An Experimental Study on the Impact of Network Segmentation to the Resilience of Physical Processes

Béla Genge, Christos Siaterlis

▶ **To cite this version:**

Béla Genge, Christos Siaterlis. An Experimental Study on the Impact of Network Segmentation to the Resilience of Physical Processes. 11th International Networking Conference (NETWORKING), May 2012, Prague, Czech Republic. pp.121-134, 10.1007/978-3-642-30045-5_10 . hal-01531118

HAL Id: hal-01531118

https://inria.hal.science/hal-01531118v1

Submitted on 1 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

An Experimental Study on the Impact of Network Segmentation to the Resilience of Physical Processes

Béla Genge and Christos Siaterlis

Institute for the Protection and Security of the Citizen,
Joint Research Centre, Via E. Fermi, 21027, Ispra, Italy
{bela.genge,christos.siaterlis}@jrc.ec.europa.eu

Abstract. The fact that modern Networked Industrial Control Systems (NICS) depend on Information and Communication Technologies (ICT) is well known. Although many studies have focused on the security of NICS, today we still lack a proper understanding of the impact that network design choices have on the resilience of NICS, e.g., a network architecture using VLAN segmentation. In this paper we investigate the impact of process control network segmentation on the resilience of physical processes. We consider an adversary capable of reprogramming the logic of control hardware in order to disrupt the normal operation of the physical process. Our analysis that is based on the Tennessee-Eastman chemical process proves that network design decisions significantly increase the resilience of the process using as resilience metric the time that the process is able to run after the attack is started, before shutting down. Therefore a resilience-aware network design can provide a tolerance period of several hours that would give operators more time to intervene, e.g., switch OFF devices or disconnect equipment in order to reduce damages.

Keywords: network segmentation, cyber-physical, resilience, security

1 Introduction

Modern Critical Infrastructures (CI), e.g., power plants, water plants and smart grids, rely on Information and Communication Technologies (ICT) for their operation since ICT can lead to cost optimization as well as greater efficiency, flexibility and interoperability between components. In the past CIs were isolated environments and used proprietary hardware and protocols, limiting thus the threats that could affect them. Nowadays CIs, or more specifically Networked Industrial Control Systems (NICS), are exposed to significant cyber-threats; a fact that has been highlighted by many studies on the security of Supervisory Control And Data Acquisition (SCADA) systems [7, 11]. The recently reported Stuxnet worm [8] is the first malware specifically designed to attack NICS. Its ability to reprogram the logic of control hardware in order to alter physical processes demonstrated how powerful such threats can be. Stuxnet was a concrete

proof of a successful cyber-physical attack but by no means a trivial attack. It required a thorough knowledge of the physical system, software and OS vulnerabilities.

The size of physical processes led plant designers to structure SCADA system components into multiple network segments, i.e., Virtual Lans (VLANs), [2] interconnected with wireless devices. One of the main advantages of this approach is that malware infections do not propagate to other VLANs unless the attacker is capable to compromise the protection mechanism, e.g., firewalls, of other VLANs as well. Nevertheless, the compromise of one network segment could cause the physical process to shut down, e.g., physical damage, unless designers take appropriate measures to limit the effects of a single compromised control network segment.

Based on these facts, in this paper we investigate the relationship between control network segmentation and the resilience of physical processes. We consider an adversary with a level of sophistication similar to the case of Stuxnet [8] that is able to take over an entire control network segment, i.e., VLAN. The goal of the attacker is to disrupt the normal operation of the physical process by reprogramming the logic of control hardware, as in the case of Stuxnet. The attack scenario was implemented with our previously developed framework [9] that uses real-time simulation for the physical components and an emulation testbed based on Emulab [17] to recreate the cyber part of NICS, e.g., SCADA servers, corporate network. In the implemented scenario we used the Tennessee-Eastman chemical process [5].

The rest of the paper is structured as follows. After an overview of related work in Section 2, we provide a discussion on the segmentation problem and implemented attack scenarios in Section 3. We continue with the presentation of experimental results in Section 4 and we conclude in Section 5.

2 Related Work

According to Wei and Ji [16], a resilient control system is one that is able to: (i) minimize the incidence of undesirable incidents; (ii) mitigate the undesirable incidents; and (iii) recover to normal operation in a short time. In this context our analysis points out an important factor to increase the resilience of industrial systems: the segmentation of process control networks into VLANs. However, this is only one factor that could be considered. Several others were identified with solutions proposed by other authors as well. This section provides a brief presentation of those approaches that mostly relate to ours.

