Tools, Techniques, and Methodologies: A Survey of Digital Forensics for SCADA Systems

Rima Asmar Awad
Oak Ridge National Lab
Tennessee Tech University
awadrl@ornl.gov

Saeed Beztchi
University of Tennessee, Knoxville
Oak Ridge National Lab
beztchisa@ornl.gov

Jared M. Smith
Oak Ridge National Lab
University of Tennessee, Knoxville
smithjm@ornl.gov

Bryan Lyles
Oak Ridge National Lab
lylesjb@ornl.gov

Stacy Prowell
Oak Ridge National Lab
prowellsj@ornl.gov

Abstract

Security aspects of SCADA environments and the systems within are increasingly a center of interest to researchers and security professionals. As the rise of sophisticated and nation-state malware targeting such systems flourishes, traditional digital forensics tools struggle to transfer the same capabilities to systems lacking typical volatile memory primitives, monitoring software, and the compatible operating-system primitives necessary for conducting forensic investigations. Even worse, SCADA systems are typically not designed and implemented with security in mind, nor were they purpose-built to monitor and record system data at the granularity associated with traditional IT systems. Rather, these systems are often built to control field devices and drive industrial processes. More succinctly, SCADA systems were not designed with a primary goal of interacting with the digital world. Consequently, forensics investigators well-versed in the world of digital forensics and incident response face an array of challenges that prevent them from conducting effective forensic investigation in environments with vast amounts of critical infrastructure. In order to bring SCADA systems within the reach of the armies of digital forensics professionals and tooling already available, both researchers and practitioners need a guide to the current state-of-the-art techniques, a road-map to the challenges lying on the path forward, and insight into the future directions R&D must move towards. To that end, this paper presents a survey into the literature on digital forensics applied to SCADA systems. We cover not only the challenges to applying digital forensics to SCADA like most other reviews, but also the range of proposed frameworks, methodologies, and actual implementations in literature.

CCS Concepts

• **Security and privacy → Embedded systems security**; • **Applied computing → Evidence collection, storage and analysis**; **Network forensics**; **System forensics**; **Data recovery**; • **Computer systems organization → Embedded and cyber-physical systems**;

Keywords

SCADA, ICS, Digital Forensics, Survey

ACM Reference Format:
Rima Asmar Awad, Saeed Beztchi, Jared M. Smith, Bryan Lyles, and Stacy Prowell. 2018. Tools, Techniques, and Methodologies: A Survey of Digital Forensics for SCADA Systems. In *Proceedings of ICSS 2018 at ACSAC 2018 (ICSS '18).* ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.475/123_4

ICSS '18, December 2018, Puerto Rico, USA

https://doi.org/10.475/123_4

1 Introduction

Industrial Control Systems (ICS) and supervisory control and data acquisition (SCADA) systems are the underpinning technologies that ensure the proper operation and functionality of critical national infrastructures. Early SCADA systems were intended to run in isolation, completely unconnected from the Internet. Thus, modern security threats over the Internet were not within the threat model. However, in recent years, SCADA systems have evolved to communicate over varying types of networks and on vastly different scales. Consequently, these systems are increasingly exposed to the class of threats which target traditional digital infrastructure.

In recent years, the number of targeted and sophisticated attacks against SCADA environments has dramatically increased. Stuxnet, Duqu, and Wiper are prominent examples of sophisticated attacks that were purpose-built to sabotage the operation of targeted SCADA systems [10, 39]. When these systems are compromised, sabotaged, or attacked, security analysts and operations personnel need to get to the root cause of the attack as quickly as possible. Along the way, collecting as much data as possible to turn over to law enforcement and other national security sources is an extremely critical function of the analyst's investigation. Yet this need lies directly in the way of arguably the more important concern: turning the lights, water, or other critical infrastructure back on, and making sure it *stays on*.

In order to balance the needs of data collection with rapid incident response and recovery, the techniques, and knowledge from the field of digital forensics can be an extremely powerful tool. Despite this need, most data collection and forensic analysis systems focus on the traditional IT infrastructure and critical infrastructure networks, leaving analysts in the dark on versatile tooling to analyze and remediate the SCADA devices themselves. Fortunately for

operators in the field, the proliferation of threats against critical infrastructure has driven a combined academic-industry response to enhance the security of the SCADA environment, including developing novel approaches to conducting forensics on these systems. Recent work has largely focused on adapting tooling and methodology used for traditional forensics to the array of challenges when dealing with forensics of typically proprietary SCADA systems.

To that end, this paper presents the first survey of research and technical efforts from literature to develop forensics solutions and tools tailored to SCADA systems. In our extensive literature review, we identify the current state of the art in existing research and shed light on research gaps. Our work provides insight into the future directions for efforts to raise SCADA system forensics to the level of sophistication and versatility of traditional digital forensics. Unlike other reviews to-date, our paper attempts to broadly cover the frameworks, methodologies, and actual implementations in this space. In particular, we aim to focus on *forensics*-specific literature. Much work exists in the domain of anomaly and intrusion detection, malware analysis, and intrusion detection; however, we instead extensively cover digital forensics and incident response for SCADA environments, specifically the process around forensic data capture and system remediation. Though we do not claim this survey is exhaustive, we believe have enumerated the vast majority of state-of-the-art and seminal work in the emerging area of digital forensics for SCADA systems. Notably, **Table 1** enumerates all the cited works, their category, and the domains, devices, and protocols involved in each.

The rest of this paper now proceeds as follows: Section 2 covers the necessary background for understanding the domain of SCADA systems and the ICS environment. Section 3 enumerates many of the key challenges to truly realizing robust and comprehensive forensics systems for SCADA. Section 4 covers the literature on frameworks and methodologies proposed for integrating digital forensics into the SCADA environment. Sections 5 and 6 present the state-of-the-art tools and techniques along with their evaluations on digital forensics for SCADA networks and end-point devices. Finally, Section 7 concludes the paper with suggested steps forward and topics ideal for stimulating discussion and action from the insights found in our review.

