DOI: http://dx.doi.org/10.14236/ewic/ics-csr2014.9

Position Paper: Safety and Security Monitoring in ICS/SCADA Systems

Andrew Nicholson, Helge Janicke and Antonio Cau
Software Technology Research Laboratory
De Montfort University, UK
{*abn,heljanic,cau*} *@dmu.ac.uk*

Supervisory control and Data Acquisition (SCADA) systems play a core role in a nation's critical infrastructure, overseeing the monitoring and control of systems in electricity, gas supply, logistics services, banks and hospitals. Monitoring safety and security properties in industrial control system (ICS) and SCADA environments faces unique challenges not found in typical enterprise networks. Novel monitoring solutions are desirable that take into account these differences. This paper presents a new approach for monitoring safety and security properties in industrial control systems. The approach is based on verifying a formal specification of ICS/SCADA components during runtime that is capable of detecting abnormal system behaviours. The solution is miniaturised and can be deployed at various points throughout the SCADA network, making masquerading and man-in-the-middle attacks more difficult to execute successfully.

Keywords: ICS, SCADA, Critical Infrastructure, Security, Monitoring

1. INTRODUCTION

SCADA systems are responsible for the monitoring and control of a wide range of a nation's critical infrastructure, including electricity, gas supply, logistics services, banks and hospitals. Protecting these systems has been outlined by various doctrine and government mandates, e.g. HM Government (2010). Advice from the UK's CPNI and U.S's DOE advise that organisations carefully monitor their networks for signs of intrusion or attack.

Past monitoring proposals have focused on revising enterprise solutions, such as IDS, for ICS/SCADA environments. These approaches are successful at identifying a limited range of attacks. A threat identified by Hadziosmanovic et al. (2013) is semantic attacks, as was the case with Stuxnet. This type of attack might not be immediately recognised, unlike denial of service (DoS) style attacks, when systems are noticably non-functional.

In this paper we identify that the security in ICS can be seen as the malicious and intentional attempt to subvert the safety of the system to cause harm to the operating organisation. We therefore use established runtime-monitoring techniques that are based on the ITL/Tempura framework and adapt them to provide an early warning system that can be deployed in an ICS/SCADA environment. The approach offers the opportunity for real-time detection and response to unusual activities through semantic monitoring of safety and security properties. The contribution of this paper is an approach to monitor safety properties of ICS components based on a formal and verifiable specification of that components behaviour. The proposed solution will be passive and is designed to be non-intrusive to the existing technology in recognition that existing ICS/SCADA systems often accommodate fragile timing constraints.

The rest of this paper is structured as follows: Section 2 introduces ICS/SCADA systems with focus on architecture and programmable logic controllers, Section 3 captures current challenges that monitoring techniques face in ICS/SCADA environments, Section 4 introduces Interval Temporal Logic (ITL) and Tempura and motivates application of ITL and Tempura to ICS/SCADA, Section 5 and 6 detail initial experimental validation and results, Section 7 describes limitations of the approach, Section 8 reviews related work, while Section 9 concludes and we outline our recommendations for further work.

2. ICS/SCADA SYSTEMS

Modern SCADA architectures are composed of three segments: the corporate network segment, SCADA

network segment and field devices segment (Nicholson et al. 2012).

Corporate network segment – mostly operate in the same way as a general IT network found in a business. It performs the same day-to-day business operations such as marketing, e-mail communication, accounting and administrative tasks, hence requiring an Internet connection. Also similar is that these networks may connect to third party cloud-based services such as file or web server providers. This means that the corporate network segment has the same attack surface as a general IT network, such as web application SQL injection, e-mail spear-phishing and exploitation of other vulnerabilities.