The work of Cárdenas, *et al.* [3] clearly pointed out that intrusion detection systems combined with a reaction mechanism that closes the system monitoring loop are able to effectively increase the resilience of the system. Their work showed that control loops implemented in control hardware, i.e., Programmable Logic Controllers (PLCs), can be adjusted in order to counteract the effects of Denial of Service attacks. In the field of Smart Grids, the work of Zhu, *et al.* [18] showed that routing is a major concern and proposed a secure routing protocol to

increase the resilience of Smart Grids. The work of Chen, *et al.* [4] addressed the importance of hierarchical control solutions for increasing the resilience of Power Grids. The proposed solution uses well-established control theory to guarantee accuracy and system stability. Finally, we mention the recent work of Ji and Wei [10] that proposed a method to quantify the resilience of NICS in terms of quality of control. The authors also proposed a control algorithm for wireless NICS that is able to keep the process in a normal operating state while it is confronted with attacks such as Radio Frequency jamming and signal blocking.

Compared to the previously mentioned techniques, the proposed segmentation-based approach addresses more sophisticated attacks, similar to Stuxnet, that might involve the reprogramming of PLCs. Such attacks are not addressed by existing approaches. Moreover, even in the case of techniques that add countermeasures to the process control network, such as the work of Cárdenas, *et al.* [3], more sophisticated attacks are not targeted. Such approaches rely on PLCs running legitimate control code with incorporated countermeasures, that could be rewritten by malware. The proposed segmentation methodology could also be combined with techniques that ensure the security of industrial systems [1, 12], leading to a system that is both secure and resilient against cyber threats.

3 Problem Statement and Attack Scenario

The Stuxnet malware was a concrete proof that nowadays attackers are capable not only to infiltrate into the process and control networks, but are also capable to reprogram PLCs. Such attack scenarios have an important impact on the physical process as the code that keeps the process in its operating limits is replaced by malicious code. Therefore, new techniques that also address more sophisticated attacks, i.e., similar to Stuxnet, must be developed. In this section we discuss the applicability of network segmentations to counteract such powerful attacks. We begin with an overview of typical process control architectures and we continue with a discussion on the proposed control network segmentation. Finally, we provide a brief presentation on the implemented adversary model and attack scenario.

3.1 Process Control Architecture Overview

Modern SCADA architectures have two different control layers: (i) the physical layer, which comprises actuators, sensors and hardware devices that physically perform the actions on the system, e.g., open a valve, measure the voltage; and (ii) the cyber layer, which comprises all the information and communications devices and software that acquire data, elaborate low-level process strategies and deliver the commands to the physical layer. The cyber layer typically uses SCADA protocols to control and manage an industrial installation. The entire architecture can be viewed as a "distributed control system" spread among two networks: the control network and the process network. The process network usually hosts the SCADA servers (also known as SCADA masters), human-machine

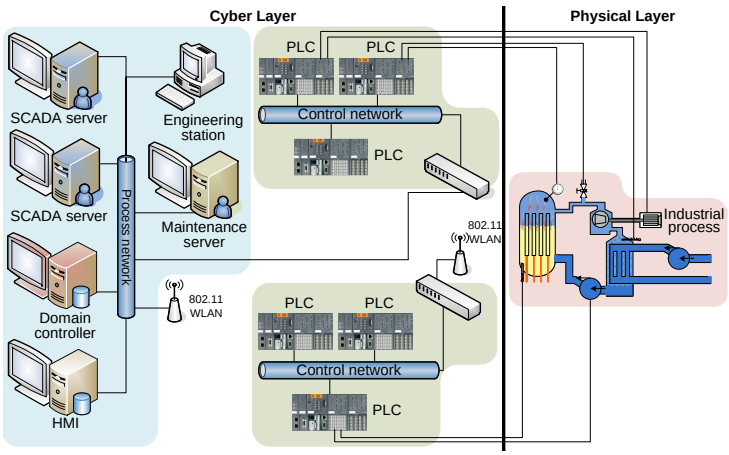


Fig. 1: Process control architecture

interfaces (HMIs), domain controllers and other installation-specific nodes, e.g., engineering stations, maintenance servers. The control network hosts all the devices that on one side control the actuators and sensors of the physical layer and on the other side provide the control interface to the process network. A typical control network is composed of a mesh of PLCs (Programmable Logic Controllers), as shown in Fig. 1.

