**GROUP No: 12  
  
Project Title: Anomaly Detection within Cyber Forensics in SCADA and ICS Systems**

**Members:**

**Ayush Shah, Gurkirat Sarabjeet Singh Nagpal, Atharva Marathe and Sarvesh Yenarkar**

**Contents:**

1. **Introduction and Abstract………………………………………… (2)**
2. **Cyber Forensics in Industrial Control Systems…………….(4)**
3. **Project Requirements………………………………………………. (5)**
4. **Functional Architecture……………………………………………. (7)**
5. **Subsystem Description……………………………………………. (10)**
6. **Project Management……………………………………………….. (17)**
7. **References…………………………………………………………….. (19)**

**01) Introduction and Abstract:**

Due to the constantly changing behavior of cyber-attacks, reactive approaches are desirable to detect and prevent malicious actors from gaining access to networks. Firewalls and intrusion detection and prevention systems (IDPS) are a line of defense in identifying and stopping suspicious internet traffic. When a suspicious event occurs, these devices generate a log file containing details of what preprogrammed rules were violated and how it was handled. Such log files contain details of the event, e.g. source and destination IP addresses, port numbers, and protocols, but not the packet and data that led to the event. Of interest is cyber/digital forensics of logged events to understand their origin and magnitude. Suspicious events include both malicious and non-malicious activities, e.g. misconfigured routers; however, each event is logged and to find malicious events for further analysis, one must search through all logged suspicious events. Although advances have been made in applying text mining and advanced analytics to cyber log data analysis, c.f. Suh-Lee, Breier and Branisova´ 5and Villa et al, the characteristics of cyber logs results in much manual analysis for interpretation and response. When considering log data, cyber analysts rely on manual sorting and experiential knowledge to find possible threats in logged events to further investigate. Thus, cyber security is heavily experiential based and uses the innate ability of humans to process large amounts of complex data similarly experience is critical and novice analysts might miss intrusions and events that a veteran analyst would not. Additionally, cyber intrusion detection is asymmetrical in nature whereby an attacker can focus on only one threat approach while a defender (cyber analyst) must constantly protect all systems and prepare for many different types of attacks, vulnerabilities and threats. Although system administrators and cyber analysts manually handle log data, this is becoming increasingly infeasible due to the big data nature of cyber traffic (unstructured, high volume and high velocity).

Normal behavior for cyber networks is generally not well defined and changes over time, resulting in high false positive detection rates. Additionally, since firewall log events are the result of network abnormalities, one is thus necessarily interested in detecting the anomalies within the anomalies. Related research, c.f. Lazarevic et al, Denning, Garcıa-Teodoro et al, Grimaila et al, Moore et al, Dube et al, 23 Shilland,24 Shen et al,25 Stewart et al has focused on anomaly detection at the device/software level, with little exploration into anomaly detection in the log files generated from the preexisting devices or software.

For analysis, data were used from a large scale distributed network with regional data nodes much like the Microsoft Cyber Defense Center, the Verizon Network Operations Center, or AT&T Global Network Operations Center. Currently, data is analyzed from enterprise-wide networks, which rely on a series of firewalls and IDPS to identify and stop intrusions. These devices, when triggered, generate a log file containing details of how it handled each incident, such as the source and destination IP addresses, port numbers, protocols, bytes transferred, etc. However, due to the wide variety of devices adding observations to the log, the data can be highly variable.

In operation, analysts employ an experiential approach whereby large log files are manually sorted to find anomalies to further investigate; this process is conceptualized in Figure below. However, due to the large size of the network and quantity of users, the data is of significant volume and emerging at high velocity; thus representative of a big data problem. Currently, analysts inspect numerous potential incidents on a daily basis, but have neither the time nor the resources available to analyze all incidents contained in the logs.

A SCADA system is an Industrial Control System (ICS) that operates public and private industrial processes including critical infrastructures. Such systems are becoming increasingly dependent on information and communication technologies and being connected to the Internet. This poses new challenges related to cyber security while allowing improved flexibility and ease of use of the systems. Being less protected and more sensitive to software updates (or malware protection updates) than a typical office environment makes additional security measures in ICS necessary. Also, ease of exposure to cyber attacks once the physical levels of security is breached (e.g. insider attacks) requires a new look at how to protect these critical environments. For example, TCP sequence prediction attacks can be difficult to recognize with existing techniques. To our knowledge, there is no SCADA-specific IDS that has successfully detected this kind of attacks.

**Abstract:**

Supervisory Control and Data Acquisition (SCADA) systems that operate our critical infrastructures are subject to increased cyber attacks. Due to the use of request-response communication in polling, SCADA traffic exhibits stable and predictable communication patterns. This paper provides a timing-based anomaly detection system that uses the statistical attributes of the communication patterns. This system is validated with three datasets, one generated from real devices and two from emulated networks, and is shown to have a False Positive Rate (FPR) under 1.4%. The tests are performed in the context of three different attack scenarios, which involve valid messages so they cannot be detected by whitelisting mechanisms. The detection accuracy and timing performance are adequate for all the attack scenarios in request-response communications. With other interaction patterns (i.e. spontaneous communications), we found instead that 2 out of 3 attacks are detected.

