SCADA Systems: Challenges for Forensic Investigators

Irfan Ahmed, *University of New Orleans*
Sebastian Obermeier and Martin Naedele, *ABB Corporate Research*
Golden G. Richard III, *University of New Orleans*

When security incidents occur, several challenges exist for conducting an effective forensic investigation of SCADA systems, which run 24/7 to control and monitor industrial and infrastructure processes.

An industrial automation and control system is a set of devices that regulate the behavior of physical processes. For example, a thermostat is a simple control system that senses the temperature and turns a heater on or off to maintain the temperature at a set point. These systems are used to monitor and control industrial and infrastructure processes such as chemical plant and oil refinery operations, electricity generation and distribution, and water management.

A control system that is spread over a wide area and can supervise its individual components is often called a supervisory control and data acquisition (SCADA) system.[1] However, here we use the term SCADA to refer to all kinds of control systems that share a common key characteristic: they are connected to physical processes and thus need to be continuously available and able to respond within a deterministic time bound.

Early SCADA systems were intended to run as isolated networks, not connected to the Internet, and thus did not require any specific cybersecurity mechanisms. These systems consisted of simple I/O devices that transmitted the signals between master and remote terminal units. In recent years, SCADA systems have evolved to communicate over public IP networks.[2] Some are also connected to a corporate intranet or directly to the Internet to seamlessly integrate SCADA data with external information such as corporate email or weather data.

The integration of SCADA systems within a much wider network brings threats that were unimagined at the time these systems were conceived. During the past decade, vendors, asset owners, and regulators recognized this growing concern and began to address it through new laws and various security mechanisms, processes, and standards.[3]

The discoveries in the wild of Stuxnet in June 2010 and Flame in May 2012 were additional eye-openers for SCADA owners and operators. Stuxnet, the first known malware designed to target automation systems, has infected 50,000 to 100,000 computers worldwide,[4] while Flame is a cyberespionage tool an order of magnitude more sophisticated than Stuxnet.[5]

SCADA ARCHITECTURE

As Figure 1 shows, a typical SCADA system for controlling infrastructures for utilities such as power, gas, oil, or water generally consists of a control center and numerous field sites. The sites are distributed over a wide geographical area and are connected to the control center by different communication media such as satellites, wide