From an operational point of view, PLCs receive data from the physical layer, elaborate a local actuation strategy, and send commands to the actuators. When requested, PLCs also provide the data received from the physical layer to the SCADA servers (masters) in the process network and eventually execute the commands that they receive. In modern SCADA architectures, communications between a master and PLCs is usually implemented in two ways: (i) through an OPC (Object Linking and Embedding (OLE) for Process Control) layer that helps map the PLC devices; and/or (ii) through a direct memory mapping notation making use of SCADA communication protocols such as Modbus, DNP3 and Profibus.

3.2 Control Network Segmentation

The main goal of the segmentation procedure is to increase the resilience of physical processes. In practice engineers might use network segmentation for a number of reasons such as physical constraints, e.g., location of devices, or protection of mission-critical services. In typical implementations the segmentation is most of the time forced by physical constraints [14] where each individual segment is isolated from the rest and includes network security protection mechanisms, e.g., firewalls. These segments are interconnected by VPNs and are remotely accessible by engineers.

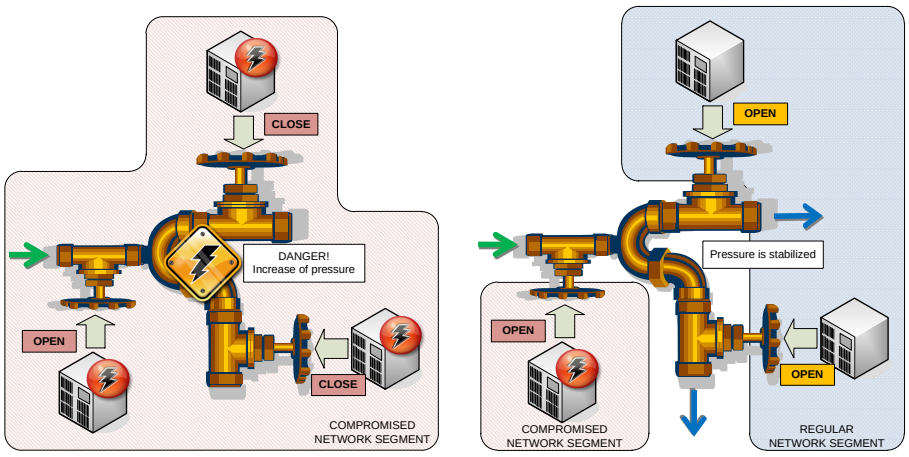
Instead of applying typical segmentation rules such as the ones mentioned previously, in this paper we propose a segmentation that focuses on the physical process. The goal of the procedure is to maximize the resilience of the physical process in case of the full compromise of one or more network segments. The procedure relies on the ability of regular control code to counterbalance the disturbance generated by malicious code running in compromised segments. More specifically, in the proposed approach we separate PLCs controlling input valves (FeedPLCs) from PLCs controlling output valves (FreePLCs), associated to the same unit. This way, the effect of compromised FeedPLCs is balanced by legitimate FreePLCs and vice-versa.

For a better understanding of the impact of the proposed approach, let us assume a simple scenario involving a pipe and 3 valves controlled by 3 PLCs. In this scenario one of the control valves is feeding products into the pipe while the other two are freeing products from the pipe. If designers would place all 3 PLCs on the same network segment (see Fig 2 (a)), in case of an attack that compromises the entire segment the adversary would be able to OPEN the input valve and CLOSE the output valves. This would lead to a sudden increase of the pressure that could cause severe damages to the physical process. On the other hand, by placing FeedPLCs and FreePLCs on separate network segments (see Fig 2 (b)), in case one of the segments is compromised, regular PLCs could balance the generated disturbance and avoid catastrophic consequences.

In the present study we compared the full network compromise setting to the segmentation with the proximity and product flow criteria. Although the analysis is limited to these settings, our main goal was not to be exhaustive, but to show that control network segmentation plays an important role in the resilience of physical processes.

3.3 Adversary Model and Attack Scenario

The employed adversary model reprograms PLCs with malicious code in order to shut down the physical process. Identifying the attack vector that could compromise the system to enable such a scenario is not the main focus of this study. However, the Stuxnet worm together with other studies such as the one performed by Nai Fovino, *et al.* [11] showed that such scenarios are possible in real settings. For instance, corporate firewalls could be compromised by infected user stations within the corporate network. A similar scenario was recently reported by Google [6], the official report stating that errors in Web browser implementations enabled the installation of a malware on a user's machine within the corporate network. From there the malware spread and infected other stations as well. Another example is the Stuxnet worm that included several attack vectors such as USB drives and vulnerabilities in the Operating System, but also vulnerabilities in the Siemens WinCC/Step 7 software. WinCC/Step 7 is the software used to communicate with a variety of PLCs produced by Siemens. By exploiting vulnerabilities in this software, the designers of Stuxnet were able not only to reprogram PLCs but to also hide the changes from human operators.