SCADA network segment – contains the SCADA system, comprised of servers and workstations that are used by operators to interact with the field devices segment. Human Machine Interfaces (HMI) are used by operators to interpret and control field devices. Operators use a software-based graphical user interface that monitors and modifies values, such as motor rotation speed. Validation in software may prevent operators overriding values that could cause harm to machinery or human life. A Historian system collects and stores an audit trail of operational data in a database. Historians may collect information from thousands of devices and therefore need to be able to process large volumes of data in real-time. Backup control centres and data backup centres may also be present. It is at this point and beyond that similarities between general IT networks and industrial networks cease.

Field devices – comprised of Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs) and Intelligent Electronic Devices (IEDs). IEDs are microprocessor devices, such as sensors, motors, circuit breakers or console lights. These devices are controlled by RTUs and PLCs, primarily with fieldbus protocols. RTUs monitor IEDs and transmit data to PLCs or the SCADA network using protocols such as Modbus and DNP3. PLCs are computers which are able to automate functions primarily using simple ladder logic statements. Sensor data flows from the sensors in the field to the RTUs and PLCs and/or to a data collection point in the SCADA network, where it can be interpreted by an operator using an HMI. The RTUs and PLCs can control parts of the infrastructure directly, such as regulating valves or activating switches through the IEDs, based on either data from the field sensors, or from operator input from the control centre. The resulting control action then flows from the operator or RTU to the IED to make the change to the system. These systems are physically connected by Ethernet, fibre-optic cabling, telephone lines, microwave, satellite or radio.

Figure 1 shows a typical SCADA environment and presents the segments, machines and devices that have been discussed.

3. MONITORING CHALLENGES IN ICS/SCADA SYSTEMS

Monitoring techniques in enterprise networks can be categorised as internal or external. External techniques are outside of the device in question and are most commonly network-based, e.g. intrusion detection systems (IDS). Internal reside inside of the device, e.g. anti-virus (AV). External is inherently limited as it can only see network traffic and has no view of inner processes such as system calls. For example, if a rootkit was installed, this would not be visible to an external technique. External techniques are also disrupted by session encryption, such that only flow/five tuple data is visible. Internal techniques that are installed on devices offer a much better view, but are tangible to an adversary and like AV and firewalls, can be detected, disabled and subverted.

Monitoring techniques can also be categorised as passive or active. Passive are those that introduce no new traffic into the environment, while active introduces new traffic, such as probing other devices on the network. Active techniques introduce additional load in ICS networks that may be undesirable, but are likely to collect more detailed data than passive techniques.

Monitoring techniques face unique challenges in ICS/SCADA environments. External techniques such as intrusion detection/prevention systems are generally based on signature-matching principles, identifying attacks such as buffer overflows in network traffic. Attacks against ICS are highly targeted and so signatures are less likely to exist. Attacks may also affect systems without displaying immediate signs of attack, as was the case with Stuxnet, known as semantic attacks (Hadziosmanovic et al. 2013). Proprietary hardware and software and stringent runtime requirements make internal techniques challenging to deploy.

4. APPLICATION OF ITL IN ICS/SCADA SYSTEMS

Interval Temporal Logic (ITL) is a flexible notation for both propositional and first-order reasoning about periods of time found in descriptions of hardware and software systems (Zhou et al. 2005). Unlike most temporal logics, ITL can handle both sequential