 Published by the IEEE Computer Society

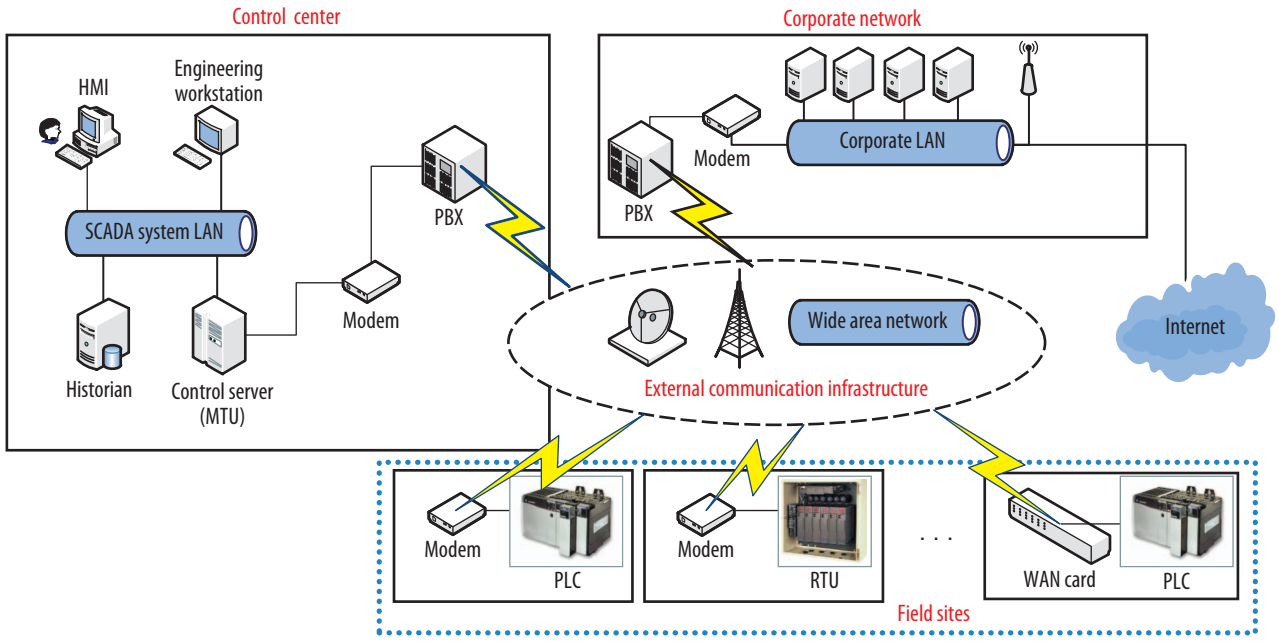

Figure 1. Simplified logical view of a typical supervisory control and data acquisition (SCADA) architecture.

area networks (WANs), and radio, microwave, or cellular networks. Field sites are equipped with devices such as programmable logic controllers (PLCs) or remote terminal units (RTUs) that control the on-site machines and periodically send information about the state of the field equipment to the control center.

The control center is the SCADA system's hub. Its major components include a human-machine interface (HMI), the database management system (historian), and the server or master terminal unit (MTU). The MTU initiates all communication with field sites and receives the data sent from the field devices. If necessary, it then preprocesses the data and sends it to the historian for archiving. The HMI presents information to the human operator.

FORENSICS FOR SCADA SYSTEMS

Digital forensics is an aspect of cyberdefense that becomes essential in the event of a security breach.[6] It can generally be defined as the collection and analysis of digital data from different sources such as computer systems, storage devices, and network streams to investigate the causes and consequences of an intrusion or some other incident. If investigators find traces of a crime such as unauthorized network access or theft of a digital file, they can present such data as evidence in a court of law. Digital forensics is also commonly used in internal corporate investigations to help limit the possibility of an incident occurring again in the future.

The recent attacks against SCADA systems by powerful malware such as Stuxnet and Flame highlight the need for forensic investigations to improve cyberdefenses against both internal and external perpetrators with malicious intent and to thwart entities that try to sabotage a country's critical infrastructure.[7] In addition to playing a vital role in developing a protection strategy for SCADA systems and assisting in the identification and prosecution of attackers, digital forensics can help deal with nonmalicious but harmful events such as malfunctioning hard disks or other hardware by performing a deep analysis of the underlying SCADA IT system.

A forensic investigation can be the most effective, if not the only, way to answer many questions about an incident. For example, consider a scenario in which malware attacks a SCADA system, causing it to malfunction:

- A virus scan revealed that the Java cache contains a known exploit. Was the exploit successful? What payload does it have? Is this what compromised the system?
- How can the operator clean the system and reliably return it to a known good state without having to shut down the complete system?
- An operator has installed a suspicious, untrusted application downloaded from the Internet. Did that application change components that are important for the system's stable operation?

From a forensics perspective, a SCADA system can be seen in different layers based on the connectivity of the various components with each other as well as with other networks such as the Internet.[1] In Figure 2, layer 0 contains

