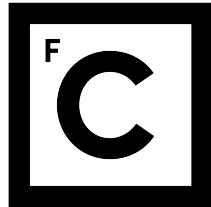


UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA



Ciências
ULisboa

PICSEL: Portable ICS Extensible Lab

Marco Manuel Santos Vieira

MESTRADO EM SEGURANÇA INFORMÁTICA

Trabalho de projeto orientado por:
Prof. Doutor António Casimiro

2020

Acknowledgments

Firstly, I would like to express my sincere gratitude to my supervisors Prof. António Casimiro, Eng. Ricardo Carona and Eng. Gaurav Srivastava for the continuous support on my master thesis, for their patience and motivation.

I am sincerely grateful to all my work colleagues Júlio Morais, Pedro Mendes, Gonçalo Louro, João Lopes and Pedro Barreiros for all their support during the writing of this thesis as well as the motivation they gave me. For the past year we had some fun and memorable moments, thanks.

Finally a huge thanks to my family and especially to my mother and brother for all the support and motivation to move forward.

à minha mãe ...

Resumo

As infraestruturas críticas como a rede elétrica, estações de energia nucleares, refinarias de petróleo e gás, serviço de transportes ou indústrias farmacêuticas, tornaram-se mais e mais importantes para as nossas vidas. Com isto, a necessidade de recorrer cada vez mais a novas tecnologias como a digitalização e Indústria 4.0 nunca foi tão grande. Este avanço tecnológico traz inúmeras vantagens para este tipo de indústrias, como a fácil integração entre diversas centrais através de redes e de novas tecnologias, ou a monitorização e manutenção mais simples e eficiente. Tendo todos estes benefícios, é normal todas as indústrias pretenderem fazer esta transição, especialmente se esta significar uma maior rentabilidade para as mesmas. Ao mesmo tempo, é normal que estas indústrias, cada vez mais tecnológicas, continuem a crescer em dimensão e complexidade, e que subsequentemente exerçam um papel cada vez mais importante para a sociedade. Com o crescimento destes setores e destas empresas, que vão assumindo uma maior importância na sociedade, verifica-se ao mesmo tempo que estas se tornam cada vez mais um alvo, ficando sujeitas a diversos tipos ataques, o que combinado com esta rápida migração tecnológica, pode tornar estas infraestruturas cada vez mais vulneráveis.

Tradicionalmente, a maioria dessas infraestruturas, contêm sistemas de controlo industrial (Industrial Control Systems) (ICS) - sistemas ciberfísicos (cyber-physical system) (CPS) de larga escala - que utilizam um sistema de controlo supervisionado e de aquisição de dados (supervisory control and data acquisition) (SCADA). Estas infraestruturas são constituídas por equipamentos muito diferentes dos utilizados em ambientes IT, especialmente nos seus requisitos. Estes requisitos podem ser: a nível físico, em que o equipamento tem de estar apto para picos de temperatura altos ou para locais extremamente secos ou húmidos; a nível de disponibilidade, uma vez que o processo industrial é a fonte de rendimento e não pode parar; a durabilidade, pois estes equipamentos têm um ciclo de vida extremamente longo, o que faz com que num processo industrial os equipamentos estejam durante longos períodos de tempo sem sofrer nenhum tipo de atualização de sistema operativo ou mudar algum tipo de configuração. Este tempo de vida extremamente longo levanta ainda outros problemas especialmente em segurança, uma vez que, devido aos requisitos de disponibilidade e de durabilidade, fica extremamente difícil de realizar algum tipo de manutenção, como por exemplo uma atualização do software do equipa-

mento, deixando assim os equipamentos vulneráveis durante longos períodos tempo.

Nos últimos anos, pessoas mal-intencionadas têm vindo a perceber a importância e o impacto que estas infraestruturas têm e/ou podem vir a ter na vida das pessoas. Tendo estas infraestruturas tomado grandes dimensões (algumas até a nível global), elas tornaram-se alvos, tanto a nível de hackers individuais como de países, com a possibilidade da próxima grande guerra mundial ser a nível informático. Dois exemplos bem claros do potencial perigo e impacto que estes sistemas têm são o Stuxnet, que foi constituído por um ataque à central de enriquecimento de urânio no Irão, com o objetivo de atrasar esse processo de enriquecimento e forçar o Irão a desistir do programa nuclear, e o BlackEnergy, que foi um ataque à rede elétrica da Ucrânia e que causou um apagão geral no país. Ambos os ataques mostram os potenciais perigos de ataques a ICS e os potenciais impactos no caso de serem bem sucedidos, o que leva a pensar na quantidade de infraestruturas e organizações que podem estar vulneráveis. Combinando isso com a privação de medidas de segurança nos ICS, o resultado está a ser uma grande quantidade de alvos valiosos que estão apenas à espera de serem explorados. Como a maioria desses sistemas são constituídos por equipamentos em que realmente o tempo de vida útil é bastante longo e, na maioria dos casos, têm requisitos de disponibilidade extremamente altos, é importante que, de alguma forma, seja possível reunir informações e executar testes de segurança de modo a proteger estas infraestruturas, sem comprometer o seu normal funcionamento. Normalmente, estas infraestruturas são muito complexas e são constituídas por uma grande diversidade de equipamentos, protocolos de comunicação e diversas tecnologias de transmissão, o que dificulta todo o processo.