2 Background

Before we dive into review of the past and current efforts in digital forensics for SCADA, we present a high-level overview of the architecture and terminology surrounding SCADA systems. Shown in Figure 1, a SCADA environment often starts with the control center. From the control center content, communication, and commands travel across either internal or external networks towards the field devices running a mix of proprietary or open-source embedded operating systems. These devices can be sensors, actuators, computing modules, physical infrastructure, and much more.

At the control center, the **Historian** captures, logs, and enforces policy-based storage of data from the downstream field devices. Driving the historian and the downstream devices is the **Master Terminal Unit** (MTU), which could be one or more devices in an actual deployment. This represents the central "logic" driving the deployed SCADA devices. Finally, a number of **Human-Machine Interfaces** (HMIs) to the control systems allow operators, managers, and engineers to monitor and steer the operations of the deployed devices.

All data and communication exits and enters the control center via traditional communication methods, often Ethernet, through either internal or external networks. One of the key challenges of protecting critical infrastructure is the varied deployment in practice; accordingly, the communication infrastructure for SCADA systems in practice could rely on any number of combinations of wired, wireless, internal, external, secured, and unsecured network devices and deployments.

Finally, at the level of SCADA system deployment in their final operating system environment, a number of controller devices exist, which take input from the actuators, sensors, and devices in the field and communicate it to the control center. These end-point devices include, but are not limited to, **Programmable Logic Controllers** (PLCs), **Real-Time Automation Controllers** (RTACs), and **Remote Terminal Units** (RTUs). These systems are not mutually-exclusive, yet they each can combine to compose most in-use SCADA deployments. PLCs often process ladder logic, and RTUs usually coordinate data and output from downstream devices such as smart meters back to the control center via the communication network. Nonetheless, each of these devices presents yet another vector of attack, each made up of hundreds of potential OSs and versions in practice.

For a deeper look into the architecture behind SCADA systems and the challenges associated with them, Eden *et al.* [14] provides broad coverage. Specifically, the authors highlight the forensic challenges in industrial control systems and their development over the past few decades. A breakdown of the SCADA system architecture is also presented in the paper, along with the most recent tools and methodologies currently used to conduct forensics investigation on ICS systems and respond to incidents in a timely manner.

2.1 Control Center Forensics

As referenced by Figure 1, the control center contains the master nodes responsible for coordinating all downstream network communications and field devices. Fortunately for operators, these systems are often composed of more traditional IT-based OSes, such as Windows or Linux. If these devices have the appropriate human-machine interface, API, network, or serial ports, interacting with them, retrieving data in a live manner, and remediating them becomes a problem that has been well-addressed by many existing forensics tools. These tools include Volatility, Rekall, Encase, Redline, and other automated analysis systems [17, 19, 34, 40].

Furthermore, significant literature exists that surveys the challenges associated with such systems, including work by Soltani *et al.* [35], Maras *et al.* [27], and Nelson *et al.* [30]. Within the control center, traditional networking tools apply as well, and are covered in depth from a forensics perspective by Khan *et al.* [23]. From a host or end-point perspective, Ligh *et al.* [26]'s seminal work on Memory Forensics defines the practical tools and techniques in the space.

3 Challenges to SCADA System Forensics

Digital forensics, a branch of forensics science that encompasses the recovery and investigation of material found in in digital devices,

