Electrical and Electronic Engineering 2012, 2(3): 158-163
DOI: 10.5923/j.eee.20120203.09

Improving Security in SCADA Systems

Samad Araghi[1], Ali Akbar Shams-Baboli[2,*]

[1]Department of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran
[2]Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract Supervisory control and data acquisition (SCADA) systems and distributed control systems (DCS) were innovated to reduce labor costs, and to permit system-wide monitoring and remote control from a headquarter. Control systems are widely implemented in critical infrastructures such as electric grid, natural gas and petroleum, water, and wastewater industries. In this paper the structure and architecture of SCADA systems will be discussed. Since control systems can be vulnerable to different types of cyber attacks which could have destroying results and consequences, we have suggested methods and solutions for increasing security and preventing vulnerability of these systems.

Keywords SCADA, Security, Attack, Vulnerability, Control Systems

1. Introduction

Control systems are computer-based systems that are used in many critical infrastructures and industries such as electric grid, natural gas, petroleum, water, and wastewater industries, to monitor and control sensitive processes and physical functions. Without a safe SCADA system, it is impossible to guard the nation's critical infrastructures. In fact, the recent GAO report[1] shows that designing secure SCADA systems has the highest priority in protecting the nation's critical infrastructures.

Typically, control systems collect sensor measurements, indications and operational data from the field, process and display this information, and relay control commands to local or remote equipments. Control systems may present extra control functions like operating railway switches, circuit breakers and adjusting valves to control flow in pipelines. The most complicated ones control devices and systems at an even higher level.

Control systems have been in place since the 1930s and there are two primary types of control systems. Supervisory Control and Data Acquisition (SCADA) systems and Distributed Control Systems (DCS). SCADA systems typically are used for large, geographically dispersed distribution operations. DCS systems typically are used within a single processing or generating plant or over a small geographic area. For example, a utility company may use a DCS to generate power and a SCADA system to distribute it. We will focus on SCADA systems and our discussions are mostly applicable to DCS systems.

In a typical SCADA system[2], data acquisition and control are performed by remote terminal units (RTU) and field devices which contain functions for signalling and communications. SCADA systems usually use a poll-response model for communications with clear text messages.

The purpose of developing SCADA was to create a control system that will provide good performance and have features that will make it easy to control and could do the tasks easily[3]. Security was not a concern then. Common mistaken belief related to SCADA security was SCADA networks were isolated from all other networks, thus attackers could not access the system[4]. As the industry grows, the necessity for more connectivity also increased. From a small range network, SCADA systems are sometimes joined to other networks to increase the scope. This circumstance lead to new security concerns to these SCADA networks. When the SCADA network is connected to other networks, it is also open to threats, because that connection is open to attackers too. This makes the SCADA system also vulnerable. The use of open standards for SCADA communication protocols are increasing, because it's not as costly as proprietary standards. This reason makes it also easier for attackers and hackers to reach to information in SCADA systems. The open standards make it very easy for attackers to reach complete knowledge about the functioning of these SCADA networks.

The rest of this paper is organized as follows. Section II introduces the architecture of a typical SCADA system. In

Section III, we describe the existent threats against SCADA systems with some examples. Then, in Section IV, we propose some methods which decreases the probability of being vulnerable and insecure for SCADA systems. Finally, in Section V, the major point of this paper and the future

* Corresponding author:
ali.shams2222@gmail.com (Ali Akbar Shams-Baboli)
Published online at http://journal.sapub.org/eee

work is summarized.

2. Architecture

This section describes the common features of the SCADA products in view of their possible application to the control systems of the LHC detectors[5, 6].

2.1. Hardware Architecture

One discerned two main layers in a SCADA system: the "data server layer" which handles most of the process data control activities and the "client layer" which handles the man machine interaction. The data servers communicate with field devices through process controllers. Process controllers, for example PLCs, are connected to the data servers directly or through networks or fieldbuses which may be proprietary (e.g. Siemens H1), or non-proprietary (e.g. Profibus). Data servers are connected to each other and to client stations with an Ethernet LAN. The data servers and client stations are NT platforms but for many products the client stations may also be W95 or higher machines. Fig.1. shows a typical hardware architecture.