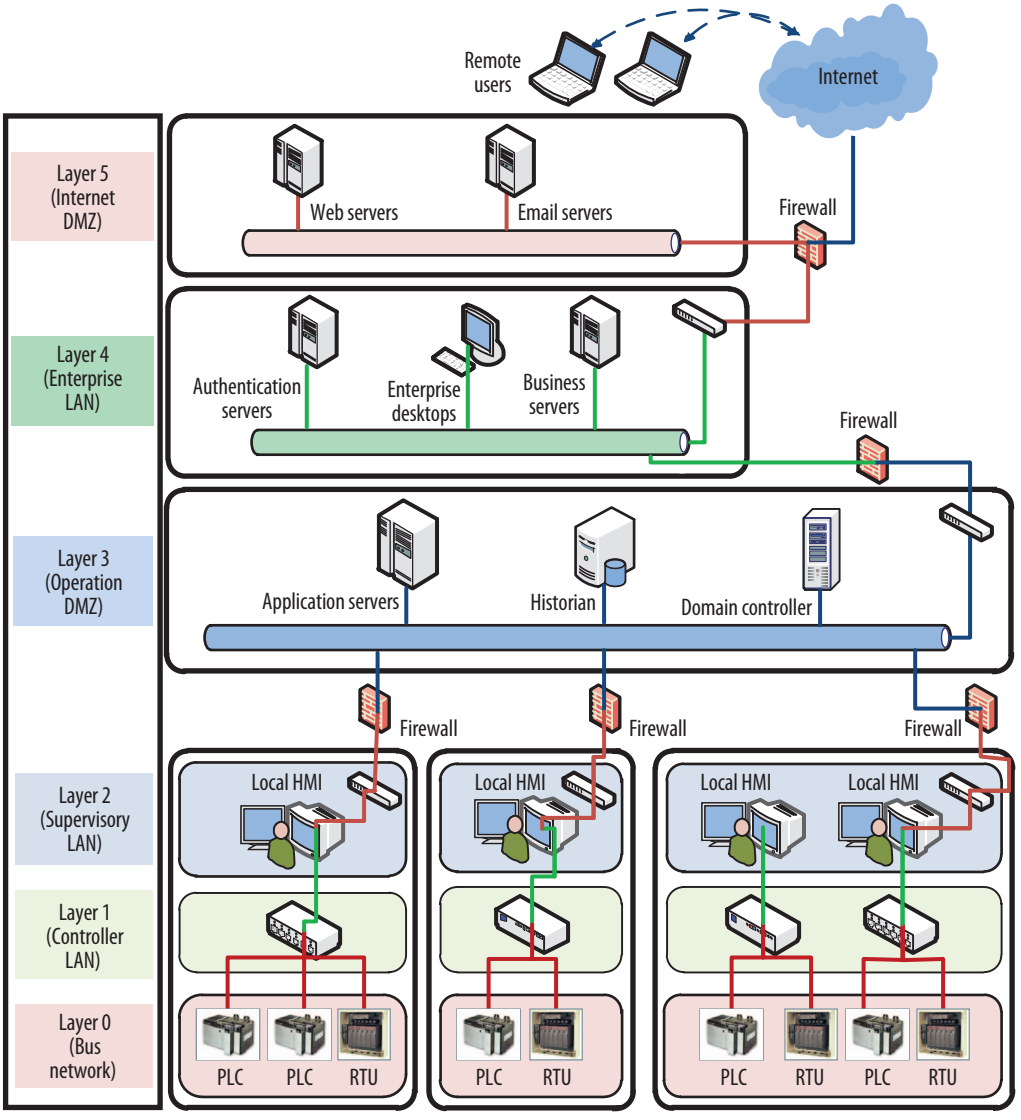

Figure 2. SCADA system layers. Most forensic analysis involves layers 0-2, which contain the components that control the underlying industrial processes.

the individual field devices connected via a bus network. Layer 1 has controllers that receive input signals from the field devices and other controllers upon which they perform operations to steer the field devices by sending output signals to them. Layer 2 consists of the supervisory network, typically a local network connected to the lower layers for specific operations such as showing the current monitoring state at the HMI. Layer 3 is the operation DMZ, where historians, domain controllers, and application servers are located. Layers 4 and 5 correspond to the enterprise IT networks, in which the enterprise desktops and business servers operate.

Most forensic analyses of SCADA systems involve layers 0-2, as they contain the components that control the underlying industrial processes. However, the analysis can extend to layers 3-5 if needed. Here, we focus on the first three layers.

LIVE FORENSICS

Because a SCADA system must be continuously operational, a forensic investigator cannot turn it off to capture and analyze data.[8] In this case, *live forensics* is a viable solution for a digital investigation.[9] A relatively new and emerging field in digital forensics, live forensics involves performing data acquisition and analysis on a running system. However, the critical nature of SCADA systems and their 24/7 availability requirement dictate that forensic investigators spend as little time as possible on a live SCADA

system. Consequently, the investigators typically acquire live data and subsequently analyze it offline.

