge of and access to internal systems. Insiders include employees, business partners, and vendors. Insiders may not necessarily be malicious, but accidental mistakes can have the same consequences as malicious attacks. The well-known GhostExodus of 2011 shows an example of this type of threat [61] where an insider was able to leverage his position as a night security guard to gain physical access to control systems and manipulate those systems.
- **Stack-based buffer overflow:** Recently Risk Based Security, the Open Security Foundation (OSF) [62] has reported remote code injection vulnerabilities in Modbus serial driver which will allow an attacker to perform stack-based buffer overflow attacks which will, in turn, give an attacker control of any PLC system [25].
- **Privilege escalation:** This is where an attacker gains some level of access which usually is user level access, attempts to increase their rights by attacking the access control configuration. Emerson Roc 800 Remote Terminal Unit Process Management Privilege Escalation in late 2014 is an example of this type [63], [64]. Any attacker who exploits the vulnerability could disable the device, compromise the device integrity, and remotely execute code on the target system.

2) PLC (Programmable Logic Controller)

PLC is a small industrial computer used in factories originally designed to replace relay logic of a process control system and has evolved into a controller having the functionality of a process controller. PLC is likewise not immune to security threats. PLC logic, hardware, application layer, communication and the operating system on which PLC runs can be exploited as follows:

- **Operating system threats:** Every RTU or PLC controller on the market has a commercial operating system in it (e.g. Microware OS-9, VxWorks, etc.). Although these operating systems are not famous like Linux, they are vulnerable to attacks because operating systems increase attack surface. Stuxnet [7] is a malicious computer worm, first identified in 2010, that targets industrial computer systems and was responsible for causing substantial damage to Iran's nuclear program. Stuxnet specifically targets PLCs and allow separating nuclear material.
- **Insider Attack:** Similar to RTU, PLC is prone to insider threats. Insider attacks remain one of the top security concerns for critical infrastructures. Many dimensions of the problem remain unsolved as to what would be an effective solution to tackle the insider threat.
- **Modifying Ladder Logic:** Ladder Logic is a method to document the design and construction of relay racks as used in manufacturing and process control. Ladder logic is used to develop PLCs used in industrial control applications. By gaining privilege to Ladder Logic, an attacker can modify ladder logic (PLC programs) and impact the functionality of the program. An exploitation of the vulnerability could allow any network user to interact with the process control and change the ladder logic. A similar vulnerability was reported in 2015 in Phoenix Contact Software's ProConOs and MultiProg applications [65]. Although vendor wrote these applications without authentication intentionally, the impact can be hazardous in critical systems.
- **Cross-Site Request Forgery (CSRF):** CSRF is an attack where the attacker forces an authenticated user to execute authorized command on a web application. One such threat reported on compromised PLC web server allows an attacker to compromise the integrity and availability of the PLC device [66].
- **Hijacking Web Session:** Web session hijacking can cause theft or modification of data. Due to lack of entropy in generating random number and attacker can hijack PLC web session without authentication as in [67].

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

D. Data Transmissions: Protocol and Networks

Data in transmission is particularly vulnerable to sniffing and modification and it is one of the most critical aspects of SCADA systems which are relatively insecure in nature. Because of its criticality, we classified threats based on their common SCADA communication protocols, and network that is in use in manufacturing as follows:

1) DNP3 Protocol

Distributed network protocol or DNP3 is the most widely used automation protocol in manufacturing and building automation SCADA environments. More than 75% of North American electric utilities currently use DNP3 for industrial control applications [68]. Because of its wide usages and inherent weak security mechanism, the DNP3 protocol is being heavily targeted by the attackers. These attacks are:

- **DDOS Attack:** Distributed denial of service attack (DDoS) occurs when multiple systems try to attack one system to eat up its bandwidth or resources. Due to its lack of authentication security properties, DNP3 protocol is vulnerable to DDOS attack [26]. Elipse SCADA DNP3 Denial of Service in Dec 2014 is an example of this vulnerability that could be exploited remotely [69].
- **Man in the Middle Attack:** Man in the middle (MITM) attack as the name implies is an attack where the attacker tries to intercept the connection by positioning himself in the middle of the connection between sender and receiver as the connection passes by. DNP3 protocol is also vulnerable to such attack due to its weak encryption [27].
- **Data Modification/interception Attack:** Due to its weak or no encryption mechanism DNP3 protocol is susceptible to data modification or interception attack. The Data interception attack occurs when an unauthorized party (service, software, computer system) gains access to data, whereas in data modification attack not only get access but also tries to make changes to the data [10].
- **Baseline Response Replay Attack:** As DNP3 protocol typically has no encryption mechanism in place, an attacker can simulate message from master to its outstation devices after observing DNP3 message patterns which to enable the attacker to take control of the devices [70].
- **Network Reconnaissance:** Due to unencrypted communication within DNP3 messages, an attacker can get information about network topologies, the device functions, and data in memory [70].
- **Reset Function Attack:** Unprotected DNP3 protocol is subject to reset function attack where an attacker sends reset function code 1 to the device which initiates a restart of outstation devices and it can create potential outage of services if the device restarts to inconsistent state [70].
- **Destination Address Alteration Attack:** DNP3 protocol message is prone to modification destination address alteration attacks where an attacker can change the destination address and reroute the messages to devices that can cause unexpected results. Also, an attacker can send malformed packets to the entire outstation device by sending packets to broadcast address 0xFFFF [70].
- **Outstation write Attack:** By sending function code 2 in modified DNP3 messages enables an attacker to write to the outstation devices which can corrupt information stored in the device memory [70].
- **Clear Object Attack:** In clear object attack an attacker sends function code 9 or 10 to the outstation devices to either hold or clear the object's data of the device. This attack can clear the critical data from the outstation devices or make them unstable which effectively destabilize the overall SCADA system [70].
- **Configuration Capture Attack:** In configuration capture attack the attacker sends a message with a fifth bit in the second byte of IIN set, which effectively instruct the outstation devices that the current configuration file is invalid requiring new configuration file to be sent out from the master device. The attacker then tries to modify the configuration file that is being sent out from master to the outstation devices with their own configuration to control the devices [70].