(a) (b)

Fig. 2: Effect of compromised PLCs on the physical process: (a) proximity-based segmentation, and (b) product flow-based segmentation

As pointed out by Cárdenas, *et al.* [3] attacks that target the minimum/maximum value of parameters/control variables are the ones that can damage the process in relatively short time periods. Such attacks cause the accumulation of products, e.g., steam or water, by completely opening valves that feed products into units and completely closing valves that free products from units. The attack model employed in this study follows the same procedure to force the physical process to shut down. More specifically, based on the documentation of the physical process, the malicious code completely opens input valves and completely closes output valves.

4 Experimental Setting and Results

The results presented in this section prove that network segmentation can be an effective approach to increase the resilience of physical processes confronted with sophisticated attacks. For this purpose we use as a resilience metric the time that the process is able to run after the attack is started, before shutting down, i.e., *shut down time* (SDT). First, the SDT is measured for each compromised VLAN, as generated by the segmentation procedure mentioned in the previous sections. Then, the SDT is compared to the SDT of the full network compromise setting to show the benefits of product flow-based segmentation over proximity/ad-hoc segmentation.

We start the presentation with an overview of the experimentation framework and of the Tennessee-Eastman chemical plant used as the physical process model.

We continue with an overview of the experimental setup and finally we present the experimental results.

4.1 Overview of the Experimentation Framework

In the context of the experimental scenario described in the previous sections we simulated the physical process and we emulated the cyber layer using the experimentation framework developed in our previous work [9]. There are several reasons why we have chosen this approach for our study. First, by testing the resilience of a real system there could be concerns about the potential side effects of the experiment. Second, software based simulation has always been considered an efficient approach to study physical systems, mainly because it can offer low-cost, fast and accurate analysis. Nevertheless, it has limited applicability in the context of cyber security due to the diversity and complexity of computer networks. Software simulators can effectively model normal operations, but fail to capture the way computer systems fail.

The experimentation framework developed in our previous work [9] follows a hybrid approach, where the Emulab-based testbed recreates the control and process network of NICS, including PLCs and SCADA servers, and a software simulation reproduces the physical processes. The architecture, as shown in Fig. 3, clearly distinguishes 3 layers: the cyber layer, the physical layer and a link layer in between. The cyber layer includes regular ICT components used in SCADA systems, while the physical layer provides the simulation of physical devices. The link layer, i.e., cyber-physical layer, provides the "glue" between the two layers through the use of a shared memory region.

The physical layer is recreated through a soft real-time simulator that runs within the SC (Simulation Core) unit and executes a model of the physical process. The cyber layer is recreated by an emulation testbed that uses the Emulab architecture and software [17] to automatically and dynamically map physical components, e.g., servers, switches, to a virtual topology. Besides the process network, the cyber layer also includes the control logic code that in the real world is run by PLCs. The control code can be run sequentially or in parallel to the physical model. In the sequential case, a *tightly coupled* code (TCC) is used, i.e., code that is running in the same memory space with the model, within the SC unit. In the parallel case a *loosely coupled* code (LCC) is used, i.e., code that is running in another address space, possibly on another host, within the R-PLC unit (Remote PLC). The main advantage of TCCs is that these do not miss values generated by the model between executions. On the other hand, LCCs allow running PLC code remotely, to inject (malicious) code without stopping the execution of the model, and to run more complex PLC emulators. The unit that implements global decision algorithms based on the sensor values received from the R-PLC units is also present in the experimentation framework as the *Master* unit. The cyber-physical layer incorporates the PLC memory, seen as a set of registers typical of PLCs, and the communication interfaces that "glue" together the other two layers. Memory registers provide the link to the inputs, e.g., valve position, and outputs, e.g., sensor values, of the physical model.