Live data acquisition includes both volatile data such as the contents of physical memory and nonvolatile data such as data stored on a hard disk. This differs from traditional dead data acquisition, which involves first taking the system offline, losing all volatile data. However, volatile data plays a significant role in an effective forensic investigation. For instance, volatile data in physical memory contains information about the system's current state, such as the number of open network connections, process information, and encryption keys.

Live data acquisition challenges

Because volatile data changes continuously on a running system, capturing live data presents two key challenges for forensic investigators.

Early data acquisition after an incident. Live data must be acquired as quickly as possible after an incident to capture any of the incident's traces before processes or services on the running system overwrite useful volatile data—for example, data about recently unloaded kernel modules or drivers.

Digital evidence validity. Digital data might not be admissible in court if its integrity is violated. The intention is to prevent the malicious manufacturing of evidence against an innocent person and to avoid errors while handling evidence in the course of an investigation. Forensic investigators normally prove the integrity of evidence by computing a cryptographic hash of the actual evidence on the compromised system and its acquired copy, which is used for all examinations and analysis. If, however, the compromised system remains live, the data's state might change between the copying and the hash calculation, rendering hashing ineffective as an integrity check.

This also creates an inconsistent data image that does not accurately represent the state when data acquisition starts or after it ends, which can cause difficulty in analyzing the acquired data. For example, due to data inconsistency, sometimes an operating system in the disk image cannot boot for experimental analysis.

Live data acquisition on SCADA systems

It is still unclear how to acquire live data on a SCADA system in a way that minimizes risk to the system's services. To the best of our knowledge, no guidelines for accomplishing this are currently available. However, safe data acquisition should be possible under many circumstances.

Specifically, SCADA systems typically have a primary and a backup system. When the primary system is broken or malfunctioning, operators switch to the backup system.[10] Forensic investigators could leverage this capability by switching the system to the backup and performing live data acquisition on the infected system without worrying about the availability of SCADA services. However, this approach might not be feasible when the malware that has infected the primary system has also infected the backup system. That scenario might even demand immediate recovery if the SCADA owners and operators decide that the incident can jeopardize the system's normal functionality. This usually results in flushing all the infected system components and bringing them back to their normal state, which would not allow sufficient time for the investigator to perform data acquisition.

FORENSIC CHALLENGES IN SCADA SYSTEMS

Beyond the challenges of live data acquisition, forensic investigators must deal with various problems arising from SCADA systems' unique features, which prevent directly applying contemporary forensic tools and techniques.

Deterministic network traffic

Network traffic in SCADA systems is deterministic in that a system component communicates with other system components in a predefined manner. This contrasts with office IT systems, in which desktop machines and servers communicate based on requests in a nondeterministic way.[11]

It is still unclear how to acquire live data on a SCADA system in a way that minimizes risk to the system's services.

Based on this deterministic behavior, administrators can apply stringent rules to harden the system's security, with any nondeterministic behavior flagged as an anomaly. For instance, an intrusion detection system might be configured to consider a specific communication pattern as normal.[12] Security tools that expect such deterministic behavior might raise false alarms or prevent forensic tools from operating properly. For example, a firewall might have strict rules that allow communication between specific SCADA components but disallow communication between the investigator's machine and SCADA components during data acquisition.

Customized operating system kernels

A SCADA system can have customized kernels running on its components to achieve better performance, support critical applications, and so on, despite the fact that updating such kernels is difficult. For example, PatriotSCADA (www.sage-inc.com/cgi-bin/products_scadasentry.php) is a firewall solution for SCADA networks that uses a customized

Linux kernel to enforce access control and role-based security for every request in the kernel.

However, data acquisition tools might not run on a customized kernel unless they are compatible with each other. For example, the DD disk copy tool might require loading the fmem kernel module (in Linux) to access the physical memory through the device /dev/fmem (which the module creates) if the regular /dev/mem device in Linux is not accessible. Until the module is compiled with the customized kernel, the module might not load into the kernel.

Resource-constrained devices

The availability of SCADA services also depends on the adequacy of system resources: CPU, memory, I/O, and so on. Some system components run on legacy/proprietary hardware and operating systems that might have been deployed for more than 10 years, have moderate computing capabilities compared to modern systems, and have limited or no vendor support.[13] Moreover, field devices such as RTUs and PLCs are generally resource constrained. SCADA systems with such limited resources demand lightweight data acquisition tools.