2) Modbus Protocol

Like the DNP3 protocol it's widely used in industrial automation too, and it's used for serial communication between SCADA components, for example communication between RTU devices with historian, etc. [30]. Though it's relatively easy to configure, lack of authentication and authorization mechanism makes this protocol vulnerable to several attacks like:

- **Unauthorized Command Execution:** It is possible to send forged Modbus message for executing an arbitrary unauthorized command to be executed by the masters and slaves because of the lack of authentication [28].
- **Replay Attack:** Replay attacks occur when an attacker retransmit validated message repeated or delayed maliciously. Due to the lack of authorization security mechanism attacker can retransmit validated Modbus message [28].
- **Stack-based buffer overflow:** Recently, the Open Security Foundation (OSF) has reported remote code injection vulnerabilities in Modbus serial driver which will allow an attacker to perform stack-based buffer overflow attacks which will, in turn, give an attacker control of any PLC system [29].
- **DDOS attack:** An attacker can send forged messages from the master exploiting Modbus lack of authorization to the RTU's to drain its resources and take it down as in [29].

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

3) BACnet Network

It is an open communication protocol used for control automation and application that is used for HVAC (heating, ventilating, and air-conditioning) systems. It's been widely used protocol and because of its open standard nature. BACnet has serious security issues like lack of authentication and authorization features leaving it vulnerable to snooping, spoofing, DOS attacks, etc.

- **Snooping Attack:** The snooping attack is like sniffing attack where the attacker tries to gain access to unauthorized data. Lack of authentication mechanism makes it possible for the attacker to read the status, location, vendor, software, etc. of SCADA devices like sensors, actuators [24].
- **Disabling Router Connection:** An attacker can send Disconnect-Connection-To-Network message to break a communication path due to the lack of authentication of users and devices [24].
- **Write-to-commandable Property:** Due to the lack of writing property source authentication, an attacker can force changes to the property value of some devices to perform harmful attacks like stopping the building control system, turning on or off critical equipment [24].
- **DDOS attack:** Lack of authentication in BACnet protocol enables Attacker to perform network and application layer DDOS attacks by sending forged Broadcast-as-SADR confirmed service request message to consume the resources of the receiving devices. Also, attacker can send repeated forged Router-Busy-To-Network message to break communication and maintain the interruption [24].

4) LonWorks Network

LonWorks is a local operating network platform for controlling Heating, ventilation and air conditioning (HVAC) applications. Like its rivals BACnet, it is proprietary network built into Lontalk protocol. It has weak encryption key making it susceptible to DDOS and brute force attacks.

- **Brute-force Attack:** Here, the attacker tries every possible combination of keys to guess the password or encrypted data. In LonWork networks the authentication is done via a pre-shared 48-bit key which is weak in nature and makes it possible for the attacker to brute force as in [23].
- **DDOS Attack:** Since LonWorks uses lontalk protocol, which has authentication vulnerabilities. An attacker can send a flood of authentication messages for which the receiver needs to generate a response and in process resources will be consumed to perform DDOS attack [31].
- **Disclosure of Information:** The data in LonWorks are sent via clear text hence disclosure of information cannot be avoided as an attacker can easily intercept the data [23].
- **Spoofing:** In LonWorks, authentication occurs at sender's end only, which makes it prone to a spoofing attack [23].

IV. TAXONOMY OF SECURITY IN SCADA

The threats on SCADA environments can be classified into different general categories. In this section, we provide our security and privacy taxonomy of SCADA Environment, as outlined in Fig. 2. Classifications in Fig. 2 are further discussed in the following subsections.

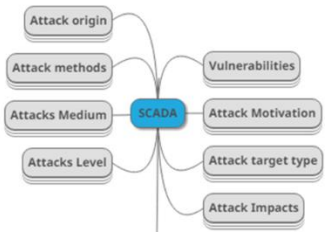

Fig. 2. Security taxonomy of SCADA environment.

We define threat on SCADA as the potential for transgressions against SCADA environment components that creates negative or harmful consequences while an attack is an action taken against SCADA environment with the intention of doing harm. According to the definitions, our SCADA classification is categorized into several main branches, namely, 1) Attack origin, 2) Attack Methods, 3) Attack Medium, 4) Attack level, 5) Attack Severity, 6) Vulnerabilities Exploited, 7) Threat Motivation, 8) Attack Target, 9) Attack Type, and 10) Attack Impacts. A detailed description of each category along with its importance is discussed as follows:

A. Attack Origin

Identifying the attack origin helps in quarantining source of the threat as well as planning for proper mitigation. We categories attacks based on its source as follows:

- **Local:** are threats against SCADA components by an attacker who already has physical access to one or more of the components from within the SCADA environment.
- **Remote:** are threats that do not require the attacker to be near the victim system rather exploiting bugs in system remotely via malware or compromising software or hardware flaws.

B. Attack Methods

This category provides great insight on how the threats are enabled. In other words, how an attack is performed. Also, it helps to better assess the risk factor associated with each possible attack which will be shown later in the assessment section. In our taxonomy, we categories SCADA security issues based on the methods used by the attacker to launch the attack as follows (see Fig. 3).

- **Malware:** Attacks that are initiated against SCADA environments by the malware like a virus, Trojan horse, worms, botnets, etc.

