

Key Factors in Industrial Control System Security

Jonathan P. Chapman, Simon Ofner, Piotr Pauksztelo

Cyber Analysis & Defense Dept.

Fraunhofer FKIE

Zanderstr. 5, 53177 Bonn, Germany

Email: {jonathan.chapman, simon.ofner, piotr.pauksztelo}@fkie.fraunhofer.de

Abstract—Industrial Control Systems (ICS) are a less obvious part of the Internet of Things with their own unique security challenges. We introduce the structure and components of typical ICS installations. We also provide an overview of a number of ICS security approaches. Contrasting these with a sampling of documented ICS security incidents highlights the relevance of this topic. Finally, this approach allows us to identify unsolved problems with respect to securing ICS.

I. INTRODUCTION

The Internet of Things (IoT) isn't just about the stereotypical "smart" thermostats and refrigerators. Your local water supply and gas station may also be part of it. Right next to smartphones and web servers we find a small but critical class of devices and networks: Industrial Control Systems (ICS). They share all the same security issues and considerations that come with this sort of connectivity.

Despite being somewhat low-profile, ICS play many roles and come in various sizes. The term encompasses anything from the smallest devices reporting single sensor readings to huge deployments controlling cross-country pipelines. They may monitor underground gas tanks or open critical valves at just the right time. They may be complex networks of devices, controlling delicate manufacturing processes.

Importantly, ICS include critical infrastructure like water supplies or power generation and distribution. Failures in such systems may result in anything from minor inconvenience to the loss of lives.

Thus, it is particularly worrisome that ICS often suffer from security shortcomings. This is proven by many past incidents, not the least of which was the Stuxnet attack. Some of those issues are rooted in their fundamental designs, often inherited from process automation designs of the 1970s, when the Internet was not yet part of the equation. Other issues lie in software vulnerabilities, increasingly in danger of being exploited as ICS exposure to the Internet grows.

In this paper, we provide a brief overview of ICS and the state of ICS security. We first introduce ICS components and structure in the abstract. We then systemize a number of approaches to securing ICS. To evaluate their effectiveness, we contrast them with real attacks against ICS. Our conclusions show that more research is needed to achieve effective security for ICS.

II. BACKGROUND: CONTROL SYSTEMS

In this section, we provide a brief overview of the structure and main components of industrial control systems (ICS).

A. Terminology

The terminology around ICS is often ambiguous and hence confusing. In particular, there are several terms for larger compounds and installations.

Industrial control system (ICS) is generally used as the overarching umbrella term. A distributed control system (DCS) is limited to a single point of presence, e.g. a specific factory. Supervisory control and data acquisition (SCADA) is used for systems that are spread over a large geographic area, like a power grid or pipeline. [1], [2]

B. Components

ICS generally consist of a mix of both general purpose computers and specialized hardware.

1) *Programmable Logic Controllers (PLCs)*: PLCs are programmable embedded computers. They read values from external sensors and process them in small programs. Their output is provided to other devices or used to control actuators. PLCs are often designed to withstand unfavorable environmental conditions [1], [3].

Separate safety PLCs may be deployed to bounds-check the main process PLCs. They can override the main PLCs to keep or return the overall process to a safe state [2].

2) *Fieldbuses and Devices*: A fieldbus is a local network for communication between field devices. These include PLCs and digital sensors and actors but not general purpose devices.

3) *Human Machine Interfaces (HMIs)*: HMIs visualize the state of an industrial process and allow direct human interaction. They may be implemented as a stand-alone hardware field device or in software running on a workstation [1], [4].

4) *Engineering Workstations*: General purpose computers equipped to program field devices like PLCs and HMIs. Usually, these are Windows PCs residing in the same local network as their respective field devices.

5) *Data Historians*: Data Historians provide long-term memory for sensor readings and process states. This may be as simple as an SQL database. Both integrated solutions, e.g. coupled with HMIs, and stand-alone devices are used [1]. Since historians can provide substantial data on the controlled process, they are valuable targets for attackers [5].