The forensic process can be improved in SCADA systems through preparedness and the selection of appropriate tools.

Inadequate logging

Collecting logs of events soon after an incident is crucial for successful forensic investigation. However, SCADA systems' logging capabilities are geared toward process disturbances, not security breaches, and thus are often inadequate.[13] In such cases, administrators must improve historical visibility in SCADA system components.

Extensive lower-layer data

Capturing and analyzing data on lower layers in SCADA systems is challenging due to the large amount of data that individual sensors generate. In electricity grids, for example, sensors can carry out up to 4,000 measurements per second.[14]

FORENSIC TOOLS AND METHODOLOGIES

The forensic process can be improved in SCADA systems through preparedness and the selection of appropriate tools.

Data acquisition plan

To help capture the most relevant data related to an incident, forensic investigators should craft a data acquisition plan that accurately documents the SCADA system's design, its unique features, the application data flow, and temporary and permanent data storage locations. The plan should also specify what data to acquire for different types of incidents.

Mark Fabro and Eric Cornelius[13] proposed guidelines for creating such a plan in three phases. The first phase involves identifying the system environment and its unique characteristics, including whether the system has modern computing capabilities, is still fully supported by vendors, uses contemporary operating systems, and has continuing support for any open source components. The second phase consists of defining environment-specific requirements such as the impact of vendor solutions on operating systems. The third phase consists of the identification and collection of data, such as activity and transaction logs.

Data acquisition monitoring

During forensic acquisition, no matter how careful an investigator is when copying data, there is always a risk of upsetting the availability of SCADA services. However, this risk can be mitigated by monitoring the availability of system services during data acquisition so that the process can be stopped in case of any serious perturbation. A monitoring tool can facilitate this process by detecting the perturbation as soon as it occurs and automating the response, to avoid any serious damage to the system. EnCase Cybersecurity (www.guidancesoftware.com/encase-cybersecurity.htm) is an exemplar of a data acquisition monitoring tool that can be integrated into management systems and configured to respond automatically to alerts or events.

Lightweight data acquisition

Data acquisition tools should have a minimal impact so that adequate system resources are available for SCADA services to work properly.

To get a preliminary idea of how resource intensive data acquisition tools are, we ran three well-known versions of the DD tool—WinDD (www.moonsols.com/windows-memory-toolkit), George Garner's DD (www.gmgsystemsinc.com/fau), and DD for Linux variants—to acquire the entire physical memory and hard-disk data of a computer and recorded the computer's resource consumption during data acquisition for analysis. We acquired the data using Garner's Netcat tool over a 100-Mbps network, a preferred method for forensic investigations.

To emulate a resource-constrained system, we used a PC with an Intel Celeron 1.7-GHz CPU, 384 Mbytes of RAM, and a 40-Gbyte hard drive running at 7,200 rpm. We used two different operating systems, Windows XP Service Pack 2 (SP2) and an Intel Centos 4, for our initial experiments. We kept the machine idle to leave all possible system resources available for the data acquisition tools so that they could

exploit the resources at their full capacity without any constraints. For data acquisition over the network, we directly connected the PC through a crossover cable with the investigative machine, where the data was transferred, to avoid the overhead of packet switching or routing.

The investigative machine was a modern computer with an Intel Core 2 Duo CPU, 4 Gbytes of RAM, and a 300-Gbyte hard drive running at 15,000 rpm, which is unlikely to have caused any performance bottleneck in the data acquisition tools.

Table 1. Resource consumption of data acquisition tools.