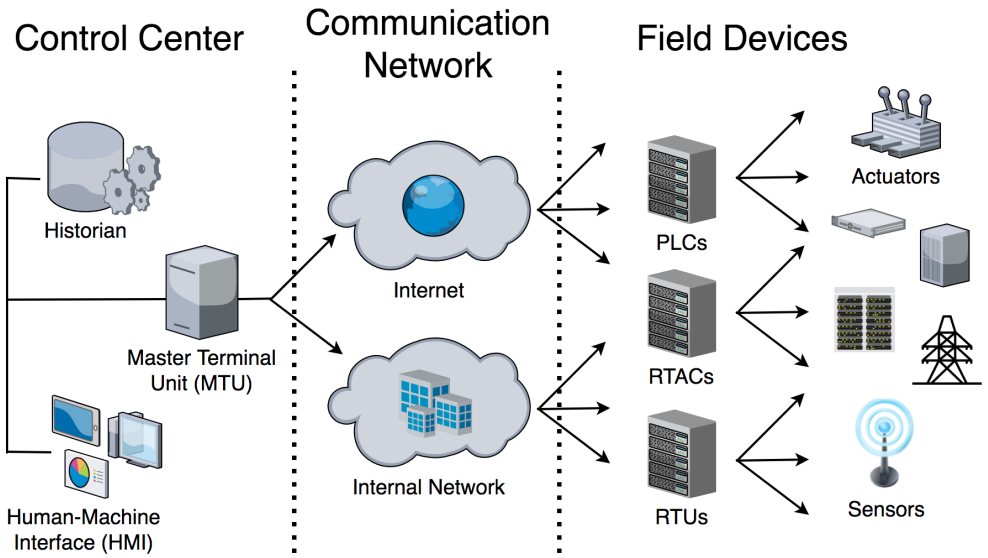

Figure 1: Overview of the Architecture of a Typical SCADA Environment

Category	Specific Domain, Device, or Protocol	Reference
Technical Challenges		van der Knijff [42], Stüttgen *et al.* [37], Iqbal *et al.* [21], Kilpatrick *et al.* [24]
Research Challenges		van der Knijff [42], Ahmed *et al.* [2], Slay *et al.* [33], Vaughn *et al.* [44]
Frameworks/Methodologies	Sensor Networks	Cardenas *et al.* [6]
	Events and Logging	Taveras *et al.* [38]
	Siemens S7 PLC	Yau *et al.* [47]
	Attacks against PLCs	Chan *et al.* [7]
	Incident Response	Eden *et al.* [15]
	Data Retrieval and Incident Response	Eden *et al.* [13]
	Applying Existing IT Tools to SCADA, Case Study on USB-based Attacks	Betts *et al.* [4]
	Data Retrieval	Stirland *et al.* [36]
	Live Data Acquisition	Ahmed *et al.* [3]
	Integrating with Legacy Systems, Network Protocol Analysis	Chandia *et al.* [9]
	Network Data and Device Memory Acquisition/Data Retrieval	Van Vliet *et al.* [43]
	Large-Scale Multi-Protocol and Device Testbed	Adhikari *et al.* [1]
Network Forensics	Fuzzing DNP3 and Modbus	Devarajan *et al.* [12]
	Siemens S7 PLCs	Kleinmann *et al.* [25]
	Wireless Sensor Networks, Multi-Agent Systems	Elhoseny *et al.* [16]
	DNP3, Modbus, Snort IDS	Valli *et al.* [41]
	Firewalls and Network Segmentation	Mittal *et al.* [29]
	GE-SRTP Protocol, GE Fanuc Series 90-30	Denton *et al.* [11]
Device Forensics	Data Retrieval, Differential Analysis	Gougeon*et al.* [20]
	Water Treatment Testbeds, Sensor State	Junejo *et al.* [22]
	PLCs, Memory Analysis (addresses)	Yau *et al.* [48]
	Siemens TIA Portal, Logging and Event Collection	Chan *et al.* [8]
	Siemens S7-1200 PLC, Ladder Logic, Data Retrieval	Yau *et al.* [46]
	File Analysis, Package and Dependency Analysis	Schlegel *et al.* [31]
	Siemens S7 PLC, Memory Analysis (addresses)	Wu *et al.* [45]
	PLC Firmware Analysis, Baseline Creation	McMinn *et al.* [28]
	Programmable Controller Communication Commands (PCCC), File Analysis	Senthivel *et al.* [32]
	JTAG data capture, Memory Analysis (raw dumps), Offline Analysis	Breeuwsma *et al.* [5]

Table 1: Summary of Reviewed Literature

is an essential element in tracking and verifying the operability and security of general and embedded computer systems. The critical nature of SCADA systems and the fact that field devices are connected to physical processes makes forensics-based techniques, including live data acquisition and analysis, a viable solution for digital investigation on SCADA systems. Additionally, these systems often need to run 24/7 to control industrial and infrastructure processes, further motivating the need for forensics oversight.

Specifically, live forensics involves acquiring volatile and non-volatile data while the system is in operation and then analyzing the device with forensics tools offline. Challenges lie at every step of the process, from handling the continuous changes of volatile

memory as well as the validity of digital evidence. Beyond general challenges, investigators conducting live forensics on SCADA systems must adapt their tools to function across a diverse set of both IT and OT infrastructure. In the following two subsections, we both summarize and highlight existing research on the challenges to applying forensics to SCADA systems.

3.1 Technical Challenges

Digital forensics investigators examining SCADA systems must understand the cause and effects of attacks against their infrastructure. However, SCADA systems present significant technical challenges to analysis in practice. Since these devices must remain online for extended periods of time, live forensics has become increasingly important in order to capture the data and analyze offline. Nonetheless, the tooling in this space is sparse, and will be discussed later in Section 6. Furthermore, SCADA devices present significant technical implementation differences from traditional IT infrastructure, and thus require different sets of forensic tooling.

To that end, van der Knijff [42] provides a "crash course" on control systems and the opportunities for improving the forensics tooling and analysis methods for such systems. Specifically, the paper provides a look into the differences between ICS and IT, as well as highlighting issues which security personnel embedded in the control system space might not yet be aware of. Furthermore, the paper provides an overview of the various protocols and device types in ICS and SCADA environments.

Stüttgen *et al.* [37] categorizes several of these methods for obtaining forensic memory images of firmware and other devices with attention to the potential for malware to interfere with the process. The authors also present a brief reference to the open-source Winpmem and Pmem programs for acquiring firmware memory, going on to describe a plugin for the Volatility program which will extract Advanced Configuration and Power Interface (ACPI) tables for certain embedded devices.

Given that the communication between SCADA devices and the control center can be tampered with, analysts often suffer from a lack of ground truth for the forensic data on a system. Worse, SCADA systems are diverse by nature, and usually have customized kernels running on their components, which complicates the process of capturing benign state of the device. Available data acquisition tools may not run on a customized kernel or proprietary OS unless compatibility has been ensured by the manufacturer. When capturing this data the lack of resources and logging capabilities on these systems can be very limiting [2].

Iqbal *et al.* [21] specifically points out that for most SCADA or industrial control systems, logging capabilities are limited. This provides difficulties for a digital forensics team needing to identify users who have compromised a system. The authors suggest that the log files of these systems must provide more effective capabilities to allow for long-term gathering of critical forensic data.

Kilpatrick *et al.* [24] emphasizes the changes to SCADA systems over recent years. With the advancement and inter-connectivity of all industrial technologies, such changes have occurred without including the security mechanisms SCADA systems need. As communication between SCADA system components becomes more standardized within IT networks, the number of vulnerabilities in a system is likely to grow.

3.2 Research Challenges

The technical challenges associated with SCADA system forensics, as well as the increased need of national governments to protect the nation's infrastructure, has led to a higher interest in research from the security community. The critical, always-online nature of SCADA systems imposes both technical and research challenges.