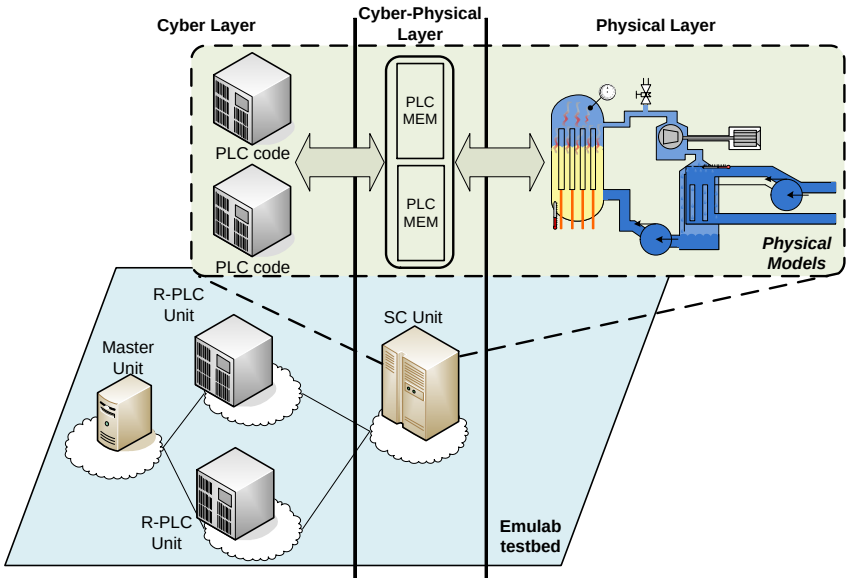


Fig. 3: Experimentation framework overview

Prototypes of SC, R-PLC and Master Units have been developed in C# (Windows) and have been ported and tested on Unix-based systems (FreeBSD, Fedora and Ubuntu) with the help of the *Mono* platform. Matlab Simulink was used as the physical process simulator (physical layer). From Simulink models the corresponding 'C' code is generated using Matlab Real Time Workshop. The communication between SC and R-PLC units is handled by .NET's binary implementation of RPC (called *remoting*) over TCP. For the communication between the R-PLC and Master units, we used the Modbus over TCP protocol.

4.2 Tennessee-Eastman Chemical Process

The TE process is a well-known problem in the automation and process control community mainly because it represents a hypothetical chemical plant that is very similar to an actual plant. The model has been provided by the Tennessee Eastman company [5]. The schematic for the TE process is presented in Fig. 5 where we also show the associated PLCs.

The process is fairly complex: it produces two products from four reactants and the plant has a total of seven operating modes that include a base operating condition. The plant simulation provides a total of 41 measurements and 12 manipulated variables. In this use case we assume that the plant is controlled by the Programmable Logic Controllers (PLCs), i.e., TCCs, that implement the base control strategy proposed by Sozio [15].

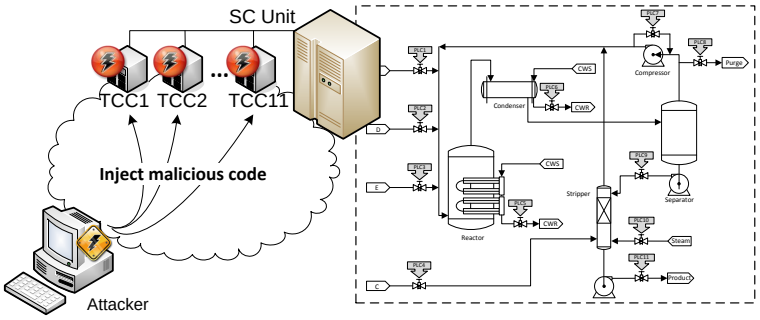


Fig. 4: Experimental setup

4.3 Experiment Setup

The attack scenario described in the previous sections was implemented in the Joint Research Centre's (JRC) Experimental Platform for Internet Contingencies (EPIC) laboratory. The Emulab testbed included nodes with the following configuration: FreeBSD OS 8, AMD Athlon Dual Core CPU at 2.3GHz and 4GB of RAM. In our experiments we used the TE model implementation given as a Matlab 'C'-based MEX S-Function, developed by Ricker [13], from which the stand-alone 'C' code was generated using the Matlab Real Time Workshop. The generated code was integrated into the experimentation framework in order to interact with the real components of the emulation testbed. Regular and malicious control code were implemented as TCCs. The experimental setup is shown in Fig. 4.

The results of the segmentation procedure, based on the segmentation criteria discussed in the previous sections, are given in Fig. 5. For the proximity criteria (see Fig. 5 (a)) - setting A, we defined 3 segments based on the proximity to the 3 main units (*Reactor*, *Separator* and *Stripper*), each implemented as a separate VLAN. By using the same 3 main units we also defined 3 segments based on the product flow criteria - setting B, as shown in Fig. 5 (b). In both figures we used a white color for PLCs on VLAN 1 and VLAN 1', light gray for PLCs on VLAN 2 and VLAN 2' and dark gray for PLCs on VLAN 3 and VLAN 3'.

4.4 Experimental Results

As a result of the previously described segmentation procedure, 6 independent VLANs were identified for both settings, i.e., settings A and B. For each VLAN we implemented the attack scenario described in the previous sub-sections and we measured the shut down time (SDT).