Tool	Operating system	Device	CPU idle time (percent)[1]	Free physical memory (percent)[2]	Disk queue length[3]
WinDD	Windows XP (SP2)	Physical memory	90.72	75.60	0.03
Garner's DD	Windows XP (SP2)	Hard drive	27.49	74.01	0.72
DD (on Linux variants)	Centos 4	Physical memory	51.98	79.69	0.00
DD (on Linux variants)	Centos 4	Hard drive	0.646	71.14	0.805

[1] CPU idle time: average percentage of time during data acquisition that the CPU was idle
[2] Free physical memory: average percentage of free physical memory during data acquisition
[3] Disk queue length: average number of (read and write) requests outstanding on the hard disk during data acquisition

As Table 1 shows, the tools did not exhaust the system resources for data acquisition per se, and might consume less with better hardware than we used. However, the results do not guarantee that the tools are compatible with a particular SCADA environment and would not significantly impact services during the data acquisition process until they are run and tested on that environment or its equivalent, such as a production environment testbed. Moreover, the tools not included in the experiments might not necessarily show a similar performance impact.

Plug-ins for forensic analysis tools

To the best of our knowledge, state-of-the-art forensic analysis tools do not support the unique features of diverse SCADA environments, including protocols and numerous applications' proprietary log formats. Researchers must develop plug-ins or modules for contemporary forensic tools to augment analysis of SCADA systems.

RESEARCH CHALLENGES AND TRENDS

The heightened focus of governments worldwide on protecting their critical infrastructures has led to increased research funding for this purpose. However, the critical nature of SCADA systems also imposes limitations on the research community.

Research challenges

While a security incident in an office environment might lead, at worst, to significant monetary loss or service disruption, breaches of SCADA systems can have dangerous consequences for both human life and the environment.[15-17] In addition, performance requirements for SCADA protection systems have an impact on some security features. For example, in certain use cases, the overhead induced by asymmetric cryptography is intolerable.[18]

Thus, research in this domain should be practical and conclusive, which requires the availability of SCADA systems for research purposes. However, building real systems is expensive. To deal with this problem, the SCADA research community usually opts for the following approaches, each of which has its own merits and limitations.

Using simulators. Some commercial simulators, such as Opal Software's simSCADA (www.opalsoftware.com.au/index.php?option=com_content&view=article&id=35&Itemid=67), provide a virtual environment for studying SCADA systems. They are mostly used to imitate the network traffic between field devices and MTUs and are effective at reducing hardware purchase and installation costs. However, simulators are subject to errors and thus typically do not provide the same level of confidence that a real system would.

Building small-scale SCADA systems. Government and academic researchers use commercial hardware and software to build laboratory-scale testbeds of some SCADA systems such as industrial blowers, gas pipelines, power grids, and petroleum storage tanks. For example, Mississippi State University has a testbed for studying and learning about multiple industrial control systems.[19] The Idaho National Laboratory has a testbed of a full-scale electrical grid that is dedicated to control system cybersecurity assessment, standards improvements, and training (www.inl.gov/research/national-supervisory-control-and-data-acquisition-test-bed).

Industry collaboration. When applying for project funding, researchers try to engage SCADA owners and operators as industrial partners. The terms of agreement for a project usually involve technical assistance, facility access (at least to the testbed the operators use for testing application patches from vendors), and financial support. Industrial collaboration provides close access to real-world SCADA systems and the technical personnel who actually experience the problems and understand the limitations of their particular system.

However, industry collaboration is often difficult to achieve due to the critical nature of SCADA systems, which discourages owners and operators from cooperating with the research community to prevent information leakage.

This creates a gap between the research community's efforts and resolving the problems that SCADA owners and operators face.

In this situation, governments are often in the best position to play a mediator role and help reduce this gap. For example, the Australian government regularly organizes community-of-interest meetings to provide a platform for discussions among SCADA owners and operators, SCADA vendors, and researchers from academia.

Research trends

Thus far, the research community has mainly focused on SCADA system security. However, there has also been limited work on the forensic investigation of SCADA systems.

Tim Kilpatrick and colleagues[11,20,21] proposed an architecture for capturing and subsequently analyzing sensor data and control actions in a SCADA network. Agents placed at strategic locations within the network capture local traffic and forward a relevant portion of packets, called a synopsis, to a data warehouse. After analyzing a synopsis, the data warehouse creates its digital signature and stores it with the synopsis in the agent's designated storage area. A relational database and query mechanisms support forensic investigations. The modular agent design and configurable synopsis engines accommodate diverse SCADA protocols, some of their implementation variations, and subsets of standard or proprietary protocols. The researchers developed a prototype of the architecture based on the Modbus TCP protocol using two control devices and one HMI station.