The information security vulnerabilities of ICS have been studied extensively, and the vulnerable nature of these systems is well-known. However, in the case of a security incident (e.g. system failure, security breach, or denial of service attack), it is important to understand what the digital forensics consequences of such incidents are, what procedures or protocols are needed to be used during an investigation, what tools and techniques are appropriate to be used by an investigator, and where the forensic data can be collected from and how.

**02) CYBER FORENSICS IN INDUSTRIAL CONTROL SYSTEMS**

**1) UNB Net flow:**

FLOW-BASED ANOMALY DETECTION Flow-based anomaly detection centers around the concept of the network flow. A flow record is a summarized indicator that a certain network flow took place and that two network end points have communicated with each other at some time in the past. A flow record typically contains the IP network addresses of the two hosts, network ports, network protocol, amount of data that was sent as part of this connection, the time when the flow occurred as well as a few miscellaneous flags. Flow-based approaches rely heavily on the ability of network devices to generate flow information. A typical anomaly detection system using flow information would be implemented as depicted in Fig. 1. An anomaly detection component would sit right behind the router to collect all flows going in / out of the network for analyzing.

**2)** **Android Malware**

Triggering the vulnerability First the adversary crafts and delivers the exploit to the victim to target a specific vulnerability known to the adversary (Step 1 ). The vulnerability is in general a memory corruption bug; the exploit is typically sent to a victim from a webpage or a document attachment from an email

**Code Reuse Shell code (ROP)**

To prevent untrusted data being executed as code, modern processors provide Data Execution Prevention (DEP) to restrict code from being run from data pages. To support JIT compilation however, DEP can be toggled by the program itself.

* Stage1: Shell code
* Stage2: Payload

The Stage1 shell code and Stage2 payload are different in size, design and function, primarily due to the operational constraints on the Stage1 shell code.

|  |  |
| --- | --- |
|  |  |

**3) Cyber forensics in SCADA /ICS systems:**

Computer forensics is the practice of collecting, analyzing and reporting on digital information in a way that is legally admissible. It can be used in the detection and prevention of crime and in any dispute where evidence is stored digitally [13]. “Traditional digital forensics is performed through static analysis of data preserved on permanent storage media. Not all data needed to understand the state of [an] examined system exists in non-volatile memory. Live analysis uses [the] running system to obtain volatile data for deeper understanding of events going on” [14]. As discussed the first problem in achieving cyber forensics for SCADA systems is that such systems are critical and cannot generally be powered off for acquisition. Additionally it is more likely that the information is generally volatile and any forensic evidence would potentially be lost if the device was powered off or interrupted. This remainder of this section looks at existing perspectives on SCADA forensics as well as the main differences between SCADA and enterprise forensics.

|  |  |
| --- | --- |
|  |  |
| Digital Forensic process | SCADA incident response and Forensic Process |

**03) Project Requirements**

The following requirements are derived from an objectives tree by taking in consideration the mission goals and expectations. The requirements will be categorized as mandatory and desirable requirements. These categories are further divided and the requirements are classified as performance requirements which are functional requirements with an associated performance measure and non-functional requirements.

**3.1 Mandatory Performance Requirements**

M.P.1 Phase 1- Identification and Preparation: Identify the potential sources of evidence, including the systems, the network and connected devices.

M.P.2 Phase 2- Identifying data sources: Identify the type of systems to be investigated including; operating system, manufacturer, serial numbers and model of PLCs, and network design and implementation

M.P.3 Phase 3- Volatility Assessment, Contamination Impact Analysis and Preservation, Prioritizing and Collection: Assess the volatility of the identified resource immediately after identification in order to drive the priority list used in Preservation, Prioritization and Collection.

M.P.4 Phase 4- Examination: Forensic examination of collected evidence by specialist trained forensic examiners is an important part of the process with the goal to provide answers to questions raised before the investigation

M.P.5 Phase 5- Analysis: Finding relationships between the recovered forensic artefacts and piecing the evidential data together to develop a timeline of the incident and its impact on the control environments.

M.P.6 Phase 6- Reporting and Presentation: Compilation of findings and analysis into a report(s) for management. This should include recommendations for engineers and consider carefully the requirements and operation of a SCADA environment

M.P.7 Phase 7 Reviewing results: For clarity the results and findings should be reviewed to ensure validation and that all forensic ‘chain of custody’ for information has been met.

**3.2 Mandatory Non-Performance Requirements**

The Network shall:

M.N.1 Operate using hardware that meets the specification of overall system.

M.N.2Operate within a windows operating system environment

M.N.3 be compatible with different type of intrusions possible to your SCADA systems.

M.N.4 Operate normally when any major risk is experienced by the system.

In addition to the mandatory performance and non-functional requirements, we have also identified certain desirable requirements. These additions are nice to have and extend the project scope in exchange for having a more robust, reliable and valuable system. The desirable requirements are formulated to extract the greatest amount of information even when operating under different conditions.