Ahmed *et al.* [2] argues that for research in this domain to be practical and conclusive, realistic SCADA systems are needed for research purposes. Unfortunately, building or acquiring realistic SCADA systems for research purposes can be prohibitively expensive. Although simulators and testbeds can be used to conduct research experiments, they are often error prone and cannot be used to create many complex real-world scenarios.

Slay *et al.* [33] asserts the need to understand the forensics of computing process before developing a solution to the SCADA forensics problem. The authors argue that forensic computing is still difficult to implement for industrial control systems. Currently, there is no standardized or well-documented strategy for collating data for SCADA systems to obtain evidence for criminal activities. A primary issue in forensics for SCADA systems is data retrieval from volatile memory and network devices. Moreover, legacy systems may not provide long-term logs due to limited memory architectures.

Vaughn *et al.* [44] address the problem of securing hardware by presenting the challenges and issues associated with modeling critical systems. Furthermore, the authors lay out in detail other test beds used in research and practice. This work provides significant motivation for conducting research on SCADA security, including forensics, with an evaluation-first approach.

Furthermore, research in this area often requires engaging SCADA device manufacturers, control center operators, and other stakeholders in order to provide researchers with a view into the technical problems that arise in operation. Nevertheless, the vital nature of most critical infrastructure organizations discourages industry staff from collaborating with the research community.

While restricting collaboration may result in greater confidentiality of an organization's devices and may potentially prevent information leakage, this restriction severely hinders the development of more capable digital forensics tools and techniques [2].

4 Frameworks and Methodologies

To address the challenges that arise when attempting to apply forensics techniques to SCADA environments, security researchers and practitioners present a range of frameworks and methodologies for such challenges. Though some of these frameworks are only at the network-level, others dig deeper into specific device-level methods. In general, we found that most work in the model-building and framework construction space just scratches the surface of possible approaches to SCADA forensics. To that end, this section covers the state-of-the-art research in models, frameworks, and methodologies for SCADA forensics.

Cardenas *et al.* [6] describes a taxonomy for the necessary security properties of sensor networks, the threat models associated with them, and the security design choices in this space. By focusing on the practical aspects of deploying secure sensor networks, the authors offer a guide to the challenges to building effective defenses against threats in a primarily SCADA-driven environment.

Taveras *et al.* [38] proposes a model that uses a finite state automaton as an agent that would monitor SCADA events in real-time. These events are then compared against a set of pre-defined rules to determine whether any changes have been made to the state. If changes are detected, the agent switches to forensic mode to log the information for use in a forensic investigation. However, the authors' model has not been tested, and further research would be required to assess its effectiveness on a real SCADA system.

Yau *et al.* [47] addresses the shortfall in logging capabilities in SCADA devices and proposes a logging system for Siemen's Step 7 (S7) Programmable Logic Controllers (PLC). The logging system is implemented as a transparent proxy between an Ethernet network and the PLC that forwards all the traffic except for the S7 PLC communication traffic. The proposed logging system functions by detecting and capturing connection requests and traffic on TCP port 102. Potentially useful forensic information is then extracted, translated, and written to an audit log file that can be accessed and read by a forensic investigator.

Chan *et al.* [7] proposes a novel method to enhance the security and forensics of industrial control systems by incorporating a security block in a PLC. The security block monitors the PLC and detects potentially compromising memory changes. Evaluation of the proposed method indicates that it can be used to support incident response and forensics investigation. Additionally, this method is capable of detecting attacks rarely identified by other methods, while maintaining low overhead.

Eden *et al.* [15] presents an overview of SCADA forensics process and discusses some of the existing challenges when carrying a SCADA forensics investigation. The authors propose a model for SCADA incident response and discuss ways in which the challenges can be controlled, and the process can be improved.

Eden *et al.* [13] identifies the assets of SCADA systems and provides a list of tools and methods used for data retrieval and acquisitions on such systems. This paper also discusses key stages during an incident response process and the order in which volatile data needs to be acquired to maintain data integrity and prevent losing useful data.

Betts *et al.* [4] presents a methodology for forensics and cyber incident response in the ICS environment. The authors also evaluate the applicability of current IT forensic tools and the requirements of an "ICS forensic toolbox." Finally, the authors present an experimental case study of a USB-based malware attack, a man in the middle attack, and a remote access attack.

Stirland *et al.* [36] explains how the SCADA systems consist of multiple components connected to a main network. Common attacks against SCADA systems can affect the hardware or applications that run on the system. A list of steps for digital forensics is provided, which specifies the ordering of each task to effectively acquire the data necessary.

Ahmed *et al.* [3] proposes a method to make live forensics a viable solution for SCADA systems. Live data acquisition involves acquiring both volatile data (such as the contents of physical memory) and non-volatile data (such as data stored on a hard disk). Live acquisition is different from traditional dead disk acquisition, which involves bringing the system offline before the acquisition and consequently losing all volatile data. However, despite the importance of live data acquisition it is still unclear how contemporary live data acquisition tools can be run on a SCADA system so that they minimize the risk of disruption of critical services.

Chandia *et al.* [9] addresses the need for increased security as SCADA systems become more interconnected. One proposed solution is a security suite that protects multiple levels of the network, as well as providing compatibility with legacy systems. Another approach involves the process of gathering network traffic and forensically analyzing the captured packets.

Van Vliet *et al.* [43] discusses the numerous reasons for industrial control system forensics that range beyond cybercrime. The forensics for these systems is also notable as the method of data acquisition may depend on the category of the incident. Network data acquisition can be done to access data from many levels, each providing network traffic for different controllers. Data acquisition for these systems requires tools that are yet to be developed, though data can still be obtained using log files for RAM dumps, listing processes, or event logs. The authors present a case study involving a wind turbine that caught fire to exemplify the necessity of effective forensic investigations.