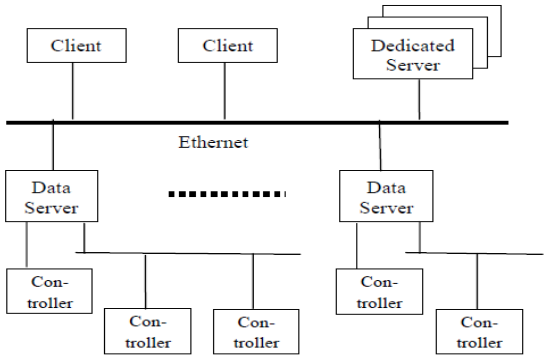

Figure 1. Typical Hardware Architecture

2.2. Software Architecture

The software products are multi-tasking and are based upon a real-time database (RTDB) placed in one or more main and standby servers. Servers are responsible for data acquisition and handling (e.g. polling controllers, alarm managing, calculations, logging and archiving) on a set of parameters, typically those they are connected to.

2.3. Communications

Internal Communication Server-client and server-server communication is in general on a publish-subscribe and event-driven basis and uses a TCP/IP protocol, that means a client application subscribes to a parameter which is 'owned' by a particular server application and only changes to that parameter are then communicated to the client application.

The communications between the control center and remote sites could be classified into following four categories.

● Data acquisition: the control center sends poll (request) messages to remote terminal units (RTU) and the RTUs dumps data to the control center. Especially, this routine includes status scan and measured value scan. The control center regularly sends a status scan request to remote sites to get field devices status (e.g., OPEN or CLOSED or a fast CLOSED-OPEN-CLOSED sequence) and a measured value scan request to get measured values from field devices. The measured values could be in two format: analog values or digitally coded values. These patterns are scaled into engineering format by the front-end processor (FEP) at the control center.

● Firmware download: the control center sends firmware downloads to remote sites. In this situation, the sent message is larger (e.g., larger than 64K bytes) than other cases.

● Control functions: the control center sends control commands to a RTU at remote sites. Control functions are grouped into four subclasses: individual device control (e.g., to turn on/off a remote device), control messages to regulating equipment (e.g., a RAISE/LOWER command to adjust the remote valves), sequential control schemes (a series of correlated individual control commands), and automatic control schemes (e.g., closed control loops).

Broadcast: the control center may broadcast messages to multiple remote terminal units (RTUs). For example, the control center broadcasts an emergent shutdown message or a time-sync message. Gathered data is automatically monitored at the control center to guarantee that measured and calculated values lie within allowed limits. The measured values are monitored with regard to dead-bands and for continuous trend monitoring. They are logged for post-fault analysis. Status indications are monitored at the control center with regard to changes and time tagged by the RTUs. Existing communication links between the control center and remote sites operate at low speeds (usually between 300bps and 9600bps). Fig. 2 indicates a simple SCADA system. In practice, more sophisticated SCADA system configurations exist. Fig. 3, lists three typical SCADA system configurations (see, e.g.,[7]).