Esta tese apresenta uma solução PICSEL, que consiste numa Testbed portátil que foi projetada e desenvolvida de maneira a tentar solucionar alguns desses problemas. O objetivo principal do PICSEL é poder executar testes de segurança red e blue team e avaliar os possíveis impactos e mitigações. Além disso, vários requisitos foram considerados de forma a recriar os inúmeros elementos encontrados em ambientes OT. Por exemplo, um dos requisitos é o tipo de indústria que vai ser recriada. Esse requisito, tendo em conta que em cada tipo de indústria há uma grande variedade de requisitos diferentes, significa que o PICSEL deve ser capaz de suportar diferentes tipos de equipamentos e arquiteturas, e também, a possibilidade de poder ser reconfigurado de forma fácil. A tese descreve a sua arquitetura e discute todas as considerações tomadas nas decisões de design. Estas considerações têm como base a necessidade de criar diferentes cenários e o equipamento disponível. Para testar todos os equipamentos, alguns cenários de teste foram estudados com base em diferentes indústrias e implementados de modo a que estes ambientes recriem o mais fielmente um ambiente industrial real.

Com esses ambientes definidos e implementados no PICSEL, são também apresentados diversos cenários de avaliação que foram definidos de acordo com os objetivos do PICSEL podendo assim concretizar os mesmos. Para testar cada um dos objetivos do PICSEL: exploits disponíveis, soluções de segurança, e novas vulnerabilidades, foi utilizado o PICSEL, onde foram recriados dois cenários, tendo sido possível perceber os seus possíveis impactos nesse ambiente industrial. Para esses objetivos e para cada um desses cenários de teste é possível também executar vários tipos de ataques e perceber os diversos vetores de ataque. Estes ataques consistem em provas de conceito que descrevem possíveis ataques utilizados, bem como exploits disponíveis na Internet. Esta tese, fornece uma forma prática de realizar testes de segurança sem ter de recorrer a um ambiente real, e assim ser possível avaliar em pequena escala o comportamento deste tipo de ataques, e tendo o ambiente configurado, é também possível testar potenciais ferramentas de mitigação para os mesmos.

Palavras-chave: PICSEL, Modular, SCADA, ICS, Security

Abstract

Critical infrastructures such as electric power grids, nuclear plants, oil and gas refineries, transportations systems or pharmaceutical industries, play an increasingly important role in our lives due to technological advancement and the precision industry. Traditionally, most of these infrastructures, also called industrial control systems (ICS), are large-scale cyber-physical systems (CPS) which all use supervisory control and data acquisition (SCADA). Over recent years, malicious actors have realized the importance and impact of these infrastructures. Combining this with the deprivation of security features in ICS resulted in a large quantity of high value targets just waiting to be exploited. Since these systems are based on equipment with a really long lifetime and, in most of the cases, have an extremely high availability requirement, its important to, somehow, gather information and perform security tests in order to protect these infrastructures, without compromising a live operation. Normally these infrastructures are very complex and often have a remarkable diversity of equipment, communication protocols and transmission technologies.

This thesis presents a portable testbed, PICSEL, which was designed and developed to achieve the following goals: to be a portable testbed testing existing exploits and new security solutions whilst exploring new vulnerabilities within the equipment or the environment. Several requirements were considered in the design of the testbed: for instance, choosing the equipment that allowed for more environment configurations; choosing power supplies that support additional equipment; and designing a static electrical diagram based on each device's requirements. With these requirements, the testbed must be able to support different types of equipment and architectures, allowing for applications in multiple industries, inside which it can be easily reconfigured. The thesis describes the testbed architecture and discusses the design decisions, presenting two test scenarios that were studied and implemented using PICSEL. In each of these test scenarios, different attacks were performed validating each of the PICSEL goals. Testing known vulnerabilities, testing exploits in the wild and exporting information from PICSEL equipment to an external tool were very important steps to validate the results. Therefore, this thesis provides proof of concept confirming the key value of a modular and reconfigurable testbed, PICSEL.