**3.3 Desirable Performance requirements**

D.P.1 Developing a forensic Toolkit:

1) Imaging/Acquisition of data

2) Analysis of acquired data

3) Forensic Reporting of findings

D.P.2 Preservation, prioritizing and collection

1) As described by Wu et al [20] “The procedure for collecting from data sources on the SCADA system depends on the volatility of data”. This is a key area for investigation as the data sources will provide a mixture of live and static data vital to the artefact discovery of the investigation

**3.4 Desirable Non-Performance requirements**

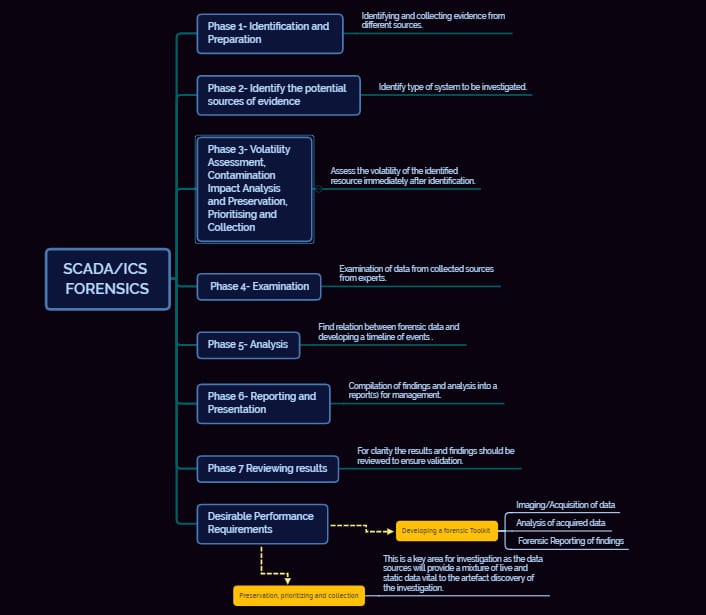
D.N.1 Checking over the parameters of the SCADA systems during an intrusion to the system.

D.N.2 Operate over various types of intrusions.

**04) Functional Architecture**

The architecture outlined below shows the functions that the system must execute to fulfill the previously mentioned requirements. The functions are derived assuming we have already have the image we want to complete or implanting.

The main functions of Cyber Forensics on SCADA/ICS Systems:

****

1. **Identification and Preparation**:

Identify the potential sources of evidence, including the systems, the network and connected devices.

For example in an oil/gas plant the PLC connected to a pressure sensor or in an electric substation Plc connected to a relay might start disfunctioning or the network systems such as IP/ARP or MAC address might have been affected.

1. **Identifying data sources:**

Identify the type of systems to be investigated including; operating system, manufacturer, serial numbers and model of PLCs, and network design and implementation.

A factory can be divided into two parts control center and field sites.

Field sites contain PLCs connected to various sensor which send data via WAN to control center. The control center consists of various network components as well as servers, Main Terminal Unit and historian (a database).

1. **Volatility Assessment, Contamination Impact Analysis and Preservation, Prioritizing and Collection:**

Assess the volatility of the identified resource immediately after identification in order to drive the priority list used in Preservation, Prioritization and Collection.

1. **Examination**:

Forensic examination of collected evidence by specialist trained forensic examiners is an important part of the process with the goal to provide answers to questions raised before the investigation.

Forensic examination usually involves which includes whether the attack was at network level or at device level. Network level includes MITM and Reconnaissance whereas at device level the attack happens on firmware resulting in denial of service.

1. **Analysis**:

Finding relationships between the recovered forensic artefacts and piecing the evidential data together to develop a timeline of the incident and its impact on the control environments.

This involves steps such as logging the PLC by tracking the values at specific memory addresses and noting down the time.

1. **Reporting and Presentation**:

Compilation of findings and analysis into a report(s) for management. This should include recommendations for engineers and consider carefully the requirements and operation of a SCADA environment.

The details of attack, type and damage is analyzed by digital forensic experts in industrial control systems and SCADA.

1. **Reviewing results**:

For clarity the results and findings should be reviewed to ensure validation and that all forensic ‘chain of custody’ for information has been met.

The end report is prepared and accordingly future course of action is decided.

**8) Developing a forensic Toolkit:**

1) Imaging/Acquisition of data: An UI can be built which will be configured with all types of components such as PLC, networking components and workstations.

Data can be acquired by data logging methods.

2) Analysis of acquired data: Analysis can be performed to detect the type of attack.

3) Forensic Reporting of findings: By comparing values at different timestamps the report will be prepared by forensic investigators.

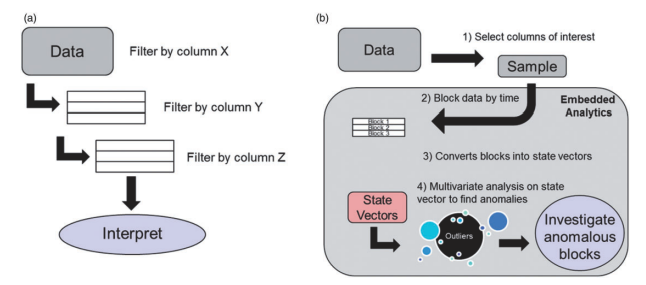
**9) Preservation, prioritizing and collection:**

The procedure for collecting from data sources on the SCADA system depends on the volatility of data. This is a key area for investigation as the data sources will provide a mixture of live and static data vital to the artefact discovery of the investigation

**05) SUBSYSTEMS DESCRIPTIONS:**

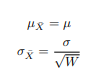
**5.1) Background for Anomaly Detection in Cyber Forensics**