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

- **Social Engineering:** Launching an attack by exploiting human nature and tricking them to gain access to compromise the SCADA system.
- **User Level Compromise:** Attackers also compromise the system by performing user-level attacks such as stolen credentials, user account hijacking, etc. preventing it from being used for its intended purpose.

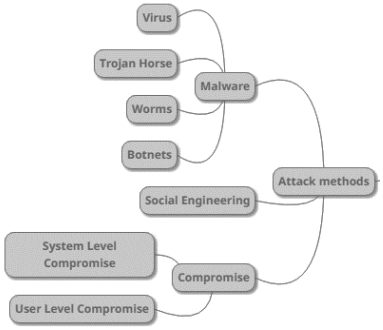

Fig. 3. Threat methods.

C. Attack Medium

Knowing the medium that can be used to carry out attacks provides a more granular overview of the threat and hence, we categorize attacks as outlined in Fig. 4.

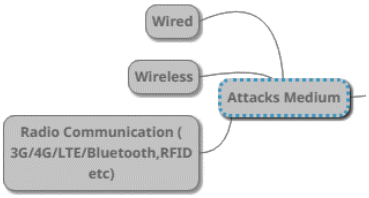

Fig. 4. Attack medium.

- **Wired**: The threats that exploit wired network vulnerabilities fall under this category.
- **Wireless**: This is when the exploited vulnerabilities are in wireless technologies.
- **Radio Communication**: The threats that exploit radio communication technologies like 3G/4G/LTE, RFID, Bluetooth, etc.

D. Threat Level

Attacks often can be part of the imminent big attack. Knowing the attack level helps administrators to lock down possible attacks before it spirals out of control. Therefore, we further categorise attacks as in Fig. 5.

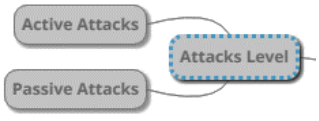

Fig. 5. Attack level.

- **Passive Attacks**: Attackers often perform attacks on a system to compromise another part of the system or gather information for launching active attacks on the system. Attacks like packet interception, snooping, sniffing, Man-in-the-middle attacks are some of the examples of passive attacks.
- **Active Attacks**: These are the attacks that involve compromising a system with the help of the information gathered in passive attacks. Attacks like DDOS, brute force attack, buffer overflow and SQL injection, etc. are active attacks examples.

E. Vulnerability Exploited

Attackers often exploit known or common weaknesses to attack SCADA systems. Knowing vulnerabilities responsible for certain types of attacks helps system administrator to lock down those vulnerabilities before getting exploited. Our taxonomy further Categorizes attacks based on SCADA Vulnerabilities into three groups here, including configuration, specification, and implementation (Fig. 6).

- **Configuration:** Configuring the SCADA system with no security in mind can expose vulnerabilities as following:

1) Default Configuration: Applications are often installed with default settings that attackers can employ to attack them. This is particularly an issue with third party software where an attacker has easy access to a copy of the same application or framework.

2) Weak Configuration: SCADA can be often configured with minimum or weak configuration settings, leaving many loopholes to exploit.

- **Specification:** Vulnerabilities of this type means designing and implementing security solutions that have low-security measures which can overly expose vulnerabilities.

1) Weak Authentication or Authorization Methods: means implementing authentication solution that is 1) easy to brute force, 2) sends authentication as clear text format or 3) authorization solution that does not authorize both sender and the receiver are some examples of weak Authentication/Authorization methods in this category.

2) Weak encryption: means implementing weak encryption methods that can be easily brute-forced.

3) Outdated systems: Choosing to design or implement a solution or a system that has no product support can leave the SCADA environment vulnerable to new dimensions of attacks throughout its lifecycle.

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

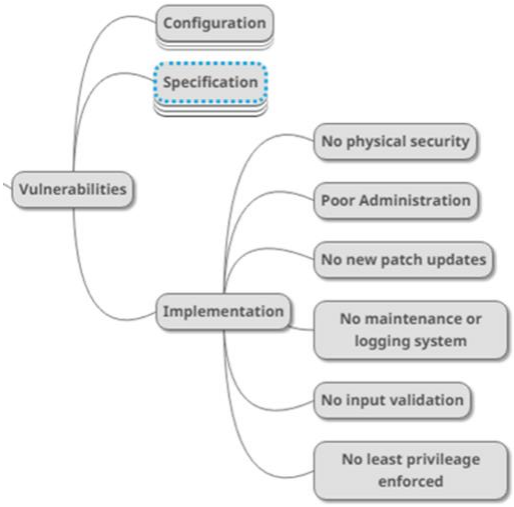

Fig. 6. SCADA vulnerabilities.

- **Implementation:** means vulnerabilities that can arise from poor implementation in following ways:

1) No physical security: Physical attacks usually take place when there are no or few physical security methods. This category represents this implementation vulnerability.

2) Poor administration: Administration is very important in the system. Poor administration makes the SCADA more vulnerable to attacks. With poor administration, the system may be disrupted by intrusion programs. Any unsuitable physical manipulation will make the system crash.

3) Lack of maintenance and patching: Patching is a never-ending task. Every system or software needs to be updated from time to time because there is no perfect system. Each system has its own vulnerabilities and they can be used by attackers. Many SCADA administrators do not maintain/patch systems because of the time it takes to maintain, and because the process is daunting and resource-heavy. Absence or lack of patching may result in that vulnerabilities are ending up on the path to exploitation.

4) Absence of logging systems: In many SCADA systems, there is a lack of central logging system to log any security events in the logging system database. This issue can turn into a vulnerability as absence of logging allows attackers to roam the SCADA environment undetected.

5) No input validation: Input validation is the accurate testing of input that is supplied by others. Without input validation, a person cannot know exactly who or what is giving input to process. Incorrect input validation could lead to security issues like information disclosure, buffer overflow, injection attacks, memory leakage, etc.