C. Architecture

Figure 1 shows an abstract ICS. A business network is connected to the Internet. A DMZ hosts ICS components accessible from both the control and the business network.

The control network contains field devices and associated components. Finally, field devices communicate with their sensors and actuators via a fieldbus [6], [1].

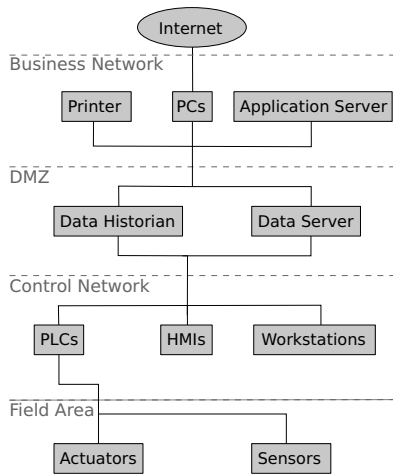


Fig. 1. Example architecture of an ICS with a DMZ

D. ICS Protocols

ICS devices inside and outside the fieldbus interact with special network protocols. Though their origins often date back to the 1970s, many now include TCP/IP variants [7], [3].

Historically, protocols were vendor-specific but vendor-independent protocols like Modbus or DNP3 are increasingly popular. They satisfy the low latency and low computing overhead requirements of ICS, but provide no or minimal security, not least due to their historic evolution [7], [8], [9].

E. Further Considerations

Industrial processes are often sensitive in their execution. Processes including toxic material or machinery generating large forces must be kept within tight operational bounds to avoid damage to machinery or personnel. In chemical processes, the smallest deviation may result in many-fold devaluation of the product [3]. Halting an ICS process may incur loss in revenue, not least due to under-utilization. Machinery may not even be designed to stand still for a prolonged time.

Since any change risks breaking the process, ICS are often kept in their original state for as long as possible. This includes their software, which may thus be vulnerable for ancient exploits. Their fundamental designs are often a mere evolution from the early days of automation, i.e. air-gapped systems with no interconnectivity to other systems, and hence offer little protection against contemporary attacks. In addition, protective mechanisms are often used poorly, e.g. standard passwords are commonly used to avoid downtimes due to forgotten passwords [6], [10].

III. SECURING INDUSTRIAL CONTROL SYSTEMS

Approaches for improving the security of ICS have been proposed. In this section, we provide an overview to a selection

of those approaches, falling roughly into three areas. First, we focus on structural approaches, which propose designs for ICS and related networks that hinder infections and attacks in the first place but also make lateral movement towards or within the ICS more difficult. Detection approaches use signatures or anomaly detection to inform the operators about possible attacks. Finally, several modifications to ICS protocols have been proposed, using cryptographic methods to provide security for their message exchanges.

A. Structural

The work of Dzung et al. [8] focusses on standards introduced by industry and regulatory organizations. These standards often require the separation of networks by firewalls and ensuring that ICS workstations must not be used for regular office tasks. In particular, the ICS should only be reachable through a tightly monitored DMZ.

NIST provides a more exhaustive guide [1] following a similar pattern. While the guide published by the United State's ICS-CERT [11] strongly overlaps with NIST's recommendations, the latter is the only to explicitly recommend air-gapping safety equipment. All of the aforementioned approaches suggest using IDS' and anomaly detection but also point out what policies and procedures are needed additionally.

B. Detection

Generally, field devices, including PLCs, are not powerful enough to run complex software in addition to their process control tasks. Thus, they cannot run host-based intrusion detection systems. However, a few network-based intrusion detection systems have been proposed for ICS. Since these IDS are running on separate hardware, they have no impact on the field devices' performance but should nevertheless be able to process traffic in near real-time to be useful in this setting [12].

The Snort IDS [13] features modules for the non-proprietary ICS protocols DNP3 and Modbus. Hence, defenders can define signatures to black- or whitelist messages from these protocols. Mander et al. propose such rules for several scenarios using the DNP3 protocol [14].