The operation of the TE process for 40h without any disturbances is shown in Fig. 6, where the target setpoints are illustrated with a dashed line. With the implemented control loops the process is able to run in a steady-state, as shown by the two sub-figures depicting the behavior of two parameters that

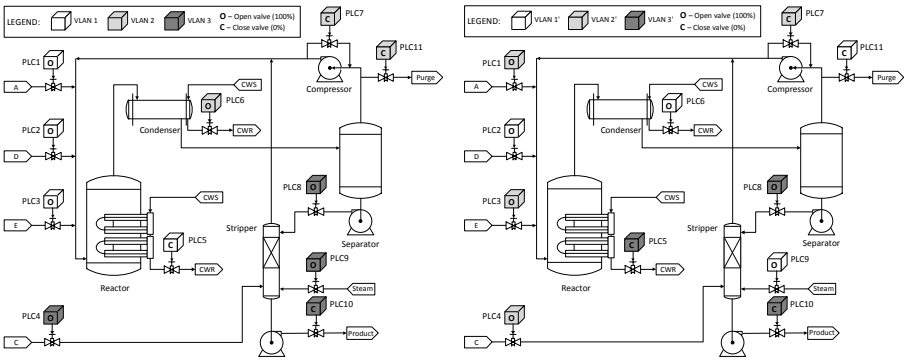


Fig. 5: Tennessee-Eastman process and associated PLCs: (a) control network segmentation in setting A, and (b) control network segmentation in setting B

could trigger a shut down of the process. Without these control loops, process parameters would reach their shut down limits after approximately 3.6h [15].

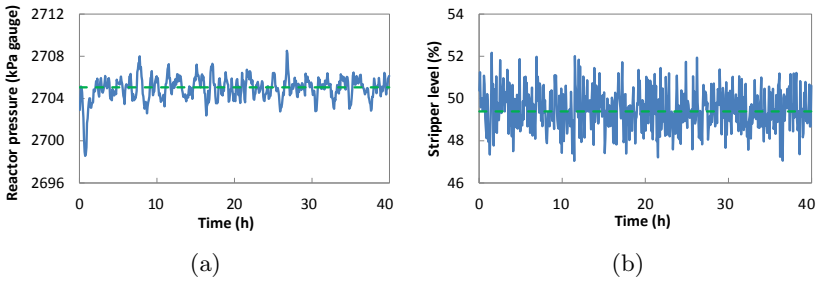


Fig. 6: Normal operation of the Tennessee-Eastman process for 40h without any disturbances: (a) Reactor pressure, and (b) Stripper level

After running the TE process for 10h, in the next step we launched the attack scenario described in the previous sections. First, the attack was launched against the full control network and then against each VLAN identified in the segmentation procedure. Because of space considerations we only illustrate the behavior of the process for the maximum SDT for settings A and B. A summary of the results is given in Fig. 7.

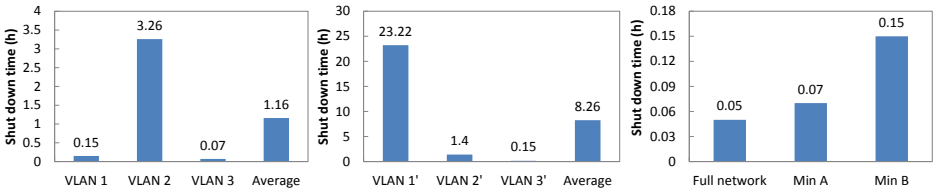


Fig. 7: Shut down time: (a) setting A, (b) setting B, and (c) compared results

Table 1: Reason for shut down of the Tennessee-Eastman process

Setting	**Compromised VLAN**	**Shut down reason**
Full network	–	high reactor pressure
A	1	high reactor pressure
	2	high reactor pressure
	3	high stripper liquid level
B	1'	high reactor pressure
	2'	high reactor pressure
	3'	high stripper liquid level

In the full network compromise setting, all PLCs were running malicious code. This lead to the shut down of the TE process in 0.05h (3min), caused by an increase in the *Reactor* pressure above the 3000kPa shut down limit. The *Reactor* pressure for this setting was illustrated in Fig. 8 (a). In the remaining of this section we use the measured SDT from this setting to show that the SDT can be increased with control network segmentation.