6) No least privilege enforced: This vulnerability means that every program and every user of SCADA system or its data can obtain or change information in unwanted ways due to unnecessary permissions to users beyond the scope/time of the necessary rights.

F. Attack Motivation

Attacks are often carried out by targeting groups or sometimes insiders. Identifying the attack motivation often dictates the attack severity. In our taxonomy, we categorize attacks based on motivation as targeted and non-targeted attacks (see Fig. 7).

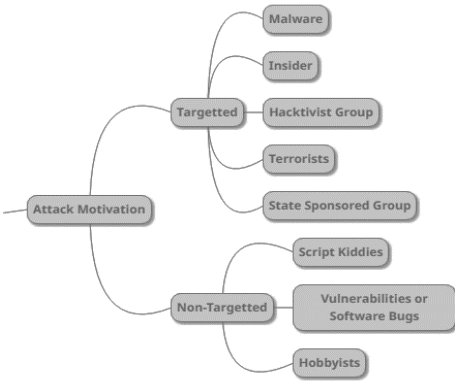

Fig. 7. Attacks motivation.

- **Targeted:** In a targeted attack, the attacker is fully aware of the extent of the attack and who is it affecting. These attacks are normally associated with malicious intent. The following are types of targeted attacks:

1) Malware: Malware is sent out by an attack group or individual in hopes that it will seek and attach to as many systems as possible. Malware can be developed to target a specific architecture or system such as SCADA networks.

2) Insider: An insider is a person or program working within the SCADA network or administration. Insiders actively and knowingly allow or implement attack or data loss within the SCADA environments.

3) Hacktivist Group: This is a group or single individual that acts to promote a political agenda through the use of attacks on an endpoint of SCADA environments.

4) Terrorist: Terrorism in the world of cyber security is an attack on an individual or group in an effort to cause harm in the name of a political regime.

5) State Sponsored Groups: State-Sponsored Attacks are attacks based on a group or individual that is being backed by a government or entity, seeking to do harm for the funding governments benefit.

- **Non-Targeted:** means attacks that are more often with unintended outcomes.

1) Script Kiddies: This is an attacker using the readily available code to execute an attack. These attacks can often lead to unknown outcomes of the original attacker.

2) Software Bugs: These are unknown vulnerabilities in SCADA systems that may produce security issues later.

3) Hobbyists: A general hobbyist often performs an attack with no malicious intent. Instead often executing an attack to find out if the attack is possible and to denote and alert.

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

G. Attack Target Type

Attacks often target system vulnerabilities that can often be the starting point of an attack. Knowing and understanding the attack target types helps in identifying the assets that need protection. We categorized target as outlined in Fig. 8.

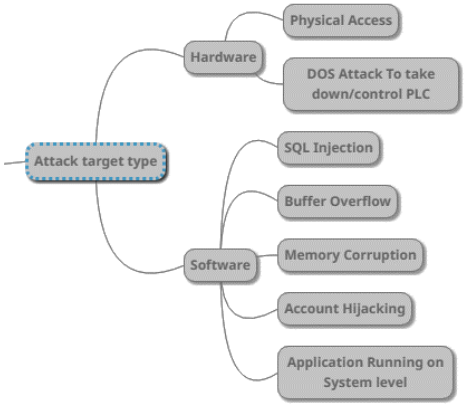

Fig. 8. Attack target type.

- **Hardware:** are attacks targeting SCADA hardware components. Those can be as follows:

1) Physical Access: An attacker can access data information while in front of the device. Whether it is physically retrieved data from a hard drive or splicing into a LAN network. Physical access can be executed in many forms.

2) DoS Attack: A DoS attack can be executed in order to take down a data acquisition device such as the PLC or RTU. When hit with a DoS attack. This can make the device malfunction and go into a shutdown state or interrupt device operations for a period of time.

- **Software:** are attacks targeting SCADA hardware components. Those can be as follows:

1) SQL Injection: is a software attack that can affect any program database when improper techniques are used in properly sanitizing SQL executions.

2) Buffer Overflow: is a software attack that can have an impact on a range of SCADA devices from the Data Acquisition devices to the data historian. It occurs when a SCADA software component overruns the buffer's boundary and overwrites adjacent memory locations that are assigned to some other program or operating system. This may cause a range of issues like changing program behavior, incorrect results, and/or system crash.

3) Memory Corruption: Memory corruption and memory corruption bugs can be used to force software to crash or behave in a way that is not originally intended. Modern programs today are less susceptible to memory corruption; however, it is still possible.

4) Account Hijacking: Account hijacking can be executed through multiple means like brute force or social engineering. Hijacking an account on SCADA systems allows an attacker to gain access to parts of the system that are otherwise inaccessible.

5) System Level Privilege: An application when installed or run can be executed at various permission levels. Often referred to as a system, administrator, and user levels. An application that runs at the administrator or system level has other can be an open the door to further attacks on both the parent host and connected hosts.

H. Attack Impacts

Attacks impacts identify the strong effect or influence of the attack on someone or something. The criticality of SCADA systems makes this classification important when quantifying vulnerabilities. Attack impact can range from very high like life threatening attacks to very low (e.g. non-critical system outage). Again, different SCADA environments have different requirements, assets, exposure to the public, criticality and tolerances to security risks thus, attack impact is a user-defined attribute. In our taxonomy, we further categorize security based on the attack impact as in Fig. 9.

- **Life Risk:** an impact in SCADA systems that can cause catastrophic results.
- **Monetary value:** a financial loss impact in the form of loss of money or decrease in financial value.
- **Reputation or Brand damage:** is causing harm or loss resulting from damages to a firm or a person reputation.
- **Disclosure of Information:** means making information accessible to interested and affected parties such as attackers or competitors.