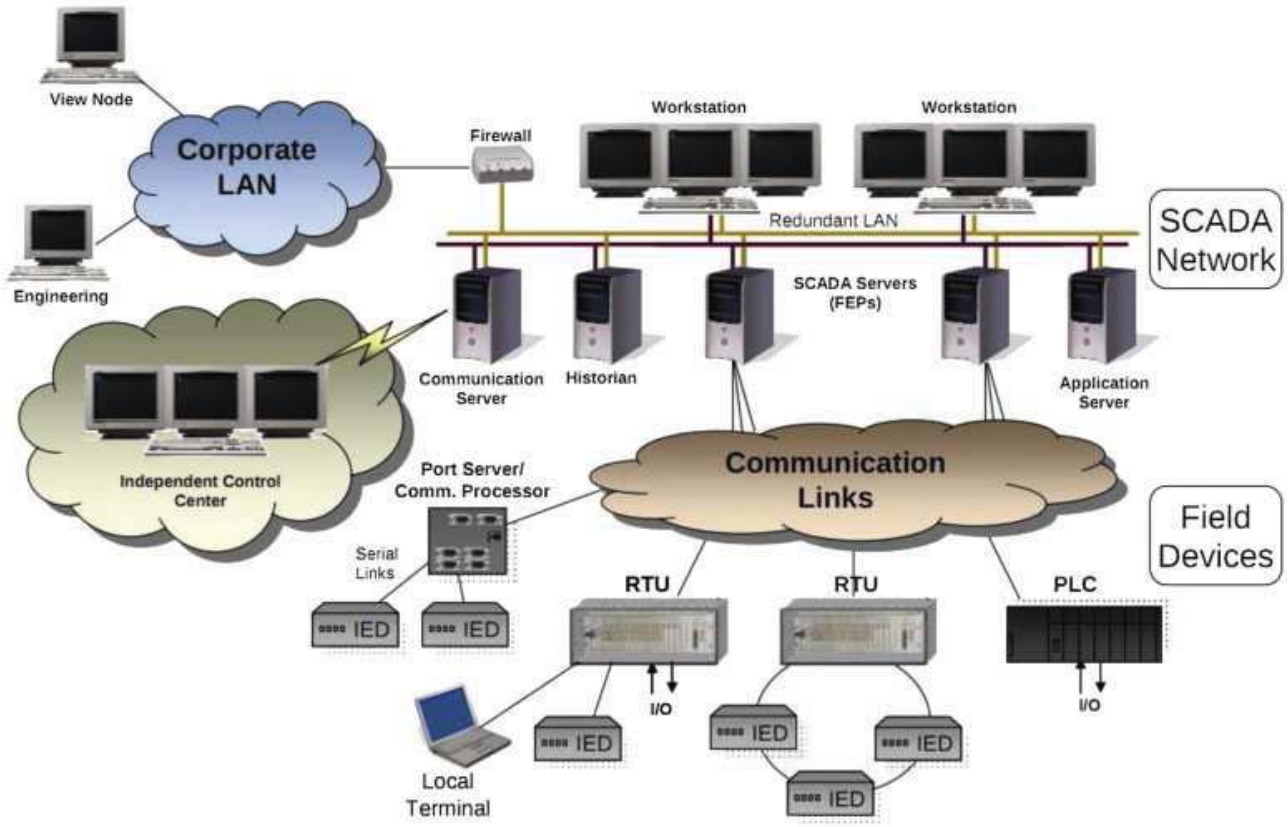

Figure 1: *Typical SCADA System Architecture (Pacific Northwest National Laboratory 2006)*

and parallel composition and offers powerful and extensible specification and proof techniques for reasoning about properties involving safety, liveness and projected time. Timing constraints are expressible and furthermore most imperative programming constructs can be viewed as formulas in a slightly modified version of ITL. Tempura provides an executable framework for developing and experimenting with suitable ITL specifications. In addition, ITL and its mature executable subset Tempura have been extensively used to specify the properties of real-time systems where the primitive circuits can directly be represented by a set of simple temporal formulae. In addition, various researchers have applied Tempura to hardware simulation and other areas where timing is important.

ITL was extended to include notation for programmable logic controllers, in particular ladder logic or function block diagram.

5. EXPERIMENTAL VALIDATION

To validate our approach we used embedded microcontrollers (Arduino) boards with Ethernet compatibility as shown in Figure 2. Our SCADA laboratory contains a number of popular PLCs from Siemens and ABB. We chose the Siemens S7-1200 as it is widely used and supported (shown in Figure 3). Siemens S7-1200 controllers use the proprietary S7 communications protocol. An existing library, Snap7 and Settimino was selected to interface with the PLCs and minimal modification was required.