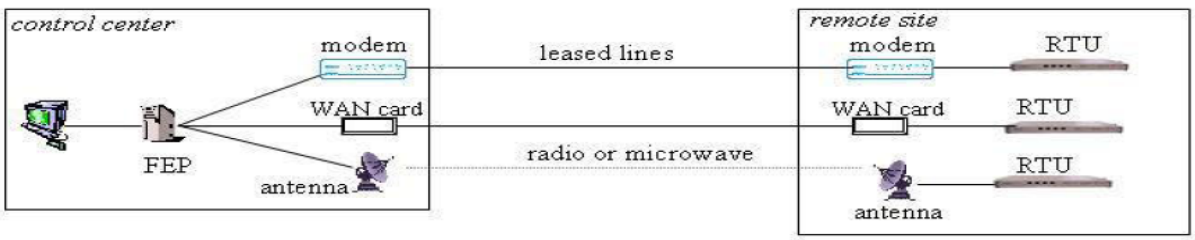

Figure 2. A simple SCADA system

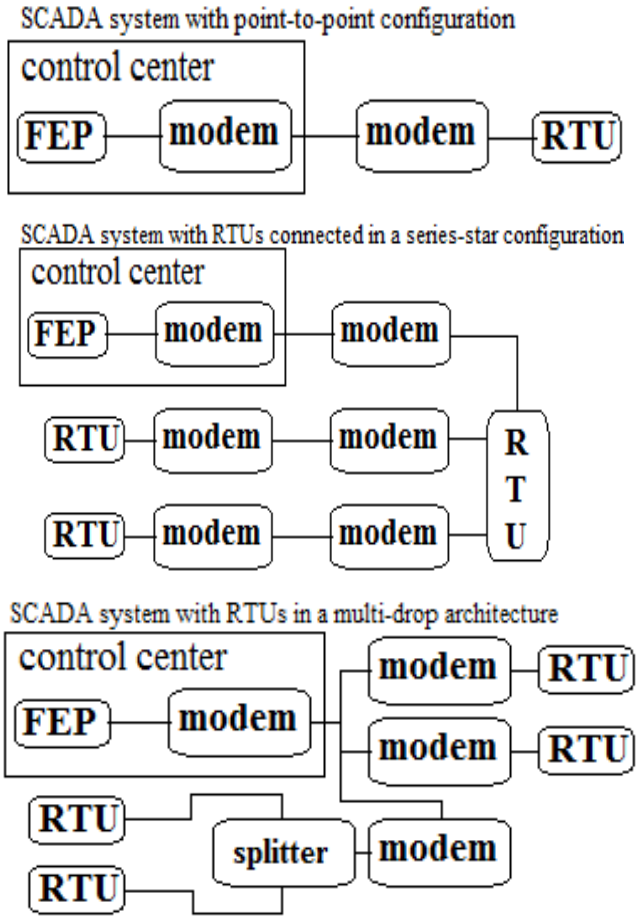

Figure 3. Typical SCADA system configurations

2.4. Remote Terminal Units (RTUs)

● Programmable Logic Controllers

Advances in CPUs and the programming abilities of RTUs made opportunity for more complicated control and monitoring. Applications that had previously been programmed at the central master station can now be programmed at the RTU. These modern RTUs normally use a ladder-logic approach to programming according to its similarity to standard electrical circuits. A RTU that uses this ladder-logic programming is called a Programmable Logic Controller (PLC). PLCs are rapidly becoming the standard in control systems.

● Analog Input and Output Modules

The configuration of sensors and actuators determines the amount and type of inputs and outputs on a RTU or PLC; depending on the model, manufacturer and brand, modules can be designed merely for input, output, digital, analog, or any combination. An analog input module has many interfaces. Typical analog input modules have 8, 16, or 32 inputs. Analog output modules take digital values from CPU and change them to analog representations, which are then sent to the actuators. An output module usually has 8, 16 or 32 outputs, and typically offers 8 or 12 bits of resolution.

● Digital Input and Output Modules

Digital input modules are normally used to indicate status and alarm signals. A specialized digital input module is used for counting pulses of voltage or current, instead of strictly indicating "open" or "closed". This functionality can also be executed using standard input modules and functions found in the ladder-logic programming language of the PLC.

2.5. Protocols

Some of the protocols[8] used in SCADA communication are:

● IEC 60870

IEC 60870 was defined basically for the telecommunications of electrical system and control information and its data structures are geared to that application. It is the most popular standard in the United States of America for electrical power grid SCADA systems, vice versa in Europe.

● DNP3

The second protocol specifically designed for SCADA communications is the Distributed Network Protocol Version 3 (DNP3). DNP3 was created for the electrical industry, but it has been adapted by other industry sectors and is the leading protocol for most SCADA applications in Europe.

● HDLC

Primarily, High Level Data Link Control (HDLC) and Modbus were two of existing other SCADA standards. HDLC, defined by ISO for point-to-point and point-to-multipoint links, is also known as Synchronous Data Link Control (SDLC) and Advanced Data Communication Control Procedure (ADCCP). It is a bit-based protocol, the precursor to Ethernet, and is quickly being replaced by DNP3, TCP/IP and Industrial Ethernet2.

● Modbus

Modbus is a messaging structure developed by Modicon in 1979[9], used to establish master-slave/client-server communication between intelligent devices. Modbus is a comparatively slow protocol that does not define interfaces, thus permitting users to choose between EIA-232, EIA-422, EIA-485 or 20mA current loop. While slow, it is widely accepted and in reality has become standard--a new study indicated that 40% of industrial communication applications use Modbus.

● Profibus

Profibus is a German standard that defines three types: Field Message Specification (FMS) for apply in general data acquisition systems, Decentralized Peripherals (DP) for use when fast communication is needed, and Process Automation (PA) for use when highly reliable and safe communication is required. Foundation Fieldbus is an extension to the 4-20mA standard to take advantage of digital technologies.