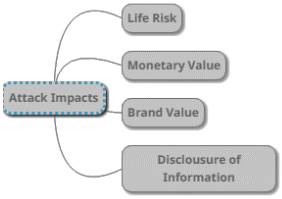

Fig. 9. Attack impacts.

V. Possible Attack Examples

To show how both taxonomies in Fig. 1 and 2 can be utilized, we provide examples of SCADA real-world vulnerability along with security classifications in Table 1. Columns 1 and 2 categorize SCADA components according to their functionality. The following columns depict the rest of the classifications from taxonomy in Fig. 2. The last column shows a reference of a real-world published vulnerability.

According to our SCADA anatomy taxonomy in Fig. 1, Historian service which works as data storage is vulnerable to buffer overflow attacks because of the no boundary protection mechanism implemented and the attacker can exploit the vulnerability remotely utilizing malware. This compromises the critical function as well as the confidentiality of the sensitive data. An example of this attack is KingView Data Historian, where a vulnerability through the network allowed for a special packet to cause a buffer overflow and crash the

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

system [16]. According to our taxonomy classification, this is an active attack on the SCADA system and the impact can be life threatening.

Also, according to our classification, data control is handled by two SCADA components. Those are 1) supervisory computer (hardware), and 2) HMI (software). Supervisory computers are just like any computer in their vulnerability to attacks such as DoS attack. DoS attacks in this case are due to outdated OS, and can be exploited remotely as in [21].

The two vulnerability examples discussed in this section are highlighted in Table 1.

VI. SCADA Vulnerability Scoring

Methods of Scoring vulnerability have been extensively researched. Many research and commercial attempts tried to quantify the vulnerability based on different criteria (e.g. severity, Common Vulnerability Exposure (CVE) [35], etc.). Although these attempts have their strengths, limitations are also present. Some of the limitations worth mentioning are subjectivity, ambiguity and contextual references [35]. In this paper, we evaluate vulnerabilities using vulnerability scoring technique from Tipwire IP360 [35]. In this vulnerability scoring method, the score is computed based on the age of the vulnerability and the skill required to successfully exploit the vulnerability. Temporal vulnerability score is important because it provides a different metric for a vulnerability at a time. In other words, old vulnerability gives an indicator that it is now easier for attackers to exploit and at the same time patches are definitely available. Skill to exploit a vulnerability is crucial because it represents the degree of difficulty associated with effectively exploiting a vulnerability [36].

To compute vulnerability score, we used the attacks shown in Table 1 and extracted their information from CVSS (common vulnerability scoring system) database [69]. The vulnerability scoring has the following components that are user-defined based on the organization security requirements:

- $\boldsymbol{t_n}$ depicts the number of days since vulnerability *n* has been reported in any major vulnerability reporting resources (e.g. CVSS, NVD, ICS-CERT, Vuldb, etc.).
- $\boldsymbol{r_n}$ represents the risk score of vulnerability *n* which is a user defined value that depends on the weight of the risk according to the user.
- $\boldsymbol{s_n}$ is the skillset required to successfully exploit the vulnerability *n* on the system, s. This parameter is also be user-defined based on the system's assets, and tolerance to risks.

After determining the value of $\boldsymbol{t_n}$, $\boldsymbol{r_n}$, $\boldsymbol{S_n}$, the equation for calculating the vulnerability score $\boldsymbol{v_n}$ is as follows:

$$v_n = \sqrt{t_n}\frac{r_n!}{s_n^2} \quad (1)$$

The vulnerability scoring equation above uses a heuristic approach to calculate the risk associated with the vulnerabilities instead of considering the risk as an absolute measurement of threat caused by exploiting a vulnerability or more. This approach rather takes a time-based approach which is independent of the system and unlike any other scoring systems.

To make our score relevant to the context of the vulnerabilities found in CVSS (common vulnerability scoring system) we must further classify our risk based on the attack impacts shown earlier in Fig. 9 and as shown in Table 2.

As in Table 2, the risk score is higher for remote level compromise than the local level access. We also must determine the skill level needed by the attacker to successfully exploit a vulnerability in SCADA environment. We introduce Skills label table (Table 3) to determine the skill set.

TABLE I. SCADA Vulnerability Examples as Mapped to Taxonomy Classification

Category	Component	Attacks	Vulnerabilities	methods	Origin	Attacks Impact	Ref
Data Storage	Historian	Buffer Overflow	No boundary protection	Malware, user Compromise	Remote	Life Risk	[16]
Data Control	Supervisory Computer	DoS	Outdated OS	Malware, User Compromise	Remote	Life Risk/Brand Value loss	[21]
	Human Machine Interface	Memory Corruption	Boundary Check Error	Malware, User Compromise	Local	Life Risk	[48]
Data Acquisition	Remote Terminal Unit	message modification	Software vulnerabilities.	Virus, Malware	local/remote	Loss of information	[58]
	Programmable Logic Controller	modifying ladder logic programs	backdoor	malware	local	Life Risk	[65]
Data Transmission (Protocols/Networks)	DNP3 Protocol	DDOS	No authorization	Malware, User Compromise	Remote	Life Risk/Brand Value	[32]
	Modbus protocol	Unauthorized Command Execution	Lack of Authentication	Malware	Remote	Life Risk	[33]
	BACnet Protocol	Snooping	No authentication	User Level User Compromise	Local	Info Disclosure	[34]
	LonWorks Network	Brute-force attacks	Weak Encryption	Malware/ User Compromise	Remote	Life Risks, brand value	[23]

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

TABLE II. RISK LABEL SCORES

Label	Description	Risk *r*
Exposure	Disclosure of the Information	0
Local Availability	Compromising Availability by local attacks (e.g. DDoS)	1
Local Access	Compromising user level access by local attacks	2
Local Privilege	Compromising Admin level privilege by local attacks	3
Remote Availability	Compromising Availability by remote attacks (e.g. DDoS)	4
Remote Access	Compromising user level access by remote attacks	5
Remote Privilege	Compromising Admin level privilege by remote attacks	6