In setting A, the maximum SDT was measured in case VLAN 2 was compromised, while the minimum SDT was measured in case of VLAN 3. As shown in Fig. 8 (b), the attack on VLAN 2 increased the *Reactor* pressure above the shut down limit of 3000kPa in 3.26h. In case of the compromise of the remaining two VLANs we measured a smaller SDT. Thus, for VLAN 1 the measured SDT was 0.15h, while for VLAN 3 it was 0.07h. For VLAN 1, the shut down of the TE process was caused by an excessive increase of the *Reactor* pressure, while for VLAn 3 it was caused by high liquid levels in the *Stripper* unit. We summarized the results for setting A in Fig. 7 (a) and Table 1.

By comparing the results from setting A with the SDT from the full control network setting, we see an increase of 0.02h for the minimum SDT and of 3.21h for the maximal SDT. In the field of Information Security it is a well known fact that the security strength of a system is given by its weakest component. Therefore, in our context it is more important to increase the smallest SDT than

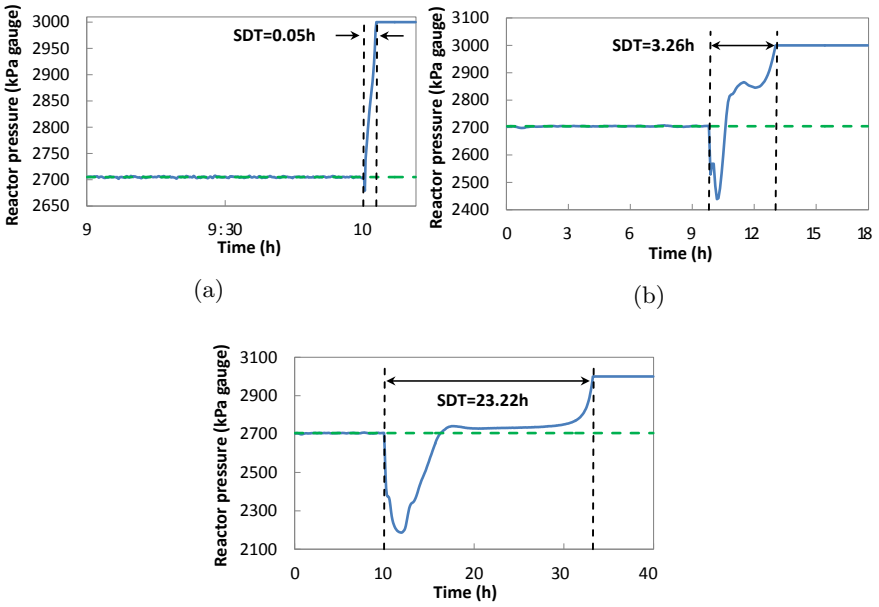


Fig. 8: Disturbed operation of the Tennessee-Eastman process: (a) full control network compromise, (b) compromise of VLAN 2 in setting A, and (c) compromise of VLAN 1' in setting B

the largest or the average value for a specific setting. As in setting A the smallest measured value was of 0.07h (4.2min), this corresponds to an increase of 40% in the value of the minimal SDT. As shown by the results from setting B, the minimal SDT can be further increased by employing process-specific information in the segmentation procedure.

For setting B the segmentation procedure also generated 3 VLANs, but with a different configuration, as shown in Fig. 5 (b). By applying the same experimental strategy for setting B, the maximum SDT was measured for the compromise of VLAN 1', with the effects shown in Fig. 8 (c). In this case the maximum SDT increased to 23.22h, that is more than 7 times the value of the maximum SDT from setting A. The monitored parameter illustrated in Fig. 8 (c) shows that initially the attack causes large deviations on the process parameters. Nevertheless, after 5h legitimate PLCs from non-compromised VLANs bring the process back into the steady-state. The TE process remains in this state for approximately 15h. After this period the accumulated disturbances exceed to capabilities of legitimate PLCs, causing the pressure within the *Reactor* unit to increase until the shut down limit. We also inspected the SDT for the remaining two VLANs. For VLAN 2' the shut down was caused by an excessive increase in the *Reactor* pressure, while for VLAN 3' the shut down was caused by a high liquid level

in the *Stripper* unit. We summarized the results for setting B in Fig. 7 (b) and Table 1.

A significant aspect that we should note for setting B is that the minimum SDT increased to 0.15h (9min), that is more than twice the minimum SDT recorded for setting A. This shows that a careful examination of the physical process can lead to a segmentation that increases the resilience of the physical process by more than 100%, compared to a segmentation based on the proximity criteria. Furthermore, if we compare the increase in the minimum SDT to the full network setting, the increase is above 200%. We summarized these results in Fig. 7 (c).