This section provides an overview of SCADA protocols and their communication modes. It also presents a brief introduction of sampling distribution of sample mean and sample range. 2.1 SCADA Protocols The communication between the master device and field devices relies on SCADA specific protocols built upon different communication technologies like serial communication and TCP/IP. In this work, we analyze three different protocols: Modbus, Siemens S7 and IEC 60870-5-104. These protocols are widely used in SCADA systems but allow different communication modes. – Modbus: There are several Modbus protocols: Modbus RTU and Modbus ASCII are used in serial communication, often RS232. Modbus TCP is used for TCP communication. In this paper, we use a dataset based on Modbus RTU for testing and refer to it as simply Modbus. The Modbus protocol uses a synchronous request-response communication mode. The SCADA master initiates requests/commands stating the request type and starting address. The field device then responds by sending the requested data. – Siemens S7: We use the S7-0x32 proprietary protocol on top of TCP/IP stack and refer to it as S7 in the rest of the paper. In addition to the synchronous communications, S7-0x32 PLCs may asynchronously send messages from predefined memory areas, called Parameter Items, under certain conditions. Moreover, S7 protocol allows requesting multiple Parameter Items in one message. – IEC 60870-5-104: This is a standardized application layer protocol built upon TCP/IP stack. The protocol allows balanced/unbalanced communications. In the unbalanced mode, only the master can initiate communications to field devices. On the contrary, both the master and field devices can initiate communications in the balanced mode. This protocol allows both synchronous and asynchronous messages. The field devices can send Spontaneous and Periodic messages from predefined addresses, called Information Object Address (IOA). We will refer to the IEC 60870-5-104 protocol as IEC104 in this paper.



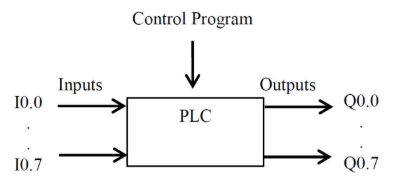
**5.2) Sampling Distribution of Sample Mean and Sample Range**

Sampling distribution of sample mean and sample range are two metrics widely used in statistical process control to monitor the stability of a production process. In these cases, quality assurance staff take a few sample sets of a certain attribute from produced products (e.g., weight of bottles) and calculate the sample mean and sample range for each sample set X = {x1, . . . , xW }. The sample mean is defined as X¯ = P(xi/W). It can provide a measure of central tendency. The distribution of X¯ is called sampling distribution of the sample mean. For a finite number of sample means X¯ i, j = {1, . . . , k,} one can compute their center of distribution as X¯¯ = PX¯ j/k and standard deviation σ and X¯ by the Central Limit Theorem (CLT). Based on CLT, for any population with mean µ and standard deviation σ, X¯ tends toward being normally distributed with



when the sample size increases. The sample range Rj = max(Xj ) − min(Xj ) can state the natural variation in a process. The distribution of Rj for finite sets of Xj is called sampling distribution of the sample range. The center of this distribution is R¯ = PRj/k. People in this area usually assume the population they take samples from follows a normal distribution. Under this assumption, one can estimate the σR with R¯ and sample size. However, we do not make any specific assumption on the distribution of the population. Instead we use quation (4) in section 5.2 to estimate it. Sample mean and sample range display variations from their historical distribution when the process is stable. If the variations exceed predefined thresholds, Upper Limitation (UL) and Lower Limitation (LL), it means the system conditions changed. In this paper, we use mean and range to model the message inter-arrival times. For the sake of simplicity, we refer to the sampling distribution of the sample mean and the sample range as the mean model and the range model in the rest of the paper.

**5.3 PROGRAMMABLE LOGIC CONTROLLER**



Programmable Logic Controller (PLC) is a special form of microprocessor-based controller that uses a programmable memory to store instructions and to implement functions such as logic, sequencing, timing, counting and arithmetic in order to control machines and processes (Fig.1) [4]. When designing and implementing control applications, PLC programming is an important task. All PLCs have to be loaded with user program to control the status of outputs according to status of inputs. PLC can identify each input and output by address. For Siemens PLC, the inputs and outputs have their addresses in terms of the byte and bit numbers.

For example, I0.7 is an input at bit 7 in byte 0 and Q0.7 is an output at bit 7 in byte 0. A PLC generates anomalous operations in the following situations [15]:

(i) hardware failure

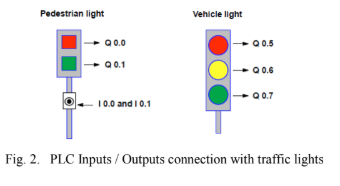
(ii) incompatible firmware version

(iii) control program bugs created by an authorized programmer or attacker;

(iv) stop and start attacks

(v) memory read and write attacks.

In order to detect these kinds of anomalous operations, we do the followings. We first capture relevant values of memory addresses used by PLC control program in normal situation. The captured values are used to train a model for the normal behavior of PLC using the semi-supervised machine learning. The trained model can be used to classify whether the PLC events are in normal operation or not. To demonstrate our proposed methodology, we developed a control program by STEP 7 (Siemens programming software for S7 PLC programming, communication and configuration) for controlling traffic light control system (Fig. 2).



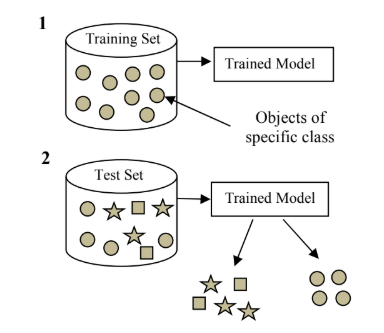
**5.4 MACHINE LEARNING**

Machine learning is a method of data analysis. It builds an automated analytical model by using algorithms to learn from data iteratively. Based on the model, machine learning allows computers to find hidden insights without being explicitly programmed [10]. Supervised learning trains a model on known input and output data so that it can predict future outputs. Unsupervised learning finds hidden patterns or intrinsic structures in input data without knowing the corresponding labels of each input [11]. Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data) [7]. One-class Support Vector Machine(OCSVM) is a semi-unsupervised algorithm.