● UCA

The Utility Communications Architecture (UCA) is an enterprise from the Electric Power Research Institute (EPRI) designed for the electrical industry. It is more than only a protocol definition; it is a comprehensive set of standards designed to allow "plug and play" integration into systems, letting manufacturers to design off-the-shelf compliant devices. IEEE adopted the UCA standards process in 1999 and

has developed extensions for the water industry. Other industries are also examining UCA for suitability.

3. Attacks against SCADA Systems

SCADA systems were not designed with security in mind[10]; rather the priority of developers has been reliability, availability, and speed. However, this does not mean they cannot be secured. If we can understand a system's features, functions and abilities, we can address its limitations. No inherent security is provided in these systems, since security is not a main concern when the efficiency of the system is under consideration. This situation is acceptable as long as the systems are separated from the outside world. However in recent times, these systems are increasingly being exposed to open access, in order to promote inter-system communication and interaction.

In today's corporate environment, internal networks are used for all corporate communications[9], including SCADA. Therefore, SCADA systems are vulnerable to many of the same threats as TCP/IP-based system. SCADA Administrators and Industrial Systems Analysts are usually deceived into thinking that since their industrial networks are on separate systems from the corporate network, they are safe form outside attacks. PLCs and RTUs are usually polled by other 3rd party vendor-specific networks and protocols like RS-232, RS-485, Modbus, and DNP, and are usually done over phone lines, leased private frame relay circuits, satellite systems, licensed and spread spectrum radios, and other token-ring bus topology systems. This usually gives the SCADA System Administrators a false sense of security since they think that these end devices are protected by these non-corporate network connections.

In an industrial network, security can be compromised in many places within the system and is most easily compromised at the SCADA host or control room level. SCADA computers logging data out to some back-office database storerooms which must be on the same physical network as the back-end database systems, or have a path to reach these database systems. This means that there is a path back to the SCADA systems and finally the end devices via their corporate network. When the corporate network is compromised, then any IP-based device or computer system can be available. These connections are open 24x7 to let full-time logging, which provides an opportunity to attack the SCADA host system with any of the following attacks[9]:

- Use a Denial of Service (DoS) attack to crash the SCADA server forcing to shut down condition (System Downtime and Loss of Operations)
- Erase system files on the SCADA server (System Downtime and Loss of Operations)
- Plant a Trojan and take complete control of system (Gain full control of system and be capable to execute any commands which Operators are authorized)
- Log keystrokes from Operators and acquire usernames and passwords (Preparation for future take down)
- Log any company-sensitive operational data for competition or personal usage (Loss of Corporate Competitive Advantage)
- Change data points or deceive Operators to thinking control process is out of control and have to be shut down (Downtime and Loss of Corporate Data)
- Change any logged data in remote database system (Loss of Corporate Data)
- Use SCADA Server as a launching point to discredit and compromise other system components within corporate network. (IP Spoofing)

Following are listed some of the known attacks, and possible scenarios, which shows the vulnerabilities in SCADA systems.

3.1. Sewage Release in Australia

In March-April 2000, a tedious employee, Vitek Boden, accessed the sewage management system of Maroochy Shire on the Sunshine Coast, Queensland, Australia[11] and released large amounts of drain water into public areas. What Boden did, was to gain access to the system, and manipulate data so that whatever function should have occurred at affected pumping stations did not occur or occurred in a different way. The central server was unable to execute proper control and, at great inconvenience and expense, technicians obliged to be mobilized throughout the system to correct faults at affected pumping stations. It is true that Boden had access to inside knowledge about the system, and access to proprietary software, because of being an ex-employee of the firm, which provided the telemetry equipment to the Maroochy Shire administration. For one, the system did not use suitable wireless protection measures, making it vulnerable at the network level. A strong security policy would have revoked credentials to the designers of the system after it was deployed, too.

3.2. Slammer Worm

In January 2003, a Slammer worm bypassed the corporate network firewall disabling a safety monitoring system for more than four hours and the "Plant Process Computer" for almost six hours at the Ohio Davis-Besse nuclear power plant operated by FirstEnergy Corp. A Davis-Besse contractor who had logged into an unsecured network had distributed the worm into the internal corporate network by one of several unclearly documented backdoor connections to deliver the Slammer worm. The point to be mentioned is that the Slammer worm exploits vulnerability in the MS SQL Server 2000. In fact, this shows that vulnerabilities in the platform or the operating environment are in inherited by the SCADA system.