Keywords: PICSEL, Modular, SCADA, ICS, Security

Contents

List of Figures	xv
List of Tables	xvii
1 Introduction	1
1.1 Motivation	3
1.2 Goals	4
1.3 Contributions	4
1.4 Timeline	4
1.5 Thesis Outline	5
2 Context and Related Work	7
2.1 IT vs OT	7
2.2 Industrial Control Systems	8
2.3 SCADA System	9
2.4 ICS Components	10
2.4.1 Control Components	10
2.4.2 Network Components	12
2.5 ICS Architecture	13
2.6 ICS Network	15
2.7 ICS Communication Protocols	16
2.7.1 Profinet	16
2.7.2 S7comm	17
2.7.3 Modbus	18
2.7.4 IEC 61850	18
2.7.5 IEC 60870-5-104	20
2.8 Testbed Concept	21
2.9 Information Gathering	21
2.10 Related Work	21
2.10.1 Large-Scale Physical Systems	21
2.10.2 Small-Scale Physical Systems	22

2.10.3	Hybrid Systems	23
2.10.4	Software Simulated Systems	23
2.10.5	Assessment	23
3	PICSEL	25
3.1	Problem Definition	25
3.2	Equipment	27
3.3	Requirements	27
3.4	Design Decisions	29
3.4.1	Test Scenarios	29
3.4.2	Network Architecture	30
3.4.3	System Monitoring	31
3.5	Electrical Scheme	32
4	Implementation	36
4.1	Test Scenarios	36
4.1.1	Industry 4.0	36
4.1.2	Energy Management	42
4.2	Network Configuration	48
4.3	System Monitoring	48
4.3.1	Control Equipment	49
4.3.2	Network Equipment	50
4.3.3	PICSEL Layout for System Monitoring	52
5	Experimental Evaluation	53
5.1	Experimental Objectives	53
5.2	Evaluation Criteria	53
5.3	Experiments	54
5.3.1	Control Command Injection Attack on IEC 60870-5-104	54
5.3.2	MITM Attack on IEC 61850	58
5.3.3	Exploits in the Wild	60
5.3.4	Security Solution	64
5.4	Discussion	64
6	Conclusion	66
	Bibliography	69

List of Figures

1.1	Gantt Chart	5
2.1	PLC Scheme	8
2.2	ICS	8
2.3	SCADA System	9
2.4	SIMATIC S7-1200 PLC	10
2.5	Block Language	11
2.6	Leder Language	11
2.7	Siemens HMI	12
2.8	Input Switch	12
2.9	Purdue levels	14
2.10	Purdue model with NIST Recommended Architecture	15
2.11	The Architecture of Profinet [12]	17
2.12	PASTA in attaché case [20]	23
3.1	Siemens Business Areas	26
3.2	Electrical Installation Part 1	33
3.3	Electrical Installation Part 2	33
3.4	Electrical Installation Part 3	34
3.5	PICSEL equipment	34
4.1	Industry Architecture	37
4.2	TIA Portal interface	38
4.3	TIA Portal project tree	39
4.4	TIA Portal equipment menu	39
4.5	TIA Portal adding new block	40
4.6	TIA Portal network view	40
4.7	S7comm Communication	41
4.8	Download interface	42
4.9	HMI print screen	43
4.10	Network Diagram	43
4.11	Energy topology	44
4.12	Real Environment	45

4.13 WinCC of the Pre-Production Lab	46
4.14 Approach to an Energy Architecture	46
4.15 Network diagram	47
4.16 Primary setup tool	48
4.17 GET_DIAG FB	49
4.18 DB from GET_DIAG FB	50
4.19 Checksum function block	51
4.20 LSyslog_Send function block	51
4.21 Information Gathering	52
5.1 Attacker inside the network	55
5.2 Wireshark showing the network communication	55
5.3 Payload for the "STARTDT act"	56
5.4 Undisturbed sequences of numbered I format APDUs	56
5.5 Exploit	57
5.6 Breaking connection between client and server	57
5.7 New malicious connection	57
5.8 Command Injection	58
5.9 Command injection server side	58
5.10 MITM Attacker	59
5.11 Ettercap console	60
5.12 MITM server side	60
5.13 Client HMI	61
5.14 Result of a enumeration script	61
5.15 ISF Framework	62
5.16 PLC state	63
5.17 Wireshark SegmentSmack Exploit	63
5.18 Security Solution Dashboard	64

List of Tables

2.1	IT vs OT [8]	7
2.2	Modbus Frame	18
2.3	Modbus object types.	18
3.1	Available Equipment	27

Chapter 1

Introduction

Since the early days, a typical industry was formed with workers and machines, and the only protection mechanism they cared about was if the industrial premises were surrounded with a wall and a barbed wire fence. Unthinkably, these primitive concerns are only now starting to change, due to exponential growth of information technologies. Therefore, these critical infrastructures must keep up with new threats, brought along by the impact of this growth.