To test Tempura we launched two exploits against the PLC. First we used an existing publicly available exploit that sets the PLC run mode to OFF. Tempura detects this as the PLC does not enter the next expected state in the duration of time expected. This type of exploit could be trivially detected by an existing monitoring system such as IDS. The second exploit is more advanced; we maliciously upload new program code that is slightly different to the original code. This is detected by Tempura as the state transition does not match the Tempura formula that was automatically generated for the PLC. This semantic attack might not be detected by an existing system as the signature matches normal behaviour and instead breaks operational safety rules.

6. INITIAL RESULTS

The monitoring technique captures a snapshot of the current state of the PLC, as shown in Table 1. Values

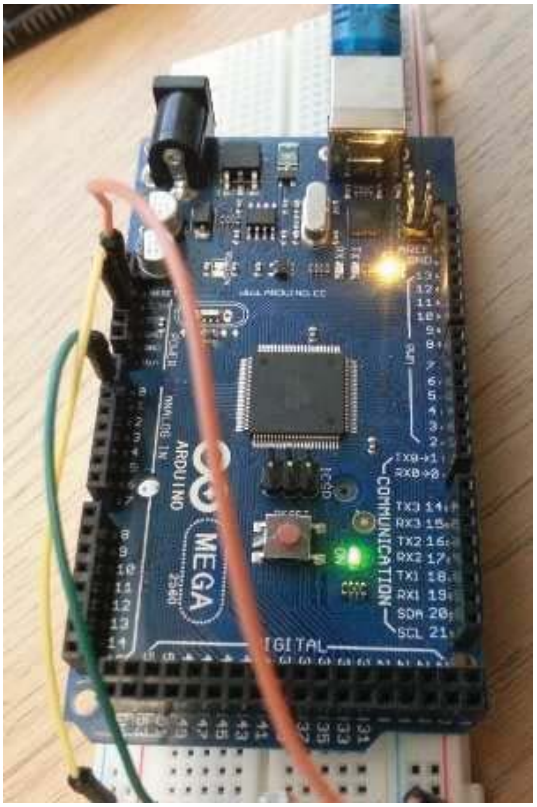

Figure 2: *Arudino microcontroller*

Figure 3: *Siemens S7-1200 PLC*

for Markers, Digital Inputs and Outputs, Counters and Timers are captured. Historic data is stored on an SD card for backup and offline analysis.

The data is sent to Tempura which is running on an x86 Linux workstation. In real time Tempura checks that the output matches the formal description of the PLC, expressed as a Tempura Formula. When Tempura notices that the output does not match the formal description, an alert is raised and an audible warning sound emits from the Arduino microcontroller. In a real deployment this of course could be enhanced by sending a text message to a Factory Manager or alerting the security operations centre (SOC) for the organisation.

7. LIMITATIONS

Our current proposal relies on receiving responses from Siemens PLCs over Profinet, which is based on Ethernet. Typical Ethernet man-in-the-middle attacks are able to sniff traffic, or worst, relay false readings from the PLC. Also at current our approach for reading values increases traffic overhead on the network. This is undesirable in an industrial network as increases in network traffic could have adverse

MK:	00 00 00 00 00 00 00 00 00 00 00 00 00 00
DI:	01 00 00 00 00 00 00 00 00 00 00 00 00 00
DO:	04 00 00 00 00 00 00 00 00 00 00 00 00 00
CT:	04 00 00 00 00 00 00 00 00 00 00 00 00 00
TI:	04 00 00 00 00 00 00 00 00 00 00 00 00 00
MK:	00 00 00 00 00 00 00 00 00 00 00 00 00 00
DI:	01 00 00 00 00 00 00 00 00 00 00 00 00 00
DO:	0F 00 00 00 00 00 00 00 00 00 00 00 00 00
CT:	0F 00 00 00 00 00 00 00 00 00 00 00 00 00
TI:	0F 00 00 00 00 00 00 00 00 00 00 00 00 00

Table 1: *Sensor captures PLC state transitions*

effects on fragile industrial equipment. In addition the scan cycle time of a modern PLC is extremely quick. To ensure that our monitoring technique could see each state we needed to slow the scan cycle time of the PLC so that we could be confident that each state was captured. We expect that this will be resolved with alternative monitoring techniques. In particular, passive observation techniques that do not create additional network traffic. In their approach Hadziosmanovic et al. (2013) identify that PLCs already need to send data to historians and HMIs so this is an ideal candidate for passive monitoring.