Based on the results from this section we can conclude that a resilience-aware network design can provide a tolerance period of several additional minutes or even hours. This would give operators more time to intervene, e.g., switch OFF devices, and reduce the damages caused to the physical process.

5 Concluding Remarks

In this paper we have shown that network design choices and specifically network segmentation in VLANs can have an important impact to the resilience of physical processes. Compared to existing approaches, the proposed method has several advantages: (i) it can be applied to a wide variety of industrial systems; (ii) it also targets more sophisticated attacks similar to Stuxnet; and (iii) it does not require new error-prone software/hardware to be installed, for each segment existing security techniques can be replicated. Our proposal can also be viewed as complementary to existing approaches and can be implemented together with other techniques that also address the resilience of industrial systems [3, 4, 18], but do not target more sophisticated attacks. Finally, we also mention that the proposed segmentation methodology can be combined with techniques that ensure the security of industrial systems [1, 12], leading to installations that are both secure and resilient against cyber threats. The study reported in this paper is a first step in our work towards the development of a method that maximizes the resilience of physical processes with network segmentation. As part of our future work, we also intend to study the applicability of our proposal in the context of more complex physical processes such as an entire Power Grid.

References

1. dos Anjos, I., Brito, A., Pires, P.M.: A model for security management of SCADA systems. In: Proceedings of IEEE International Conference on Emerging Technologies and Factory Automation. pp. 448–451 (2008)
2. Boyer, S.: Supervisory Control And Data Acquisition. International Society of Automation–USA (2010)
3. Cárdenas, A., Amin, S., Lin, Z., Huang, Y., Huang, C.Y., Sastry, S.: Attacks against process control systems: Risk assessment, detection, and response. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. pp. 355–366 (2011)

4. Chen, M., Nolan, C., Wang, X., Adhikari, S., Li, F., Qi, H.: Hierarchical utilization control for real-time and resilient power grid. In: Proceedings of the 21st Euromicro Conference on Real-Time Systems. pp. 66–75 (2009)
5. Downs, J., Vogel, E.: A plant-wide industrial process control problem. Computers & Chemical Engineering 17(3), 245–255 (1993)
6. Drummond, D.: A new approach to China (2010), http://googleblog.blogspot.com/2010/01/new-approach-to-china.html
7. East, S., Butts, J., Papa, M., Shenoi, S.: A taxonomy of attacks on the DNP3 protocol. IFIP Advances in Information and Communication Technology 311, 67–81 (2009)
8. Falliere, N., Murchu, L.O., Chien, E.: W32.stuxnet dossier (2010), http://www.wired.com/images_blogs/threatlevel/2010/11/w32_stuxnet_dossier.pdf
9. Genge, B., Fovino, I.N., Siaterlis, C., Masera, M.: Analyzing cyber-physical attacks on networked industrial control systems. In: Critical Infrastructure Protection. pp. 167–183 (2011)
10. Ji, K., Wei, D.: Resilient control for wireless networked control systems. Journal of Control, Automation, and Systems 9(2), 285–293 (2011)
11. Nai Fovino, I., Carcano, A., Masera, M., Trombetta, A.: An experimental investigation of malware attacks on SCADA systems. Journal of Critical Infrastructure Protection 2(4), 139–145 (2009)
12. Pal, O., Saiwan, S., Jain, P., Saquib, Z., Patel, D.: Cryptographic key management for SCADA system: An architectural framework. In: Proceedings of International Conference on Advances in Computing, Control, & Telecommunication Technologies. pp. 169–174 (2009)
13. Ricker, N.: Tennessee Eastman challenge archive (2002), http://depts.washington.edu/control/LARRY/TE/download.html
14. Siemens: Security concept pcs 7 and wincc - basic document (2008), http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=26462131&caller=view
15. Sozio, J.: Intelligent parameter adaptation for chemical processes. Master's thesis, Virginia Polytechnic Institute and State University, USA (1999)
16. Wei, D., Ji, K.: Resilient industrial control system (RICS): Concepts, formulation, metrics, and insights. In: Proceedings of the 3rd International Symposium on Resilient Control Systems. pp. 15–22 (2010)
17. White, B., Lepreau, J., Stoller, L., Ricci, R., Guruprasad, S., Newbold, M., Hibler, M., Barb, C., Joglekar, A.: An integrated experimental environment for distributed systems and networks. In: Proceedings of the 5th Symposium on Operating Systems Design and Implementation. pp. 255–270 (2002)
18. Zhu, Q., Wei, D., Başar, T.: Secure routing in smart grids. In: Workshop on Foundations of Dependable and Secure Cyber-Physical Systems (2011)