A. One-class Support Vector Machine (OCSVM)

In machine learning, OCSVM is an One-class classification, also known as unary classification, tries to identify objects of a specific class amongst all objects, by learning from a training set containing only the objects of that class [13] (Fig. 3).

This paper utilizes OCSVM to train a model using data of normal situations (Training set), and classify PLC anomalous behavior that deviates from the trained model. This approach is suitable to deal with PLC anomalous behavior detection because OCSVM is suitable to deal with large amount of training data, since class labelling is not necessary. Also, it is relatively easy to gather training data of normal situations. On the other side, it is relatively difficult or impossible to collect data with a faulty system state. Even a faulty system state could be simulated, there is unlikely to guarantee that all the faulty state are simulated [12].



**5.5) Vulnerability and Typical Attacks of SCADA/ICS Systems**

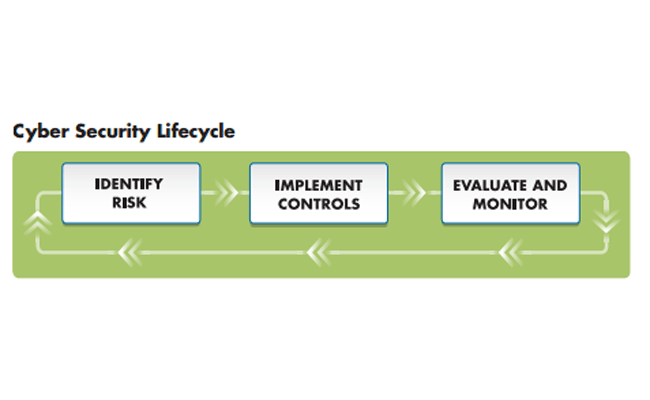
When SCADA systems were originally designed they were isolated from the network and engineers focused on providing availability of data and operations rather than confidentiality and integrity. This isolation is commonly referred to as an “air-gap”, and while originally designed as a complete physical separation, this increasingly has become to mean technological separation by the means of configurable firewalls or similar mechanism.

Originally these systems often used bespoke and manufacturer independent protocols and architectures and were therefore very difficult to understand and affect without physical access. More recent SCADA systems however, have moved to more interoperability and open standards for cost efficiency and integration into management IT systems. For example, communication is now common over Ethernet TCP-IP including more standardized control protocols and applications. Thus, SCADA systems are now susceptible to external attacks and IT based vulnerabilities.

Many SCADA systems are safety critical and must be operational for a large proportion of time, as they provide services that are vital to the economy and well-being of citizens. Downtime is managed carefully and scheduled maintenance periods are often irregular and infrequent. Therefore, many critical infrastructures are still running legacy components and systems including amongst others; Windows 95, XP, and 2000. Access to these systems for patching is a problem and therefore many IT vulnerabilities still remain that are considered resolved in the more mainstream Business IT environments.

SCADA components such as PLCs and RTUs are designed purely for functionally and are limited by their processing capability and therefore do not contain many of the authentication and access control specifications that are common in corporate IT infrastructures. Specific vulnerabilities of control devices is beyond the scope of this paper but are well documented.

As SCADA control systems become increasingly complex and distributed, the number of potential attack vectors also increases including via; the internet, enterprise network, and direct connections to the control networks and field devices. Some of the most common types of attack vectors against SCADA are: Backdoors and holes in the network perimeter. Especially in the configuration of “Air Gaps” or links to corporate enterprise IT infrastructure Vulnerabilities in common control system protocols Attacks on field devices Database attacks Communications hijacking and man-in the middle attacks Cinderella attack on time provision and synchronization.



**Typical Attacks against SCADA Systems:**

In order to undertake any forensic investigation we must first understand the types of attacks that are facing the systems and environments so as to inform the forensic process. To guide the development of a forensics framework we classify attacks against SCADA systems into 3 categories; the communication stack, hardware and software:

**Communication stack**:

Attacks can occur on the network layer for example through a diagnostic server on the UDP port. Attacks can occur on the transport layer such as a SYN flood attack saturating resources by sending TCP connection requests faster than the machine can process them.

At application layer many of the protocols used on a SCADA system have little security considerations. For example DNS forgery and packet replay are common.

**Hardware:**

Attackers gain unauthenticated remote access to devices and change data set points that may cause the devices to fail at low threshold or an alarm not to go off.

Lack of authentication for administrative tasks on the hardware mean an attacker can reprogram the logic or values and affect the functional behavior of the device.

**Software:**

SCADA systems use a variety of software to provide functionality from traditional IT applications to bespoke embedded device applications and more custom HMI or Historian control applications.

There is no privilege separation in embedded OS for example VxWorks embedded OS used in field devices provides minimum memory protection.

1) Buffer overflow attacks are possible in bespoke applications mainly through workstations similar to standard IT systems or in industrial control automation software such as historian servers. In addition, field devices themselves that rely on real time operating systems (RTOS), are more susceptible to memory challenges by exploiting the fixed memory allocation time requirement in RTOS system.