Ghaleb *et al.* [18] presents a SCADA simulation environment that can be used in security analysis and digital forensics training. The environment functions independently of included devices, is easy to configure and deploy, and supports hybrid device architectures. The authors analyze the usefulness of their SCADA simulation environment for digital forensics on water distribution systems and electrical grids.

Adhikari *et al.* [1] presents a robust, multi-device, hardware and virtual testbed for studying the security properties of SCADA devices. With their automated attack system and the control infrastructure, real-world cyber attacks are capable of evaluation against both research and industry tools setup in the test bed to monitor, detect, and remediate the attack. Notably, this work integrates "industry standard hardware, software, and wide area measurement systems (WAMS)" unlike most other testbeds, which are often limited in nature to only one or few of those components.

5 SCADA Network Forensics

Beyond frameworks and methodologies, the next two sections describe the state-of-the-art approaches and systems implemented in practice for network-level and the device/endpoint-level in applying digital forensics to SCADA systems. Though several of the works presented in the prior section contained implementations or evaluations of frameworks, we consider any work in the following two sections to address a specific or smaller subset of the general SCADA environment. To begin, we start with work at the network-level, covering network communications between field devices and PLCs, RTUs, or RTACs, and network protocols such as Modbus and DNP3.

Devarajan *et al.* [12] presents the SCADA fuzzer, a tool for detecting protocol anomalies, unauthorized communication, and possible denial of service attacks in widely used SCADA protocols such as Modbus and DNP3. The tool is composed of various components including agents that monitor the SCADA network communication and log PCAP files to detect faults at runtime.

Kleinmann *et al.* [25] describes the packet parsing and protocol models needed to build an IDS for networks with Siemens S7 PLCs. The paper describes the packet formats and types used by the S7. It

also proposes a Deterministic Finite-state Automaton (DFA) model for interpreting the traffic and identifying "not-normal" sequences. The evaluation was mildly positive (authors claimed success) but still had a one percent false positive rate.

Elhoseny *et al.* [16] addresses the challenging nature of SCADA systems and the urgent need for a framework to automate the SCADA forensics process. After discussing the challenges and available opportunities, the authors propose an architecture for an automated forensic framework for SCADA networks. Their proposed architecture takes the need for live data acquisition into consideration and is based on emerging technologies such as Multi-Agent System (MAS) and Wireless Sensor Network (WSN). The proposed framework relies on two phases: Phase one preserves live data by continuously monitoring the SCADA network through the implementation of sensors and online agents. Phase two launches offline agents to analyze the data gathered in phase one after an incident has occurred. To guarantee secure communication between the components of the proposed framework, the authors propose a trust-based security model that establishes various trust levels between components and grants communication permissions accordingly.

Valli *et al.* [41] incorporates the use of open-source tools for network analysis to provide resiliency from attackers targeting SCADA networks. This planned framework allows for an effective approach towards network security with DNP3 and MODBUS as the main protocols for examination. The research involves a methodology of allowing SCADA systems to be provided with a robust IDS system that is fulfilled by testing possible vulnerabilities of a system. After determining any form of mitigation for the vulnerability, a solution can be employed if an attack is later recognized by an intrusion detection system such as Snort.

Mittal *et al.* [29] identifies the defensive and forensic issues in SCADA systems. The author also discusses possible methods to protect SCADA systems through the use of firewalls and by controlling all data that is being transferred both in and outside the network. Incorporating live data acquisition remains a challenge, due to the difficulty of obtaining volatile memory and storing logs with the limited storage in the SCADA system.

Denton *et al.* [11] examines the GE-SRTP network protocol, a proprietary protocol developed and used by General Electric. The protocol is reverse-engineered and then analyzed in relation to the PLC requirements, allowing for the ability to change the logic of the program running on the PLC. The authors then develop a tool that can communicate with the PLC to read memory and provide access to memory registers.

6 End-Point Device Forensics

Though network-based approaches to traditional forensics covers many methods of potential device or endpoint compromise, it is by no means exhaustive. Similarly, SCADA systems cannot be protected, monitored, and remediated by network-level defenses and analysis tools alone. In order to provide researchers and practitioners with an understanding and view into the current state-of-the-art beyond network-level forensics for SCADA, we now cover recent advances on device-specific approaches to SCADA forensics.

Gougeon*et al.* [20] argues that interpreting the data stored on embedded device is especially useful for forensic investigation. In many cases the data on these devices can be obtained with no authentication using the API of the device or simply by sniffing a genuine communication. However, the dumped raw data is not easy to interpret since it is usually a mix of cryptographic material and meaningful information. The authors introduce a statistical and automatic recognition technique that can distinguish meaningful information from cryptographic material. The proposed memory carving technique performs differential analysis by comparing dumps of different devices belonging to the same application and is based on machine learning method called "boosting". The proposed approach was applied on EMV based dumps, and Calypso-based dumps. The authors claim 99.8% recognition of meaningful data.

Junejo *et al.* [22] utilizes the Secure Water Treatment (SWaT) testbed to detect vulnerabilities within PLCs. To ensure that a system has not been compromised, a machine learning based intrusion detection application is developed. The states from sensors and actuators in the associated testbed are recorded once every second for a number of hours and saved into the historian. The dataset gathered is then divided into the training set and testing data set. Ten unique attacks are used on both sets, defining the proper states that the actuators should provide. One critical issue for the testbed is the possibility of zero-day attacks.

Yau *et al.* [48] tackles the challenge of the varying architectures of PLC by incorporating a one-class support vector machine (OCSVM), a semi-supervised machine learning algorithm, in their model. The algorithm is used to accurately determine abnormal behavior from the given PLC. OCSVM is incorporated by classifying anomalous behavior from a trained model of normal operations. Understanding the structure of the program for multiple PLC architectures is another challenge that is solved by obtaining certain memory addresses along with associated timestamps from the PLC while in operation. The collected values are then used to train the model and ultimately pinpoint anomalous events.