Genge et al. propose the SPEAR approach [15] to reduce the need for manually defining rules. Instead, the user has to describe the network topology and expected traffic flows between the devices in that network. SPEAR uses this information to generate Snort rules to whitelist the expected traffic, i.e. all other traffic will generate alerts.

Implementing ideas sketched by Cardenas et al. [16], Fovino et al. [17] use an approach that does not require defining rules on the protocols directly. Instead, they rely on a simulation of the process which is updated based on the control messages exchanged in the ICS network. The state of the simulated process is then compared against a database of critical states to trigger alerts.

An anomaly detection based approach was proposed by Caselli et al. [18]. They argue that control traffic is more uniform than user generated traffic, which allows them to

derive states where each state refers to an observed ICS protocol message. In a learning phase, state transitions between two messages are added whenever a given former message is observed before the latter. Once complete, the state machine is then used to detect unusual message exchanges.

C. Protocols

As discussed in section II-D, even the modern TCP/IP-based variants of ICS protocols bear a strong heritage of their origins in the 1970s. In particular, they do not implement cryptographic security mechanisms. Since field devices usually do not have the processing power to execute cryptographic protocols used in general computing, Igure et al. propose relying on approaches developed for wireless sensor networks, which face similar challenges [7].

Several groups follow a different avenue, suggesting to integrate security mechanisms into the protocols themselves. Fovino et al. for instance suggest a security layer Modbus that works in a very similar manner to a VPN point-to-point tunnel. They argue that instead of implementing the protocol directly on a field device, it could also be implemented by a gateway and hence used without modifying any field devices [19].

Majdalawieh et al. proposed the DNPsec approach for the DNP3 protocol, which has similar properties like Fovino's approach [20]. Later however, the Secure Authentication feature was introduced for DNP3, providing authenticity and integrity but not confidentiality for DNP3 message exchanges [21]. However, Lee et al. argue that ICS protocols should also provide confidentiality and hence propose a modified version of the DNP3 protocol that provides similar features like Fovino's protocol [22].

IV. EVALUATION

To evaluate the approaches presented in section III, we discuss whether they would have been able to prevent or detect a small sampling of attacks.

A. Attacks

1) *German Nuclear Plant*: In 2016, the loading system of the Gundremmingen nuclear plant was infected with "office malware", later identified as the Conficker and Ramnit worms [23]. The operator claimed no interference with other systems in the plant due to system decoupling [24].

2) *Dragonfly*: In the Dragonfly campaign, attackers used spear fishing emails and injected an exploit kit into benign websites to install malware on the victim's computers. While they only targeted desktop computers, they specifically targeted companies in the energy sector, exfiltrating credentials and data from the compromised machines [25].

3) *Stuxnet*: Stuxnet [26] is still considered the most sophisticated attack against ICS. Discovered in 2010 in an Iranian nuclear facility, it infected Engineering Workstations to inject malicious code into programs running on PLCs. That code would occasionally run cycles modifying the speed of the centrifuges controlled by those PLCs. Eventually, the excessive wear induced by these cycles would result in the destruction of the centrifuges.

B. Evaluation of Defensive Approaches

1) *Structural*: The structural approaches all suggest the use of anti-virus software. This is generally able to detect and prevent further damage from general purpose malware like the one discovered in the *German Nuclear Plant*. Hence, this attack would have been prevented by these approaches.

These approaches do not include recommendations for protecting business networks against sophisticated attackers like those acting in the *Dragonfly* campaign. Hence, they would not have been stopped by the measures proposed by the structural approaches.

A main focus of these approaches is how the business network should be separated from the ICS network. In the facility targeted by Stuxnet, the ICS was air-gapped, i.e. fully physically separated. Nevertheless, it was penetrated by the Stuxnet malware. Since it was relying on exploits undisclosed at that time and use them to gain access to legitimate components of the ICS, there is little reason to assume that any of the suggested measures would have prevented this attack.

2) *Detection*: Intrusion detection approaches for ICS like the ones presented in section III-B focus on the communication with or between field devices. In the malware infection that struck the *German Nuclear Plant*, no unusual communication with field devices occurred, so these approaches would not have been able to detect any conspicuous communications. Similarly, they would have missed the *Dragonfly* attack, which also did not affect the targeted companies' ICS' directly.