8. RELATED WORK

Past proposals applied enterprise solutions to ICS/SCADA environments. For example Yang et al. (2006) and others have created rulesets for IDS that are applicable to ICS protocols. This requires extensive vulnerability assessments of ICS protocols (Igure et al. 2006). This approach is useful but limited when considering semantic attacks (Hadziosmanovic et al. 2013), e.g. an attack may adhere to protocol standards, but may set a gauge to a value that is not allowed by the business/industrial rules, as was the case with Stuxnet. Alternative approaches to IDS have been proposed, such as model-based IDS (Cheung et al. 2007).

McLaughlin and McDaniel (2012) proposed SABOT, a tool that automatically maps PLC logic to a provided specification to derive semantics. Mohan et al. (2012) proposed S3A, an architecture deployed on an FPGA that detects malicious changes to state when execution times differ.

Our work is most closely aligned with Hadziosmanovic et al. (2013) who proposed analysing the internal state of PLCs for safety and security montioring, using existing network traffic, to identify semantic attacks. Our work differs by the choice of formal language; we propose the use of ITL and Tempura and low cost microcontroller deployment.

9. CONCLUSION

Initial experiments have demonstrated that this approach to safety and security monitoring in ICS environments is promising. This approach is by no means a *silver-bullet* solution and should be incorporated as part of a multilayer security architecture. Future work should validate the approach alongside other techniques. In the near future the experimental validation will take place in a simulated red versus blue team ICS exercise. Attack data sets will be recorded so that new techniques can be tested against existing data sets. At current the Tempura processing takes place on a x86 workstation with data that is collected by the microcontroller. We intend to move the ITL library to the microcontroller so that we can deploy test devices en masse, throughout industrial networks. As the devices are low cost, simple to configure, deploy and monitor, they could be a disruptive technology to monitor safety and security properties in ICS environments.

REFERENCES

Cheung, S. (2007) Using model-based intrusion detection for SCADA networks. In: *Proceedings of the SCADA Security Scientific Symposium*, 1–12.

Hadziosmanovic, D. (2013) *Through the eye of the PLC: Towards semantic security monitoring for industrial control systems*. Berkeley, CA, USA: International Computer Science Institute, Tech. Rep.

HM Government (2010, Oct.) *A strong Britain in an age of uncertainty: The national security strategy.* Available from https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/61936/national-security-strategy.pdf

Igure, V. M., Laughter, S. A., and Williams, R. D. (2006) Security issues in SCADA networks. *Comput. Security*, 25 (7). 498–506.

McLaughlin, S. and McDaniel, P. (2012) SABOT: Specification-based payload generation for programmable logic controllers. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, 439–449.

Mohan, S. et al. (2012) S3A: Secure system simplex architecture for enhanced security of cyber-physical systems. *arXiv preprint arXiv:1202.5722*.

Nicholson, A. et al. (2012) SCADA security in the light of cyber-warfare. *Comput. Security*, 31 (4). 418–436.

U. D. o. E. Pacific Northwest National Laboratory. (2006) *The role of authenticated communications for electric power distribution.* Available from http://www.truststc.org/scada/papers/paper34.pdf

Yang, D., Usynin, A., and Hines. J. W. (2006) Anomaly-based intrusion detection for SCADA systems. In: *5th Intl. Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies (NPIC&HMIT 05)*, 12–16.

Zhou, S., Zedan, H., and Cau, A. (2005) Run-time analysis of time-critical systems. *J. Syst. Archit.*, 51 (5/May). 331–345.