Craig Valli[22] created a framework that produces forensically verified signatures for the Snort intrusion detection system (IDS) for known and published vulnerabilities of SCADA and control systems, enabling investigators to trace exploits during analysis. Valli first looked for vulnerability announcements or traces at Black Hat, hacker, vendor, CERT (Community Emergency Response Teams), and other relevant sites and reproduced the vulnerability scenarios. He then examined the vulnerabilities of SCADA communication protocols such as Modbus and DNP3.

Valli conducted experiments involving an attacker, a victim/target machine running SCADA software, the Snort IDS, and a network sniffer that captures all network traffic in a tcpdump binary capture file for analysis to generate Snort rules. He analyzed the exploit's modus operandi and used this to create a rule set to reduce or stop the attack. Valli later included the rule set in the Snort configuration to test its resilience under sustained attack.

According to Jill Slay and Elena Sitnikova,[23] a generic approach to forensics in SCADA systems requires a big picture view of the process that encompasses a range of technical and procedural issues at the government, industry, and academic levels.

Because of the underlying industrial processes they control, performing a forensic investigation of SCADA systems is fundamentally different from investigating corporate or home networks. The critical nature of SCADA systems demands that investigators be well trained and thoroughly understand the requirements to manage such systems. Engaging investigators early so they can become accustomed to a particular environment is highly recommended. It is also desirable to encourage SCADA system owners and operators to initiate steps that can facilitate an investigation if needed—for example, by maintaining a data acquisition plan and regularly testing data acquisition tools to ensure that they will not affect the availability of SCADA system services.

SCADA-focused forensic research is essential to address the unique challenges associated with these systems. The forensic research community must engage SCADA owners and operators and those actively working on SCADA systems to highlight research problems and develop solutions. Governments must take a more active role in organizing these efforts and helping to provide researchers with resources and suitable access to SCADA systems.

Acknowledgment

This work was supported in part by NSF grant CNS #1016807.

References

1. D. Bailey and E. Wright, *Practical SCADA for Industry*, Newnes, 2003.
2. R. Kalapatapu, "SCADA Protocols and Communication Trends," *Proc. 2004 ISA Industrial Network Security Symp.* (ISA Expo 04), Instrumentation, Systems, and Automation Soc., 2004; www.isa.org/journals/intech/TP04ISA048.pdf.
3. M. Brändle and M. Naedele, "Security for Process Control Systems: An Overview," *IEEE Security & Privacy*, Nov./Dec. 2008, pp. 24-29.
4. T.M. Chen and S. Abu-Nimeh, "Lessons from Stuxnet," *Computer*, Apr. 2011, pp. 91-93.
5. G. Keizer, "Development Timeline Key to Linking Stuxnet, Flame Malware," *Computerworld*, 30 May 2012; www.computerworld.com/s/article/9227580/Development_timeline_key_to_linking_Stuxnet_Flame_malware.
6. K. Mandia, C. Prosise, and M. Pepe, *Incident Response and Computer Forensics*, 2nd ed., McGraw-Hill/Osborne, 2003.
7. R.N. Charette, "Gone Missing: The Public Policy Debate on Unleashing the Dogs of Cyberwar," blog, 4 June 2012; http://spectrum.ieee.org/riskfactor/telecom/security/gone-missing-the-public-policy-debate-on-unleashing-the-dogs-of-cyberwar.
8. M. Naedele, "Addressing IT Security for Critical Control Systems," *Proc. 40th Hawaii Int'l Conf. System Sciences* (HICSS 07), IEEE CS, 2007; doi:10.1109/HICSS.2007.48.
9. F. Adelstein, "Live Forensics: Diagnosing Your System without Killing It First," *Comm. ACM*, Feb. 2006, pp. 63-66.
10. K. Stouffer, J. Falco, and K. Scarfone, *Guide to Industrial Control Systems (ICS) Security*, NIST special publication 800-82, Nat'l Inst. Standards and Technology, 2011; http://