Chan *et al.* [8] demonstrates that the Siemens PLC logging system provides detailed information about event activities for forensic investigations. However, the authors explain that the system only works under two conditions. First, the incidents must be created by a workstation that runs the proprietary Siemens TIA Portal. Second, the workstation with Siemens TIA Portal must not be compromised; otherwise, the logging system cannot be trusted.

Yau *et al.* [46] runs two experiments on a S7-1200 PLC using two programming applications that transforms ladder logic into a series of Boolean "detection rules" which can be verified by reading internal state and checking to see if the rules hold. The paper suggests and claims that monitoring only the "important" variables is the solution. The authors explain that the proposed method may be vulnerable to race conditions due to one-at-a-time retrieval of information. Additionally, the method often fails when there are many ladder rungs and requires continuous connectivity to a PC that generates the logs with the results of the detection rules.

Schlegel *et al.* [31] addresses how incidents related to the security of industrial control systems are not efficient and usually result in the loss of valuable data. The authors propose a framework used to automate a forensic investigation. This can be done by listing all packages installed on a machine, analyzing the machine during earlier periods of time to determine any file modifications, and verifying the integrity of installed software. The framework

operates by hashing files that are used on the Industrial Forensics Analysis Tool (IFAT), which incorporates a database that reports any matches from hashes already gathered.

Wu *et al.* [45] describes a forensics model for SCADA systems that can be used to effectively gather and analyze data from hardware. The forensic process consists of preserving, identifying, extracting, and documenting the digital evidence. A Siemens S7 PLC is used to demonstrate how hardware can be monitored by observing the changes of values in certain memory addresses over time.

McMinn *et al.* [28] attempts to enhance PLC firmware security by developing a verification tool that extracts the firmware by capturing the serial data during firmware upload and compares the captured firmware to a known benign baseline version. The authors claim that their developed tool does not require any modification to the SCADA system, and can be implemented on various platforms and architectures. The tool developed is also capable of creating a protocol profile that is then used to emulate future communication to replay captured data and analyze firmware without the presence of a PLC.

Senthivel *et al.* [32] the authors reverse-engineer the Programmable Controller Communication Commands (PCCC) protocol to recognize the information being relayed. The tool developed in conjunction with this paper, Cutter, is a PCCC parser that can obtain digital artifacts which, when analyzed, can construct into its associated files. The files can then be compared with the baseline files for forensic investigations in determining whether or not a PLC was compromised.

Breeuwsma *et al.* [5] extracts raw memory dumps of end-point device current state through the use of the Joint Test Access Group (JTAG) port. The extracted memory dump can be analyzed offline without interrupting the SCADA system functionality.

7 Discussion

This section presents a discussion of the key trends we uncovered and suggests future directions for digital forensics research in the SCADA domain. In particular, we believe that although much work has been done to illustrate the challenges of effective forensics in a SCADA environment, the community needs to push towards developing more practical, experimentally tested, and generally applicable tools and techniques. Our recommendations to the security community fall into *three categories*: develop broad applicable frameworks, build device-specific forensic tools, and follow the principle of generality when designing and implementing new forensic systems. We suggest that developed general frameworks can be later customized by integrating proposed device specific tools.

7.1 Develop Broad Applicable Frameworks

Our review indicates major efforts by security researchers in defining the challenges of applying traditional digital forensics to SCADA systems via a number of frameworks and general methodologies [3, 4, 6, 7, 9, 13, 15, 36, 38, 43, 47]. However, the majority of these frameworks suffer from being too high-level or lack practical evaluation. For the frameworks with case studies or experimental methodologies, there are often not immediate paths forward for building generally useful tools to accomplish the goals claimed by implementing the proposed framework. Nonetheless, the frameworks and methodologies in literature do enumerate many already encountered or potential real-world problems, research challenges, and human factors standing in the way to realizing a widespread implementation of security for SCADA systems. Moving forward, we suggest researchers continue to develop more broadly applicable frameworks, and that these frameworks enumerate more real-world use cases while also producing forensic tools and artifacts for use by other researchers and practitioners.

7.2 Building Device-Specific Forensics Tools and Techniques

Beyond gathering network-based forensic data, the analysis of the content and state of SCADA field devices must be factored into the forensics process to achieve a holistic approach to security. We have highlighted papers that target device specific volatile memory or memory addresses [7, 11, 20, 45, 48], device sensor state and ladder-logic [6, 22, 46], and analysis of device events, packages, and firmware [6, 8, 31, 49]. Future work should continue to push forward methods to analyze specific device state, including memory, firmware, packages, and other forensic data, rather than only targeting the network communications.

7.3 Towards General SCADA Forensics Tooling

Given the magnitude of manufacturers, OEMs, distributors, and software vendors in the critical infrastructure space, building a general forensics toolset similar to digital forensics for IT infrastructure remains a constant challenge. The challenge faced by security researchers in SCADA forensics is not unlike the wide and disparate set of devices mobile device forensic scientists continue to face: proprietary OSes, diverse manufacturers, and lack of technical standards. A goal should be to construct security primitives, forensic tooling, and methodologies that apply to many devices and protocols. We have shown here that researchers have taken the first steps by developing tools and techniques for specific systems, from OEMs such as GE [11] to Siemens [8, 25, 45, 47]. To truly unite the versatile and comprehensive toolsets available for IT systems, forensics researchers should strive to find unifying principles and methods to build tools that function for multiple devices and communication protocols.