2) SCADA components especially in legacy networks are subject to accumulated memory fragmentation which can lead to programs stalling.

3) Structured Query Language (SQL) is widely used to store sensor information in historians and other databases thus, if not designed properly at application level the systems are susceptible to SQL injection attacks [12].

Whilst these types of attacks are also prevalent in enterprise IT systems, and indeed some of the SCADA environments are inheriting the vulnerabilities from enterprise applications it is worth reiterating that the implementation in these environments in very different. Thus, a forensic framework for SCADA must consider the requirements of this operating environment carefully. We establish some of these particular requirements in the following section.

|  |  |
| --- | --- |
|  |  |
| Vulnerability of Scada/ICS systems | Attacks Comparison on Scada Systems |

**5.6)** **ICS Forensics**

**5.6.1 ICS Forensic Challenges**

Forensic acquisition tools are widely available for conventional ICT systems like hard disks, volatile memory (RAM) and common consumer electronics like mobile phones and navigations systems. Similar tools do not exist for most ICS devices. Besides, in ICS systems, safety is the main goal rather than Security. If ICT people talk about Security and Safety in ICT systems they do mean:

• Firewalls to prevent hackers from entering the system since confidential information must be protected.

• Antivirus and Antimalware for protecting the users and the systems against viruses.

• Anti-spam to protect the users against spam in their mail. However, If ICS people talk about Security and Safety in ICS systems they do mean:

• Protect the system against dangerous issues like wrong values in PLC’s.

• Flow control and temperature sensors in the chemical plant.

• Voltage and current control in electrical grid installations.

Not only the other interpretation or different jargon can be an issue, also the difference between ICT people who are working mainly in the office or data centers and the ICS people working on the field inside the plant or control room. There is a big gap between the two different departments; other goals and other problems are creating completely different priorities on a daily base.

**5.6.2 ICS Forensic Process**

The purpose of our approach is to safeguard the important information from the ICS system. Depending if we talk about a running system which is still intact and connected to other devices like a distributed control system, or if we talk about a standalone control system like a single PLC or a post mortem investigation after a big incident like a fire or explosion in a chemical plant, several information sources are available to ac- quire important digital evidence for digital forensic investigation purposes. For this reason we have to setup an ICS Forensic process. Inside this process, we split up the information from two different sources:

• Network data

• Device data

**Network data acquisition**: For network data acquisition network investigation (depending on our investigation) we have to decide on what level (or levels) we need to analyze the network traffic. Network Levels: A typical distributed ICS system has at least three different levels of network types:

• Device level such as sensor, programmable logic controller (PLC), actuator.

• Cell Level that irresponsible to control the device controllers.

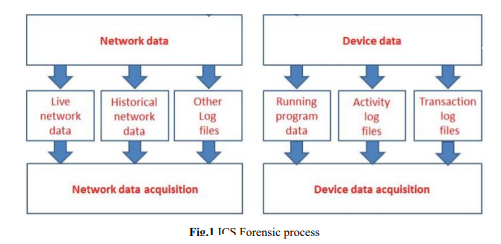
• Plant Level that is responsible to control the cell controllers.

Beside, network data can also be historical information like backup files, logging databases etc. Sources of network data can be listed as:

• Live Network Data (raw network data, Arp tables, flow records, etc.)

• Historical Network Data (host based logs, database queries, firewall-logs, etc.)

• Other Log Files (backup archives, access point logs, historians, etc.)



Not all tools or methods are safe to use in ICS environments. ICS systems often monitor or control processes in which a failure may have disastrous consequences (or may be otherwise very undesirable). For this reason active probing (like scanning for open ports and then opening arbitrary TCP connections) should generally be avoided.

**5.6.3) Device data acquisition:**

Device data acquisition forensic tools do not exist for most ICS devices. Product specific service tools for programming a PLC, saving the program and servicing log files from a PLC to a service computer do exist. The question is can we use those service tools in a forensic matter to save important data from the PLC for later analyses? The sources of device data can be listed as:

• Running Program Data such as RAM dump, CHIP images, Memory cards…

• Activity Log Files such as RAM dump, active processes, control room logs, etc.

• Transaction Log Files such as Serial communication logs, Error logs, Event logs, etc.)

**5.7)** **SCADA DIGITAL FORENSIC PROCESS**

Digital forensics is an important part of an incident response strategy in an IT forensic investigation following an incident and will provide an effective response in a forensic manner Imtiaz (2006).There are several steps to conduct a digital forensic investigation with basic steps being; preservation, identification, extraction and documentation of digital evidence. The purpose of the digital forensic process model is to demonstrate in the court of law that the evidence has been collected in the correct manner and following legal procedures with scientific backing.

**Currently a SCADA forensics model identified by Radvanovsky and Brodsky (2013) has the following investigative steps:**

**Step 1 Examination**

Identify the potential sources of evidence, including the systems, the network and connected devices. In addition to these sources an investigation should examine other systems that have a relationship to the SCADA system such as access terminals, servers and routers.

**Step 2 Identification**

Identify the type of systems to be investigated, this includes operating system, the manufacturer including the serial numbers and model types of PLCs, the network design and implementation. Once the operating system has been identified it is important to note a system could be running more than one operating system such as a Linux variant. Many SCADA systems run a child system over a base OS. During the identification process several areas can assist, including manufacturer’s documentation, design specifications, network diagrams, and the HMI (Human Machine Interface).