In case of the *Stuxnet* attack, the infected workstations did communicate with field devices in the ICS. However, even ignoring the fact that none of the approaches are able to deal with the proprietary protocol used here, we have to conclude that they would not have detected this attack. The Stuxnet malware reprogrammed PLCs using workstations that were otherwise legitimately used for this very purpose. The anomaly that occurred here was on a semantic level and could not have been detected by a syntactic approach.

3) *Protocols*: In section III-C, we discussed five approaches for cryptographically securing ICS protocols. Since these protocols are used for communication with or between field devices, they cannot prevent attacks against workstations like the malware infection in the *German Nuclear Plant*. Neither do they protect against attacks that do not involve field devices or their associated workstations. Hence, they could not have prevented the attacks of the *Dragonfly* campaign.

The main property of the secured protocols is that they prevent the disclosure of information or the delivery of unauthorized commands. In the *Stuxnet* attack, workstations were infiltrated by means outside the scope of field protocols. Once these workstations were penetrated, the malware could act as an authenticated peer and issue legitimate commands. Thus the discussed approaches would not have prevented this attack.

V. RELATED WORK

Igure et al. [7] discuss challenges of SCADA security, pointing out problems of possible counter-measures. They do not contrast those counter-measures with reported security

incidents. Dzung et al. [8] discuss how issues of IT security affect ICS security and apply discussed concepts in case studies, albeit without systemizing them. Galloway and Hancke [3] give an introduction to ICS, including an overview of security concepts. However, they do not present a systemized evaluation of those concepts. Nicholson et al. [27] provide an extensive overview on SCADA security. They highlight the importance of SCADA security with a case study of critical infrastructure but do not relate them to each other. Cheminod et al. [6] discuss ICS security, focusing on differences between ICS and general purpose computing. However, they are mostly concerned with the meta-level of the approaches, e.g. if the performance impact of a method was analyzed.

VI. CONCLUSIONS AND FUTURE WORK

A. Conclusions

In section II, we provided an overview of industrial control systems. Due to their heritage but also due to the requirements of industrial processing, their protocols and devices provide little or no security. However, a number of approaches have been proposed to reduce the risk to ICS. We provided an overview to them in section III and then evaluated them against typical ICS-related attacks in section IV.

Our evaluation shows that with the methods at hand, only marginal security can be achieved. While approaches do consider the business network as a potential attack vector, they do not consider it as a target of an ICS-related attacks by itself. Hence, they provide no guidance with respect to securing them. Also, our analysis uncovers the limits of approaches that only consider communications with and between field devices. For the Stuxnet attack used in our evaluation, the compromised endpoint would use protocol messages that are within the bounds of its legitimate use and could use the respective device's credentials where needed. Thus, we have to conclude that only limited security can be achieved for ICS even with state-of-the-art mechanisms.

B. Future Work

In this work, we could only briefly discuss how defensive approaches for ICS relate to a few observed attacks. Nevertheless, we were able to identify limits with respect to the security achievable for ICS. Future work should however take into consideration more attacks. These attacks should be systemized in a way that allows contrasting them with defensive approaches to obtain a clearer picture of what work is needed to secure ICS. Finally, the systemization should lead to the development of approaches that allow appropriately securing the systems that control our power grids, water supply and other critical infrastructures.

ACKNOWLEDGMENT

The research leading to these results has received funding from the German Federal Office for Information Security (BSI) under grant No. 0623/532 04. The opinions expressed are those of the authors. They do not necessarily represent the view of the German Federal Office for Information Security.