8 Conclusion

With the rise of attacks against critical infrastructure, SCADA environments, and industrial control systems, security practitioners must leverage digital forensics in increasingly complex ways. By collecting, aggregating, and analyzing forensics data, breaches and attacks are able to be discovered and remediated. However, there exists a significant gap in the complexity, generality, and versatility of forensics tools, techniques, and methodologies for SCADA environments compared to the realm of IT-based forensics. To enable researchers to fill this gap, we have provided a road-map to the challenges that lie ahead, the existing frameworks for approaching SCADA forensics, and the current state-of-the-art device and network-specific tools. To that end, this is *first* survey on digital forensics applied to SCADA systems. In general, we suggest researchers continue to focus on building general tooling for SCADA forensics, extend their work beyond high-level, architectural frameworks, and focus on enabling forensics for SCADA field devices beyond the network communications alone.

Acknowledgements

This work was completed as part of the US Department of Energy Cybersecurity for Energy Delivery Systems (CEDS) program at Oak Ridge National Laboratory. CEDS is a program under the DOE Office of Science. Oak Ridge National Laboratory is managed by UT-Battelle, LLC for the US Department of Energy under contract DE-AC05-00OR22725.

References

[1] U Adhikari, T Morris, and S Pan Grid. 2017. WAMS Cyber-Physical Test Bed for Power System, Cybersecurity Study, and Data Mining. *IEEE Transactions on Smart Grid* (2017).

[2] Irfan Ahmed, Sebastian Obermeier, Martin Naedele, and Golden G Richard III. 2012. SCADA Systems: Challenges for Forensic Investigators. *Computer* 45, 12 (2012), 44–51.

[3] Irfan Ahmed, Sebastian Obermeier, Sneha Sudhakaran, and Vassil Roussev. 2017. Programmable Logic Controller Forensics. *IEEE Security & Privacy* 15, 6 (2017), 18–24.

[4] Molly Betts, Joseph Stirland, Funminiyi Olajide, Kevin Jones, and Helge Janicke. 2016. Developing a state of the art methodology & toolkit for ICS SCADA forensics. *The International Conference on Information Security and Cyber Forensics* (2016).

[5] MF Breeuwsma. 2006. Forensic imaging of embedded systems using JTAG (boundary-scan). *digital investigation* 3, 1 (2006), 32–42.

[6] Alvaro A Cardenas, Tanya Roosta, and Shankar Sastry. 2009. Rethinking security properties, threat models, and the design space in sensor networks: A case study in SCADA systems. *Ad Hoc Networks* 7, 8 (2009), 1434–1447.

[7] Chun-Fai Chan, Kam-Pui Chow, Siu-Ming Yiu, and Ken Yau. 2018. Enhancing the Security and Forensic Capabilities of Programmable Logic Controllers. *IFIP Int. Conf. Digital Forensics* 532, Chapter 19 (2018), 351–367.

[8] Raymond Chan and Kam-Pui Chow. 2016. Forensic Analysis of a Siemens Programmable Logic Controller. *CrWuitical Infrastructure Protection* 485, Chapter 7 (2016), 117–130.

[9] Rodrigo Chandia, Jesús González 0004, Tim Kilpatrick, Mauricio Papa, and Sujeet Shenoi. 2007. Security Strategies for SCADA Networks. *Critical Infrastructure Protection* 253, Chapter 9 (2007), 117–131.

[10] Dorothy E Denning. 2012. Stuxnet - What Has Changed? *Future Internet* (2012).

[11] George Denton, Filip Karpisek, Frank Breitinger, and Ibrahim Baggili. 2017. Leveraging the SRTP protocol for over-the-network memory acquisition of a GE Fanuc Series 90-30. *Digital Investigation* 22 (2017), S26–S38.

[12] G Devarajan. 2007. Unraveling SCADA protocols: Using sulley fuzzer. *DEFCON Conference* (2007).

[13] Peter Eden, Andrew Blyth, Pete Burnap, Yulia Cherdantseva, Kevin Jones, Hugh Soulsby, and Kristan Stoddart. 2016. Forensic Readiness for SCADA/ICS Incident Response. *ICS-CSR* (2016).

[14] Peter Eden, Andrew Blyth, Kevin Jones, Hugh Soulsby, Pete Burnap, Yulia Cherdantseva, and Kristan Stoddart. 2017. SCADA System Forensic Analysis Within IIoT. In *Cybersecurity for Industry 4.0*. Springer, Cham, 73–101.

[15] Peter Eden, Pete Burnap, Andrew Blyth, Kevin Jones, Hugh Soulsby, and Yulia Cherdantseva. 2015. A Forensic Taxonomy of SCADA Systems and Approach to Incident Response. In *3rd International Symposium for ICS & SCADA Cyber Security Research 2015*. BCS Learning &Development Ltd, 42–51.

[16] Mohamed Elhoseny, Abbas Hosny, Aboul Ella Hassanien, Khan Muhammad, and Arun Kumar Sangaiah. 2017. Secure Automated Forensic Investigation for Sustainable Critical Infrastructures Compliant with Green Computing Requirements. *IEEE Transactions on Sustainable Computing* (2017), 1–1.

[17] FireEye. 2018. RedLine. https://www.fireeye.com/services/freeware/redline.html

[18] Asem Ghaleb, Sami Zhioua, and Ahmad Almulhem. 2016. SCADA-SST: a SCADA security testbed. In *2016 World Congress on Industrial Control Systems Security (WCICSS)*. IEEE, 1–6.

[19] Google. 2018. Rekall Forensics. http://rekall-forensic.com

[20] Thomas Gougeon, Morgan Barbier, Patrick Lacharme, Gildas Avoine, and Christophe Rosenberger. 2016. Memory Carving in Embedded Devices - Separate the Wheat from the Chaff. *ACNS* 9696, 3 (2016), 592–608.