**Step 3- Collection**

Collect potential evidence from the memory systems that are suspected to be part of the SCADA system being investigated. It is critical that volatile and dynamic information across various network cards and controller units be collected first to prevent any loss of data from network connections. Network traffic is also captured to discover anomalous traffic.

**Step 4- Documentation**

In this step it is critical to keep accurate documentation of the investigation to ensure chain of custody. Records need to be kept of potential evidence as well as case numbers and the time when the evidence was collected. Many investigators will photograph the entire investigation process including the systems that could be connected to the SCADA system or that are presently connected, to ensure that the examiner will be able reconnect them if needed later. A detailed report would need to be produced of the whole digital forensic process to include the captured system throughout the collection process. The last stage is to gather all the information together and store in a secure and safe location.

|  |
| --- |
|  |
| SCADA FORENSIC SYSTEM |

**Part 05: Importance**

The importance of SCADA systems is automation. It allows an organization to carefully study and anticipate the optimal response to measured conditions and execute those responses automatically every time. Relying on precise machine control for monitoring equipment and processes virtually eliminates human error. More importantly, it automates common, tedious, routine tasks once performed by a human, which further increases productivity, improves management of critical machine failure in real-time, and minimizes the possibility of controllable environmental disasters.

In addition, SCADA systems are needed to monitor and control a large geographical displacement where an organization may not have enough manpower to cover. Thus, reliable communication and operability of these areas or sites is critical to profitability.

SCADA monitors and controls entire sites and systems spread out over large geographic areas. You know how your sites and systems are performing remotely, 24/7.

• LESS TIME SPENT BEHIND THE WINDSHIELD TRAVELING FOR SYSTEMATIC SITE VISITS.

• LESS FUEL

• LESS DEPRECIATION

• MORE TIME TO BE PRODUCTIVE!

SCADA systems can alert you of changes in your system, tell-tale signs that something is about to fail in your system.