3.3. Stuxnet

VirusBlokAda warned the first detection of malware that attacks SCADA systems (Siemens' WinCC/PCS7 systems) running on Windows operating systems, in June 2010. The malware is called Stuxnet and uses four zero-day attacks to

install a rootkit which one by one logs in to the SCADA's database and steals design and control files[12,13]. The malware is also able to change the control system and hide those changes. The malware was discovered by an anti-virus security company on 14 systems, the majority of which were located in Iran[13].

4. Securing SCADA

4.1. Securing Remote Connection

Most of cheap attacks could be accomplished on SCADA system communication links between the control center and remote terminal units (RTU) because there is neither authentication nor encryption on these links. Under the umbrella of NIST "Critical Infrastructure Protection Cybersecurity of Industrial Control Systems", "American Gas Association (AGA) SCADA Encryption Committee" has been trying to identify the functions and requirements for authenticating and encrypting SCADA communication links. They propose[7] to build cryptographic modules that could be invisibly implanted into existing SCADA systems (especially, one could attach these cryptographic modules to modems of Fig. 3.) so that all messages between modems are encrypted and authenticated when needed, and they have identified the basic requirements for these cryptographic modules. However, because of the restrictions and constraints of SCADA systems, no viable cryptographic protocols have been identified to satisfy these requirements. Specially, the challenges for building these devices are (see[7]):

- encryption of repetitive messages
- reducing delays because of cryptographic operations
- guarantee integrity with minimal latency _ intra-message integrity: if cryptographic modules buffer message until the message authenticator is verified, it introduces message delays that are unacceptable in most cases _ inter-message integrity: request again messages, replay messages, and destroy specific messages
- accommodating various SCADA poll-response and retry strategies: delays introduced by cryptographic modules may interfere with the SCADA system's error-handling procedures (e.g., time-out errors)
- supporting broadcast messages
- unifying key management
- expense of device and management

4.2. Securing SCADA Network

- Implement Common Criteria evaluations on SCADA control systems: As standards and technology continued to change, the United States of America and Europe began working on standards for a common evaluation criteria for information security. The various evaluation criteria projects begun by the United States of America and Europe combined into a single International Common Criteria project ISO/IEC 15408 with the plan to standardize methods for evaluating information systems security. SCADA control systems products must be included and evaluated based on the Common Criteria standards to guarantee the implementation of SCADA products does not compromise the safety and security of the critical infrastructure.
- Adopt "best practices" and procedures: Most of the vulnerabilities of computer systems are famous and documented. Adopting "best practices" (i.e. implementing secured network equipments and operating systems, and patch management) and procedures (i.e. backups) will allow administrators to protect systems not only from the cyber threats, but also normal system failures.
- Isolate & harden SCADA networks: Isolation of SCADA networks to a closed-loop network with limited and highly restrictive access from physical and electronic outside sources would help in appeasing the threat to them. If the connection of a SCADA network to the Internet or another open network is inevitable, appropriate buffers and checks should be placed between the layers.

Segmented network topologies could increase the level of restrictive access and survivability.

Utilization of authentication mechanisms such as passwords, tokens and biometrics could protect against unauthorized access.

Enabling strong encryption for all data communications would minimize the risk of a security breach.

Vulnerability and threat assessments should be performed regularly on current and newly implemented systems.

Risk estimations and assessments should be conducted on each interconnection between the SCADA and corporate enterprise network.

All unnecessary networks, especially if an open pathway to the Internet is formed, should be disconnected.

Unnecessary services that are not required to support the operation of the SCADA control systems should be disabled or removed.

Firewalls and intrusion detection systems should be implemented, to not only prevent entries but also monitor unintentional security breaches on the SCADA and corporate enterprise network.

Detailed network knowledge should be restricted.

Communicating IP addresses and DNS names is unnecessary and can be costly if in the wrong hands. Implementing single-sign-on procedures will pass authorized users to the command prompt of a device without knowledge of the password or IP address.

Planning an IPSec deployment would increasingly protect the used and unused ports in SCADA networks.

Removing all 'open' ports/backdoors for third party access would reduce the risk appears from the probability of a simple port scan causing in the discovery of vulnerability by an attacker.