[21] Asif Iqbal, Mathias Ekstedt, and Hanan Alobaidli. 2017. Exploratory studies into forensic logs for criminal investigation using case studies in industrial control systems in the power sector. *BigData* (2017), 3657–3661.

[22] Khurum Nazir Junejo and Jonathan Goh. 2016. Behaviour-Based Attack Detection and Classification in Cyber Physical Systems Using Machine Learning. *CPSS at AsiaCCS* (2016), 34–43.

[23] Suleman Khan, Abdullah Gani, Ainuddin Wahid Abdul Wahab, Muhammad Shiraz, and Iftikhar Ahmad. 2016. Network forensics: Review, taxonomy, and open challenges. *Journal of Network and Computer Applications* 66 (2016), 214–235.

[24] Tim Kilpatrick, Jesus Gonzalez, Rodrigo Chandia, Mauricio Papa, and Sujeet Shenoi. 2006. An Architecture for SCADA Network Forensics. In *Advances in Digital Forensics II*. Springer New York, Boston, MA, 273–285.

[25] Amit Kleinmann and Avishai Wool. 2014. Accurate Modeling of The Siemens S7 SCADA Protocol For Intrusion Detection And Digital Forensic. *JDFSL* (2014).

[26] Ligh, Michael Hale, Case, Andrew, Levy, Jamie, and Walters, Aaron. 2014. *The Art of Memory Forensics*. John Wiley & Sons.

[27] Marie-Helen Maras and Others. 2015. *Computer Forensics*. Jones and Bartlett Learning.

[28] Lucille McMinn and Jonathan Butts. 2012. A Firmware Verification Tool for Programmable Logic Controllers. *Critical Infrastructure Protection* 390, Chapter 5 (2012), 59–69.

[29] Sandeep Mittal. 2015. The Issues in Cyber-Defence and Cyber-Forensics of the SCADA Systems. (2015).

[30] Bill Nelson, Amelia Phillips, and Christopher Steuart. 2014. *Guide to Computer Forensics and Investigations*. Cengage Learning.

[31] Roman Schlegel, Ana Hristova, and Sebastian Obermeier. 2015. A Framework for Incident Response in Industrial Control Systems. *SECRYPT* (2015), 178–185.

[32] Saranyan Senthivel, Irfan Ahmed, and Vassil Roussev. 2017. SCADA network forensics of the PCCC protocol. *Digital Investigation* 22 (2017), S57–S65.

[33] Jill Slay and Elena Sitnikova. 2009. The Development of a Generic Framework for the Forensic Analysis of SCADA and Process Control Systems. *e-Forensics* 8, Chapter 9 (2009), 77–82.

[34] Jared M Smith, Elliot Greenlee, and Aaron Ferber. 2017. DEMO: Akatosh - Automated Cyber Incident Verification and Impact Analysis. *CCS* (2017), 2463–2465.

[35] Somayeh Soltani and Seyed Amin Hosseini Seno. 2017. A survey on digital evidence collection and analysis. In *International Conference on Computer and Knowledge Engineering (ICCKE)*. IEEE, 247–253.

[36] Joe Stirland, Kevin Jones, Helge Janicke, and Tina Wu. 2014. Developing Cyber Forensics for SCADA Industrial Control Systems. (2014), 98–111.

[37] Johannes Stüttgen, Stefan Vömel, and Michael Denzel. 2015. Acquisition and analysis of compromised firmware using memory forensics. *Digital Investigation* 12 (2015), S50–S60.

[38] Pedro N. Taveras. 2013. Scada Live Forensics: Real Time Data Acquisition Process to Detect, Prevent or Evaluate Critical Situations. *European Scientific Journal, ESJ* 9, 21 (2013).

[39] Dell Threat Intelligence Team. 2013. Wiper Malware Threat Analysis. *Secureworks* (2013). https://www.secureworks.com/research/wiper-malware-analysis-attacking-korean-financial-sector

[40] The Volatility Foundation. 2018. Volatility Foundation. https://www.volatilityfoundation.org

[41] Craig Valli. 2009. Snort IDS for SCADA Networks. *Security and Management* (2009).

[42] R. M. van der Knijff. 2014. Control systems/SCADA forensics, what's the difference? *Elsevier Digital Investigation* (2014).

[43] Pieter Van Vliet, M-T Kechadi, and Nhien-An Le-Khac. 2016. Forensics in Industrial Control System: A Case Study. *arXiv.org* (2016). arXiv:cs.CR/1611.01754v1

[44] R. B. Vaughn Jr and T Morris. 2016. Addressing Critical Industrial Control System Cyber Security Concerns via High Fidelity Simulation. *Proceedings of the 11th Annual Cyber and Information Security Research Conferenc* (2016).

[45] T Wu, JFP Disso, K Jones, A Campos Proceedings of the 1st, and 2013. [n. d.]. Towards a SCADA forensics architecture. *ewic.bcs.org* ([n. d.]).

[46] Ken Yau and Kam-Pui Chow. 2015. PLC Forensics Based on Control Program Logic Change Detection. *JDFSL* (2015).

[47] Ken Yau, Kam-Pui Chow, and Siu-Ming Yiu. 2018. A Forensic Logging System for Siemens Programmable Logic Controllers. *IFIP Int. Conf. Digital Forensics* 532, Chapter 18 (2018), 331–349.

[48] Ken Yau, Kam-Pui Chow, Siu-Ming Yiu, and Chun-Fai Chan. 2017. Detecting anomalous behavior of PLC using semi-supervised machine learning. *CNS* (2017), 580–585.

[49] Muhammad Sharjeel Zareen, Adeela Waqar, and Baber Aslam. 2013. Digital forensics: Latest challenges and response. In *2013 2nd National Conference on Information Assurance (NCIA)*. IEEE, 21–29.