csrc.nist.gov/publications/nistpubs/800-82/SP800-82-final.pdf.
11. T. Kilpatrick et al., "An Architecture for SCADA Network Forensics," *Advances in Digital Forensics II*, M.S. Olivier and S. Shenoi, eds., Springer, 2006, pp. 273-285.
12. H. Hadeli et al., "Leveraging Determinism in Industrial Control Systems for Advanced Anomaly Detection and Reliable Security Configuration," *Proc. 14th Int'l Conf. Emerging Technologies and Factory Automation* (ETFA 09), IEEE, 2009, pp. 1189-1196.
13. M. Fabro and E. Cornelius, *Recommended Practice: Creating Cyber Forensics Plans for Control Systems*, tech. report INL/EXT-08-14231, Idaho Nat'l Lab., 2008.
14. H. Kirrmann, "Seamless Redundancy: Bumpless Ethernet Redundancy for Substations with IEC 61850," *ABB Rev.*, Aug. 2010, pp. 57-61.
15. D. Dzung et al., "Security for Industrial Communication Systems," *Proc. IEEE*, June 2005, pp. 1152-1177.
16. F. Köster et al., "Collaborative Security Assessments in Embedded Systems Development—The ESSAF Framework for Structured Qualitative Analysis," *Proc. Int'l Conf. Security and Cryptography* (Secrypt 09), INSTICC Press, 2009, pp. 305-312.
17. E. Levy, "Crossover: Online Pests Plaguing the Offline World," *IEEE Security & Privacy*, Nov./Dec. 2003, pp. 71-73.
18. S. Fuloria et al., "The Protection of Substation Communications," *Proc. SCADA Security Scientific Symp.* (S4 10), 2010; www.cl.cam.ac.uk/~rja14/Papers/S4-2010.pdf.
19. T. Morris, R. Vaughn, and Y.S. Dandass, "A Testbed for SCADA Control System Cybersecurity Research and Pedagogy," *Proc. 7th Ann. Cyber Security and Information Intelligence Research Workshop* (CSIIRW 11), ACM, 2011; doi:10.1145/2179298.2179327.
20. T. Kilpatrick et al., "Forensic Analysis of SCADA Systems and Networks," *Int'l J. Security and Networks*, Feb. 2008, pp. 95-102.
21. R. Chandia et al., "Security Strategies for SCADA Networks," *Critical Infrastructure Protection*, E. Goetz and S. Shenoi, eds., Springer, 2008, pp. 117-131.
22. C. Valli, "SCADA Forensics with Snort IDS," *Proc. 2009 Int'l Conf. Security and Management* (SAM 09), CSREA Press, 2009, pp. 618-621.
23. J. Slay and E. Sitnikova, "The Development of a Generic Framework for the Forensic Analysis of SCADA and Process Control Systems," *Forensics in Telecommunications, Information, and Multimedia*, M. Sorrell, ed., Springer, 2009, pp. 77-82.

Irfan Ahmed *is a postdoctoral research associate in the Department of Computer Science at the University of New Orleans. His research interests include industrial control system security, digital forensics, and malware detection and analysis. Ahmed received a PhD in computer science from Ajou University, South Korea. Contact him at irfan.ahmed@uno.edu.*

Sebastian Obermeier *is a principal scientist at ABB Corporate Research. His research interests include IT security for industrial control systems and database technology. Obermeier received a PhD in computer science from the University of Paderborn, Germany. Contact him at sebastian.obermeier@ch.abb.com.*

Martin Naedele *is the R&D program manager for Industrial Software Systems at ABB Corporate Research. His research interests include software engineering and IT security. Naedele received a PhD in computer engineering from ETH Zurich. He is a GIAC-certified security auditor and a member of IEEE, ACM, and the International Council on Systems Engineering. Contact him at martin.naedele@ch.abb.com.*

Golden G. Richard III *is a professor of computer science and a university research professor at the University of New Orleans. His research interests include digital forensics, reverse engineering, and operating systems internals. Richard received a PhD in computer science from the Ohio State University. He is a member of IEEE, Usenix, ACM, and the American Academy of Forensic Sciences. Contact him at golden@cs.uno.edu.*