TABLE III. SKILLS LABEL SCORE

Label	Description	Skill, *s*
Very Easy	Automated exploits or toolkit available	1
Easy	The exploit code or kits out there to take advantage of	2
Medium	The exploit is available, but require modification or testing	3
Hard	Only proof of concept available, requires the development of own module or exploitation	4
Very Hard	The full details of the exploits is not available, usually report with no proof of concept	5
Unrealistic	No known exploits or details available, e.g. zero-day vulnerabilities or hype based vulnerabilities	6

As noticed from Table 3, exploits that are readily available require relatively fewer expertise exploits whereas zero-day exploits or exploits with no details require highly skilled attacker to successfully exploit.

VII. VULNERABILITY SCORE: EMPIRICAL EXAMPLE

In this section, we exemplify the use of the vulnerability scoring metric. We used vulnerabilities from Table 1. To calculate the vulnerability score of Advantech Web Access Cross-Site Scripting on HMI (CVE-2013-2299) basic assumption has been made that all the software in SCADA system is running in user level privilege and the least privilege policy has been maintained.

To compute $\boldsymbol{t_n}$ for vulnerability CVE-2013-2299, we need the vulnerability record as follows to determine the number of days since the vulnerability was first published:

CVE ID: CVE-2013-2299
Published Date: 13th August 2013
Today's date: 23rd Feb 2017 (Example)
***t*:** 1290 days**,** Thus, $\sqrt{t}$ = 35.91

The risk score label $\boldsymbol{r_n}$ determines the exposure level of the vulnerabilities if successfully exploited. The risk score is classified as label 6 from Table 2. According to the CVE database, exploiting this vulnerability grants remote access to the attacker. So, we calculate a risk score $\boldsymbol{r_n}$ as:

Exploits: CVE-2013-2299
Risk Class, r= 5. Thus, $\boldsymbol{r}!$ = 120

Since the exploit is available, but require modification we consider the skill level of our vulnerability as medium (S=3), so we calculate the skill score, s as

Exploit Availability: Publicly available exploit, so, $\boldsymbol{s^2}$= 9

Therefore, vulnerability score $v_n = \sqrt{t_n} x \frac{r_n!}{s_n^2}$ is:

$$\boldsymbol{v_n} = 35.91 x \frac{120}{9} = 476 \quad (2)$$

As we can see, the vulnerability score reflects the metric at the time it was calculated. Let's say the vulnerability was discovered on 1st of January 2016 and the exploit was not publicly available and it would require highly skilled attacker to exploit it. In this case, to calculate the vulnerability score:

$\boldsymbol{t_n}$= 419 days, $\boldsymbol{r_n}$ = 5 (Remote Accessibility), $\boldsymbol{t_n}$=5 (Very Hard Skills Set)

$$v_n = 20.46 x \frac{120}{25} = 98.25 \quad (3)$$

From the two examples above, we see a significant drop in vulnerability score because of the relatively new vulnerability and the unavailability of the exploit code. To calculate the overall risk score of a SCADA environment, first all vulnerabilities need to be identified using the taxonomy proposed in this paper. Then, the score for each of the vulnerabilities needs to be calculated using (1). Finally, all the vulnerability scores need to be summed up to calculate total overall vulnerability score, v_s as shown below.

$$v_s = v_1 + v_2 + \cdots v_n \quad (4)$$

VIII. CONCLUSION

We surveyed and discussed the state of the art in SCADA security and introduced a taxonomy of security issues that highlights the various aspects of this area. This taxonomy defines SCADA security risks according to their properties and wherein the system they manifest. We also demonstrated the usability of a vulnerability scoring method to assess vulnerabilities based on their age and attacker's required skill. This work is still in progress. Our next goal is to continue investigating security metrics and implement a framework of metrics to provide better and customized SCADA security evaluation. To the best of our knowledge, we believe that this work provides a current comprehensive reference to industry and research community on the security of SCADA environments.

REFERENCES

[1] A. Daneels and W. Salter, "What is SCADA?", International Conference on Accelerator and Large Experimental Physics Control Systems, pp. 339-343.

[2] Patel S, Tantalean R, Ralston P, Graham J. Supervisory control and data acquisition remote terminal unit testbed. Intelligent Systems Research Laboratory technical report

[3] Nicholson A, Webber S, Dyer S, Patel T, Janicke H. SCADA security in the light of cyber-warfare. Comput Secur 2012;31(4):418– 36TR-ISRL-05-01, Department of Computer Engineering and Computer Science. Louisville, Kentucky: University of Louisville, 20

[4] Guan J, Graham J, Hieb J. A digraph model for risk identification and management in SCADA systems. 2011 IEEE international conference on intelligence and security informatics, IEEE CP., 2011, p. 150–5

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

[5] Miller B, Rowe D. A survey SCADA of and critical infrastructure incidents. In: Proceedings of the 1st annual conference on research in information technology. ACM; 2012.

[6] The Real Story of Stuxnet. (2009). IEEE Spectrum: Technology, Engineering, and Science News. Retrieved from http://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet

[7] Phil Muncaster, Stuxnet-like attacks beckon as 50 new SCADA threats discovered 21st Apr. 2011, http://www.v3.co.uk/v3-UK/news/2045556/stuxnet-attacks-beckon-scada-threatsdiscovered

[8] P. Eden, P. Burnap, A. Blyth, K. Jones, H. Soulsby, and Y. Cherdantseva, "A Forensic Taxonomy of SCADA Systems and Approach to Incident Response," 3rd International Symposium for ICS & SCADA Cyber Security Research 2015, 2015.