REFERENCES

- [1] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn, *Guide to Industrial Control Systems (ICS) Security*, 2015.
- [2] T. Macaulay and B. Singer, *Cybersecurity for Industrial Control Systems: SCADA, DCS, PLC, HMI, and SIS*, 1st ed. Boston, MA, USA: Auerbach Publications, 2012.
- [3] B. Galloway and G. P. Hancke, "Introduction to industrial control networks," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 2, 2013.
- [4] Bundesamt für Sicherheit in der Informationstechnik (BSI). (2013) ICS-Security-Kompodium [ics security compendium].
- [5] B. Zhu, A. Joseph, and S. Sastry, "A taxonomy of cyber attacks on scada systems," in *2011 iThings/CPSCoM*, 2011.
- [6] M. Cheminod, L. Durante, and A. Valenzano, "Review of security issues in industrial networks," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, 2013.
- [7] V. M. Igiure, S. A. Laughter, and R. D. Williams, "Security issues in scada networks," *Computers & Security*, vol. 25, no. 7, 2006.
- [8] D. Dzung, M. Naedele, T. Von Hoff, and M. Crevatin, "Security for industrial communication systems," *Proceedings of the IEEE*, vol. 93, no. 6, 2005.
- [9] I. N. Fovino, M. Masera, L. Guidi, and G. Carpi, "An experimental platform for assessing scada vulnerabilities and countermeasures in power plants," in *Proc. of the 3rd HSI*, 2010.
- [10] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proc. of the 6th ASIACCS*, 2011.
- [11] Department of Homeland Security. (2009) Recommended practice: Improving industrial control systems cybersecurity with defense-in-depth strategies.
- [12] B. Genge, F. Graur, and P. Haller, "Experimental assessment of network design approaches for protecting industrial control systems," *IJCIP*, vol. 11, 2015.
- [13] "Snort users manual 2.9.8.2," 2016.
- [14] T. Mander, F. Nabhani, L. Wang, and R. Cheung, "Data object based security for dnp3 over tcp/ip for increased utility commercial aspects security," in *2007 IEEE PES GM*, 2007.
- [15] B. Genge, D. A. Rusu, and P. Haller, "A connection pattern-based approach to detect network traffic anomalies in critical infrastructures," in *Proc. of the 7th EuroSec*, 2014.
- [16] A. Cardenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," *HOTSEC'08*, 2008.
- [17] I. N. Fovino, A. Carcano, T. D. L. Murel, A. Trombetta, and M. Masera, "Modbus/dnp3 state-based intrusion detection system," in *24th AINA*, 2010.
- [18] M. Caselli, E. Zambon, and F. Kargl, "Sequence-aware intrusion detection in industrial control systems," in *Proc. of the 1st CPSS*, 2015.
- [19] I. N. Fovino, A. Carcano, M. Masera, and A. Trombetta, "Design and implementation of a secure modbus protocol," in *Proc. of the 3rd IFIP WG 11.10 CIP*, 2009, vol. 311.
- [20] M. Majdalawieh, F. Parisi-Presicce, and D. Wijesekera, *Proc. of the IETA/TeNe/EIAE 2005*. Springer, 2006, ch. DNP3: Distributed Network Protocol Version 3 (DNP3) Security Framework.
- [21] (2011) Further information regarding the release of dnp3 secure authentication version 5 (sav5). [Online]. Available: www.dnp.org/DNP3Downloads/DNP3_SAV5_Further_Info_Announcement_20111201.pdf
- [22] D. Lee, H. Kim, K. Kim, and P. D. Yoo, "Simulated attack on dnp3 protocol in scada system," in *Proc. of the 31st SCIS*, 2014.
- [23] C. Steitz and E. Auchard, "German nuclear plant infected with computer viruses, operator says," 2016. [Online]. Available: www.reuters.com/article/us-nuclearpower-cyber-germany-idUSKCN0XN2OS
- [24] "Betriebsbericht nr. 4/2016 [operational report 4/2016]," 2016. [Online]. Available: www.kkw-gundremmingen.de/betriebsberichte.php?id=45
- [25] Symantec Security Response, "Dragonfly: Cyberespionage attacks against energy suppliers," 2014.
- [26] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security & Privacy Magazine*, vol. 9, no. 3, 2011.
- [27] A. Nicholson, S. Webber, S. Dyer, T. Patel, and H. Janicke, "Scada security in the light of cyber-warfare," *Computers & Security*, vol. 31, no. 4, 2012.