It is advised to limit access privileges on an equipment and port level. There is no logical reason for PBX maintenance staff to access a data center database or for IT consultants to access all network devices.

Implementing a virtual private network (VPN) for administrative channel access and partitioning dependent upon privileges provides additional levels of safety.

● Provide Leadership, Accountability and Law Enforcement support: An effective security policy needs the backing and commitment from superior management. Provide for individual accountability via protected system logs or the equivalent. Perform audits, site surveys and penetration tests to ensure the security effectiveness. Increase support for law enforcement to track malevolent access and software, containing support for additional R&D for forensic tools and technologies.

● Establish enterprise security policy through a life-cycle risk management process: Combining the Common Criteria evaluations and an enterprise assurance policy on all SCADA control systems and interconnections of the computer systems could greatly reduce risk by ensuring that the security of one system is not compromised by vulnerabilities of other systems connected to it. Standardizing a process that will minimize the risks associated across the shared network infrastructure and computer systems contains activities to:

Develop a methodology for recognizing important and sensitive infrastructure assets and evaluate security requirements.

Perform vulnerabilities threat assessment.

Develop regular security monitoring and warning process.

Develop and plan for a response and repairing process against possible vulnerabilities and other incidents.

● Establish an Enterprise Assurance Awareness Program: Provide support on wider awareness of the importance and necessity of security, promoting the understanding of security vulnerabilities and corrective measures, and in facilitating greater awareness for the SCADA network. Awareness depends on accessing broad audiences with attractive packaging techniques.

● Develop Continuity Plans: Develop disaster recovery plans to ensure the safety and continued operation of the SCADA network caused by unforeseen and undesirable occurrences or contingencies that interrupt the normal SCADA operations.

5. Conclusions

What concerns countries with background of SCADA systems, will be our basic problems in the future. We have introduced different methods for preventing the vulnerability of SCADA systems. Localization of these methods with the present structure in Iran can be an effective step in preventing these problems. The issues discussed in this paper are predictive. It has been tried to pay careful attention to what is considered to be the goal of a secure and invulnerable SCADA system.

REFERENCES

[1] GAO-04-628T. Critical infrastructure protection: challenges and efforts to secure control systems. Testimony Before the Subcommittee on Technology Information Policy, Intergovernmental Relations and the Census, House Committee on Government Reform. March 30, 2004.

[2] T. Cegrell, *Power System Control Technology*, Prentice-Hall International (UK) Ltd. 1986.

[3] R.J. Robles1, M. Choi1, E. Cho1, S. Kim1, G. Park1 and S. Yeo1, "Vulnerabilities in SCADA and Critical Infrastructure Systems," *International J. of Future Generation and Networking*, vol. 1, no. 1, pp. 102-103.

[4] C. Rolf, Sandia SCADA program – high-security SCADA LDRD final report, 2002.

[5] A. Daneels, W. Salter, "Technology Survey Summary of Study Report", IT-CO/98-08-09, CERN, Geneva 26th Aug 1998.

[6] A. Daneels, W. Salter, "Selecting and Evaluation of Commercial SCADA Systems for the Controls of the CERN LHC Experiments", *International Conf. on Accelerator and Large Experimental Physics Control Systems*, Trieste, Italy, 1999.

[7] AGA Report No. 12. Cryptographic Protection of SCADA Communications: General Recommendations. Draft 2, 5th February, 2004.

[8] M. P. Ward, "An Architectural Framework for Describing Supervisory Control and Data Acquisition (SCADA) Systems", US Naval Postgraduate School, September 2004.

[9] (Technical information bulletin 04-1*), Supervisory Control and Data Acquisition (SCADA) Systems,* NCS TIB 04-1, Communication Technologies, Inc., Chantilly, Virginia, October 2004, pp. 41-42

[10] Stamp, Dillinger, Young, DePoy, "Common Vulnerabilities in Critical Infrastructure Control Systems", Sandia National Laboratories, May 2003.

[11] J. Slay, M. Miller, Lessons learned from the Maroochy water breach, Critical Infrastructure Protection, vol. 253/2007, Springer, Boston, 2007, pp. 73-82

[12] Mills, Elinor, "Details of the first-ever control system malware (FAQ)", CNET. Retrieved 21th July 2010.

[13] "SIMATIC WinCC / SIMATIC PCS 7: Information concerning Malware / Virus / Trojan". Siemens, 21th July 2011.