[9] B. Miller, B. Young, A survey of SCADA and critical infrastructure incidents, Conference on Information Technology Education (SIGITE/RIIT), pp. 1-6, Canada , October 2012

[10] B. Zhu, A. Joseph and S. Sastry, "A Taxonomy of Cyber Attacks on SCADA Systems," 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, Dalian, 2011, pp. 380-388.

[11] E. E. Miciolino, G. Bernieri, F. Pascucci and R. Setola, "Communications network analysis in an SCADA system testbed under cyber-attacks," 2015 23rd Telecommunications Forum Telfor (TELFOR), Belgrade, 2015, pp. 341-344.

[12] R. S. Ramachandruni and P. Poornachandran, "Detecting the network attack vectors on SCADA systems," 2015 International Conference on Advances in Computing, Communications, and Informatics (ICACCI), Kochi, 2015, pp. 707-712.

[13] D. Krauß and C. Thomalla, "Ontology-based detection of cyber-attacks to SCADA-systems in critical infrastructures," 2016 Sixth International Conference on Digital Information and Communication Technology and its Applications (DICTAP), Konya, 2016, pp. 70-73.

[14] E. Bou-Harb, "Passive inference of attacks on SCADA communication protocols," 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, 2016, pp. 1-6.

[15] US-CERT/NIST, "Vulnerability Summary for CVE-2011-0406," in National Vulnerability Database, 2011. [Online]. Available: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2011-0406.

[16] ICS-CERT, "Advisory (ICSA-11-243-02)," in Industrial Control Systems Cyber Emergency Response Team, 2011. [Online]. Available: https://ics-cert.us-cert.gov/advisories/ICSA-11-243-02.

[17] ICS-CERT, "Advisory (ICSA-12-032-01)," in industrial control systems cyber emergency response team, 2012. [Online]. Available: https://ics-cert.us-cert.gov/advisories/ICSA-12-032-01.

[18] US-CERT/NIST, "Vulnerability Summary for CVE-2011-4033," in National Vulnerability Database, 2011. [Online]. Available: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2011-4033.

[19] US-CERT/NIST, "Vulnerability Summary for CVE-2011-4036," in National Vulnerability Database, 2011. [Online]. Available: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2011-4036.

[20] Howarth, F. (2016). The Role of Human Error in Successful Security Attacks. Retrieved December 03, 2016, from https://securityintelligence.com/the-role-of-human-error-in-successful-security-attacks/.

[21] Loukas, G.; Oke, G. (September 2010) [August 2009]. "Protection Against Denial of Service Attacks: A Survey". Comput. J. 53 (7): 1020–1037.

[22] Aquino-Santos, Raul (30 November 2010). Emerging Technologies in Wireless Ad-hoc Networks: Applications and Future Development. IGI Global. pp. 43–. ISBN 978-1-60960-029-7.

[23] W. Granzer, W. Kastner, G. Neugschwandtner, F. Praus. "Security in Networked Building Automation Systems", Proceedings of IEEE International Workshop on Factory Communication Systems, pp. 283-292, 2006.

[24] D. G. Holmberg, "BACnet Wide Area Network Security Threat Assessment", Technical report, National Institute of Standards and Technology, 2003.

[25] Three SCADA Vulnerabilities Disclosed. (2016). Riskbasedsecurity.com. Retrieved 4 December 2016, from https://www.riskbasedsecurity.com/2013/05/three-scada-vulnerabilities-disclosed/

[26] Wanying, Q., Weimin, W., Surong, Z., Yan, Z.: The study of security issues for the industrial control systems communication protocols. In: JIMET 2015 (2015)

[27] S. East, J. Butts, M. Papa, and S. Shenoi, A Taxonomy of Attacks on the DNP3 Protocol. New York: Springer, vol. 311/2009, pp. 67–81, 1868-4238, IFIP Advances in Information and Communication Technology

[28] I. Nai Fovino, A. Carcano, M. Masera, A. Trombetta, Design and implementation of a secure Modbus protocol, in: C. Palmer, S. Shenoi (Eds.), Critical Infrastructure Protection III, Springer, Heidelberg, Germany, 2009, pp. 83–96.

[29] Three SCADA Vulnerabilities Disclosed. (2016). Riskbasedsecurity.com. Retreived from https://www.riskbasedsecurity.com/2013/05/three-scada-vulnerabilities-disclosed/

[30] I. Nai Fovino, A. Carcano, M. Masera, A. Trombetta, Design and implementation of a secure Modbus protocol, in C. Palmer, S. Shenoi (Eds.), Critical Infrastructure Protection III, Springer, Heidelberg, Germany, 2009, pp. 83–96

[31] A. D. Wood and J. A. Stankovic, "Denial of Service in Sensor Networks", IEEE Computer, vol. 35, no. 10, pp. 54–62, 2002

[32] DoS Vulnerability Found in MatrikonOPC Server for DNP3 SecurityWeek.Com. (2016). Securityweek.com. Retreived from http://www.securityweek.com/dos-vulnerability-found-matrikonopc-server-dnp3

[33] TCP MODBUS - Unauthorized Read Request: Attack Signature - Symantec Corp.. (2016). Symantec.com. Retreived from https://www.symantec.com/security_response/attacksignatures/detail.jsp?asid=20674

[34] Gordy, F. et al. (2016). Network Security Threats | ControlTrends. Controltrends.org. Retreived from http://controltrends.org/category/building-automation-and-integration/network-security-threats/

[35] Tripwire, *Tripwire Vulnerability Scoring System* Retrieved from: http://www.tripwire.com/it-resources/tripwire-vulnerability-

[36] Farahmand, F., Navathe, S. B., Sharp, G. P., & Enslow, P. H. (2005). A management perspective on risk of security threats to information systems. Information Technology and Management, 6(2–3), 203–225.