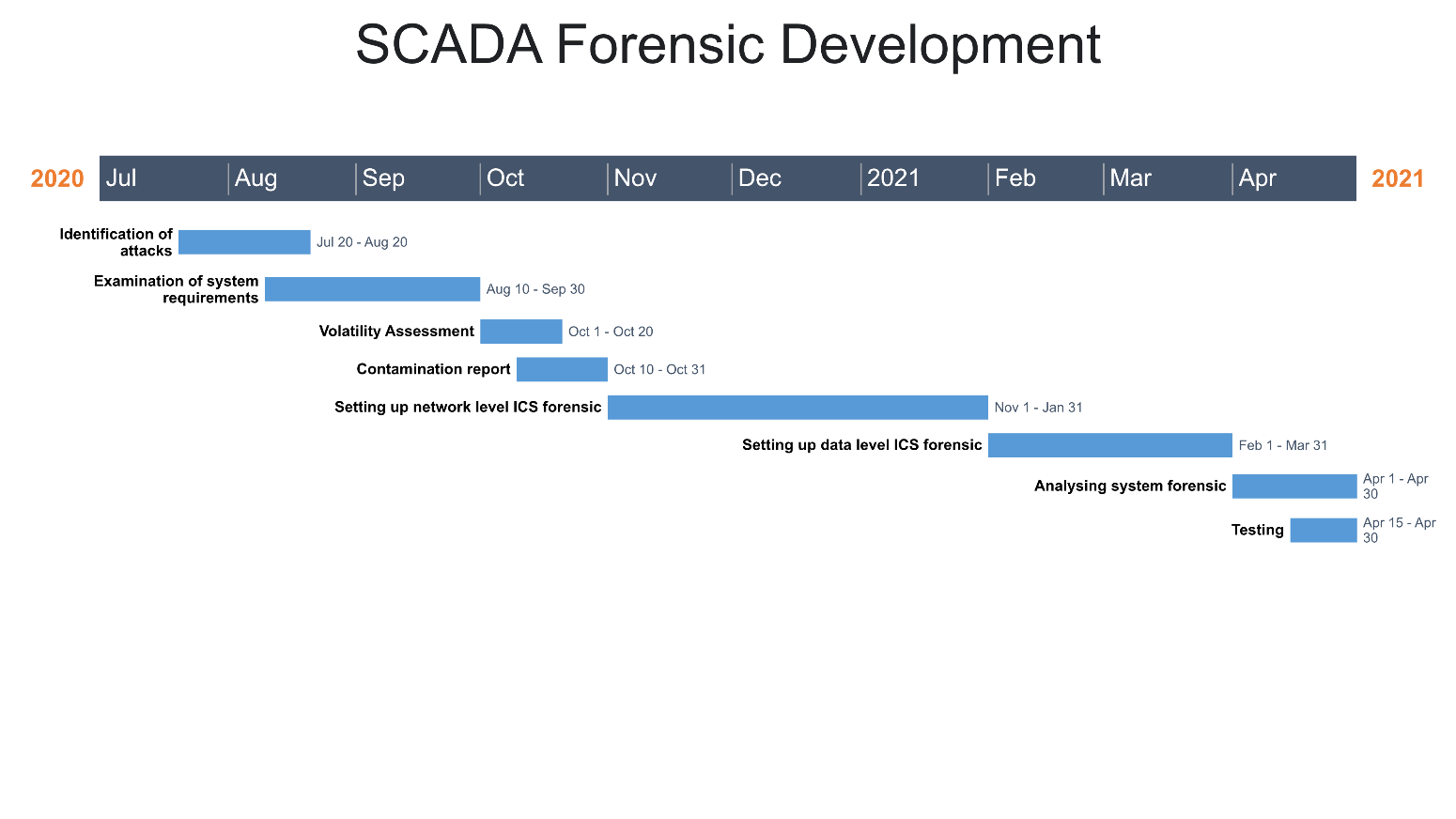
# **06) Project Management**

## Work Plan and Tasks:

The majority of the work required for this project is software based. Our approach will be to focus on identifying all possible threats by typical attacks on SCADA system, examining other systems that have a relationship to the SCADA system such as access terminals servers and routers, identify the operating system, tackling these attacks independently in the odd semester, and then integrating a defence mechanism to tackle all 3- Hardware, software and communication stack attacks together in even semester. Our work plan reflects project management throughout the year.

|  |  |  |
| --- | --- | --- |
|  | WORK PLAN |  |
| Identification and Examination | 1.1 | Identify the type of attacks possible on SCADA system and the layer it affects. |
| 1.2 | Identify the potential sources of evidence, including the systems, the network and connected devices. |
| 1.3 | Examine other systems that have a relationship to  the SCADA system such as access terminals,  Servers and routers. |
| 1.4 | Examining the type of systems to be investigated, this includes operating system, etc. |
| Volatility Assessment, Contamination Impact Analysis and Preservation,  Prioritising and Collection | 2.1 | Assess the volatility of the identified resource |
| 2.2 | Building and imaging the contamination impact report |
| 2.3 | Integrating a collection of hardware, software, communication stack based attacks |
| 2.4 | Setting up priority mechanism |
| Setting Network level ICS Forensic process | 3.1 | Step-by-step procedure for forensic analysis including creating environment scanning for anomaly detection |
| 3.2 | Production and replay for each vulnerability is performed |
| 3.3 | Each vulnerability is analysed, resulting in the creation of an IDS rule set. |
| 3.4 | Testing of each rule set is performed. |
| Setting Device level ICS Forensic process | 4.1 | Acquisition of  data from suspicious PLCs |
| 4.2 | Analysis of the data acquired |
| Analysis | 5.1 | Finding relationships between the recovered forensic artefacts |
|  | 5.2 | Create timeline of the incidents and its impact on control environment |
| Testing and Reviewing results | 6.1 | Testing the system in its integrated form |
|  | 6.2 | Reviewing each mechanism |

# 02) Work Schedule GNATT CHART:



**07) References:**

1. E. Byres. (2012) Securing SCADA systems from APTs like Flame and Stuxnet – Part 1. Tofino Security. Weblog.
2. Detecting Anomalous Behavior of PLC using Semi-supervised Machine Learning by Ken Yau, KP Chow, SM Yiu, CF Chan Department of Computer Science The University of Hong Kong Hong Kong, China
3. Digital Forensic Analysis of Industrial ControlSystems Using Sandboxing: A Case of WAMPAC Applications in the Power Systems Asif Iqbal \*, Farhan Mahmood and Mathias Ekstedt
4. Programmable Logic Controller Forensics by Irfan Ahmed | University of New Orleans, Sebastian Obermeier | ABB, Sneha Sudhakaran and Vassil Roussev | University of New Orleans
5. Towards a SCADA Forensics Architecture by Tina Wu, Jules Ferdinand Pagna Disso, Kevin Jones and Adrian Campos EADS Innovation Works Quadrant House Celtic Springs ,Coedkernew, Newport NP10 8FZ UK
6. E. Casey (2011) Digital Evidence and Computer Crime - Forensic Science, Computers and the Internet, 3rd Edition. Academic Press 2011, ISBN 978-0-12- 374268-1, pp. I-XXVII, 1-807
7. A. Nicholson, S. Webber, S. Dyer, S. Patel, and H. Janicke (2012). {SCADA} Security in the light of cyberwarfare. Computers & Security 31(4), 418 – 436.
8. Forensic Analysis of SCADA/ICS System with Security and Vulnerability Assessment by Umit Karabiyik, Naciye Celebi, Faruk Yildiz, James Holekamp. Khaled Rabieh at Department of Computer Science Sam Houston State University
9. Timing-based Anomaly Detection in SCADA Networks by Chih-Yuan Lin, Simin Nadjm-Tehrani, and Mikael Asplund, Department of Computer and Information Science, Link¨oping University, Sweden
10. ACE – An Anomaly Contribution Explainer for Cyber-Security Applications by Xiao Zhang, Manish Marwah, I-ta Lee, Martin Arlitt and Dan Goldwasser
11. Detectng Anomalies in Programmable Logic Controllers Using Unsupervised Machine Learning by Chun-Fai Chan and Kam-Pui Chow
12. Detecting Anomalous Behavior of PLC using Semi-supervised Machine Learning by Ken Yau, KP Chow, SM Yiu, CF Chan Department of Computer Science,The University of Hong Kong
13. Cyber anomaly detection: Using tabulated vectors and embedded analytics for efficient data mining by Robert J Gutierrez, Kenneth W Bauer, Bradley C Boehmke, Cade M Saie and Trevor J Bihl.
14. Unsupervised Anomaly-based Malware Detection using Hardware Features by Adrian Tang, Simha Sethumadhavan, and Salvatore Stolfo