[37] Whitman, M., & Mattord, H. (2005). Principles of information security (2nd ed.). Boston: Course Technology.

[38] Y. Y. Haimes, "Hierarchical Holographic Modeling," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 11, No. 9, pp. 606-617, 1981.

[39] C.G. Chittester, Y.Y. Haimes, "Risks of Terrorism to Information Technology and to Critical Interdependent Infrastructures," Journal of Homeland Security and Emergency Management, Vol.1, Issue 4, 2004, article 402.

[40] Haimes, Y. Y., Kaplan, S., & Lambert, J. H. (2002). Risk filtering, ranking, and management framework using hierarchical holographic modeling. Risk Analysis, 22(2), 381–395

[41] Haimes, Y. Y., & Chittester, C. G. (2005). A roadmap for quantifying the efficacy of risk management of information security and interdependent SCADA systems

[42] Nozick, L. K., Turnquist, M. A., Jones, D. A., Davis, J. R., & Lawton, C. R. (2005). Assessing the performance of interdependent infrastructures and optimizing investments. International Journal of Critical Infrastructures, 1(2-3), 144–154.

[43] ICS-CERT |. (2017). Ics-cert.us-cert.gov. retrieved from https://ics-cert.us-cert.gov/

[44] Common Vulnerabilities and Exposures, retrieved from https://cve.mitre.org/

[45] common Vulnerability Scoring System Support v2 retrieved from: https://nvd.nist.gov/cvss.cfm

[46] Vulnerability Database. (2017). Vuldb.com. Retrieved from https://vuldb.com/

978-1-5386-2823-2/17/$31.00 ©2017 IEEE

[47] MICROSYS PROMOTIC Memory Corruption Vulnerability | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-16-026-01

[48] PROMOTIC SCADA/HMI system. (2017). Promotic.eu. Retrieved from http://www.promotic.eu/en/index.htm

[49] Bellettini, C., & Rrushi, J. (2008, March). Combating memory corruption attacks on scada devices. In International Conference on Critical Infrastructure Protection (pp. 141-156). Springer US.

[50] PLC HMI Buffer Overflow. (2017). Tools.cisco.com. Retrieved from https://tools.cisco.com/security/center/viewIpsSignature.x?signatureId=35986&signatureSubId=0

[51] Visual C++ and VC++, with MFC, HMI and CAD, GIS, UML, SCADA, Simulation, Real-time, Graphics, Component, ActiveX Control, OCX, Diagram, Report Print, FlowChart, Source Code, Tutorial, Example, library, Drawing Component, Workflow, Electronic Form, BPM. (2017). Ucancode.net. Retrieved from http://www.ucancode.net/

[52] HMI UCanCode AddDWordUserProperty buffer overflow. (2017). Vuldb.com. Retrieved from https://vuldb.com/?id.93838

[53] Siemens SIMATIC STEP 7 DLL Vulnerability | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-12-205-02

[54] Pure Web SCADA with HTML5, CSS3 & SVG - Ecava IGX Web SCADA. (2017). Ecava IGX Web SCADA. Retrieved from https://www.integraxor.com/

[55] NVD - Detail . (2017). Web.nvd.nist.gov. Retrieved from https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2016-2299

[56] GE Proficy HMI SCADA CIMPLICITY Privilege Management Vulnerability | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-16-194-02

[57] HMI UCanCode denial of service. (2017). Vuldb.com. Retrieved from https://vuldb.com/?id.93839

[58] SubSTATION Server Telegyr 8979 Master Vulnerabilities | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-14-196-01

[59] Business Security Products - Management Systems, Access Control. (2017). Business & Commercial IP Security Systems | Pacom. Retrieved from http://www.pacom.com/products/

[60] Pacom 1000 CCU GMS System Cryptographic Implementation Vulnerabilities | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-15-337-03

[61] Leali, N. (2017). Lessons From an Insider Attack on SCADA Systems. blogs@Cisco - Cisco Blogs. Retrieved from http://blogs.cisco.com/security/lessons_from_an_insider_attack_on_scada_systems

[62] Open Information Security Foundation. (2016). Open Information Security Foundation. Retrieved 3 March 2017, from https://oisf.net/

[63] Emerson Roc 800 Remote Terminal Unit Process Management privilege escalation. (2017). Vuldb.com. Retrieved from https://vuldb.com/?id.73134

[64] Emerson ROC800 Multiple Vulnerabilities (Update B) | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-13-259-01B

[65] Phoenix Contact Software ProConOs and MultiProg Authentication Vulnerability | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-15-013-03

[66] NVD - Detail . (2017). Web.nvd.nist.gov. Retrieved 3 March 2017, from https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-2249

[67] NVD - Detail . (2017). Web.nvd.nist.gov. Retrieved 3 March 2017, from https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-2251.

[68] East, S., Butts, J., Papa, M., & Shenoi, S. (2009, March). A Taxonomy of Attacks on the DNP3 Protocol. In International Conference on Critical Infrastructure Protection (pp. 67-81). Springer Berlin Heidelberg.

[69] Elipse SCADA DNP3 Denial of Service | ICS-CERT. (2017). Ics-cert.us-cert.gov. Retrieved from https://ics-cert.us-cert.gov/advisories/ICSA-14-303-02.

[70] S. East, J. Butts, M. Papa, S. Shenoi, A Taxonomy of Attacks on the DNP3 Protocol., New York:Springer, vol. 311/2009, pp. 67-81.

[71] NVD - CVSS . (2017). Nvd.nist.gov. Retrieved from https://nvd.nist.gov/cvss.cfm.

978-1-5386-2823-2/17/$31.00 ©2017 IEEE