For the past couple of years, attackers are targeting organizations linked with critical infrastructures such as electric power grids, oil and gas pipelines and refineries, transportation systems, water and sewage plants, amongst others. These types of infrastructures are all large scale cyber physical systems (CPS) which mostly use supervisory control and data acquisition systems (SCADA). The SCADA or Industrial Control Systems (ICS) are turning into the core of critical infrastructures and industries, and this number is growing exponentially due to the large quantity of equipment needed by these industries and data they generate. With this arrives the need to incorporate the pros and cons from the digital world, resulting in modern industry or Industry 4.0 [11]. These new type of systems are due to the advantages of advanced and round-the-clock monitoring, dynamic process changes, the option to track all supply chain and real time reports about incidents that may have occurred. However, this type of critical infrastructures and all their stages, from supplier, manufacturing to distribution, has unleashed an epidemic of largely IP-based digital technologies. Most of them are filled with security vulnerabilities, and with this, attack vectors are growing exponentially making ICS security incidents one of the most imperative topics in the industrial world.

Cyber attacks on critical infrastructure have been escalating in the past few years and, with the development of new tools and techniques, these are also becoming increasingly more complex and disruptive, causing systems to shut down, disrupting operations, or simply enabling attackers to remotely control affected systems. According to Kaspersky ICS 2018 report, the likelihood of a cyber security incident involving ICS is 32% very likely and 56% quite likely and these numbers have been continuously increasing over

the years [18] - this is clearly a reason why companies should worry about and start to take security measures into consideration. However, for better or worse, these concerns are starting to get the deserved attention when the first cybersecurity problems started to appear.

Perhaps, the first and the biggest attack was Stuxnet. This was a perfect example of how powerful these incidents were and the potential impact they may have in our world. Stuxnet was a malware first discovered in 2010 on an Iranian computer. It was precisely designed to disrupt a Siemens industrial control system that in this case was controlling centrifuges in the Iranian nuclear facility of Natanz. This malware has been specifically designed to delay the Iranian Nuclear Program and this was found all over the world spreading for all systems. Despite that, the behavior in normal equipment (e.g. PC, Servers, etc..) wasn't affected. This malware was designed to target specific environment conditions and when those conditions were met, the malware deployed the payload to disrupt all and faking a normal behaviour, impossible for the plant supervisors to detect what was really going on. Another good example appeared on December 23 of 2015, where unknown cyber attackers disrupted the power-grid operations for the first time ever, causing blackouts for over 225,000 customers in Ukraine. Among the most striking features of this attack where the complexity needed for organizing, planning and executing many discrete tasks exhibited by threat actors.

To have a better idea of the complexity of the attack, the timeline of this particular incident started nearly a year prior to the attack, these unknown actors clandestinely established persistent access to multiple industrial networks, identified targets, and ultimately carried out a complex set of actions, which not only disrupted electricity distribution in Ukraine, but also destroyed IT systems, flooded call centers, sowed confusion, and inhibited incident response.

Both attacks, Ukraine power-grid and Iran nuclear program, are a clear example of an Advanced Persistent Threat. Meaning, an enormous quantity of resources, skilled professionals, and monetary aspects were involved. These are perhaps the most important and relevant events that took place, which reflects the poor development in ICS security and the need to secure them.

This takes us to the initial question that was the need of better security mechanisms in the industrial world and new ways to prevent security flaws and proactively try to prevent this type of incidents. These can be achieved by constantly analyzing and testing these systems before and after they are deployed, especially when the device is already in use. Sometimes some vulnerabilities are due to the device environment, (e.g. communication protocol, number of devices, other brands etc..), and sometimes it's hard to simulate these environments. The ideal solution is to recreate all conditions that a specific equipment is going to encounter. For all of those reasons, it involves a large number of possible configurations and occasionally this solution can be easy to recreate, but in the majority

of cases this can be too complex and time consuming. As a result, PICSEL, based on a testbed concept, is a solution to these problems.

A testbed is a platform for simulating a specific environment with the ability to conduct rigorous and transparent tests. In this case, an ICS infrastructure would be recreated in such a way that a security specialist or “pentester” can exhaustively test and try to compromise the system. Another important goal is when an equipment vulnerability is disclosed to a responsible team, and they have a platform with the right conditions where they can try to perform the attack and perform additional tests, and through that evaluate if the system is vulnerable and how could this disturb a real system. If it is a valid threat and somehow a real system could be impacted, the responsible team can issue a report classifying the vulnerability and if possible release a patch that fixes the flaw.

This master thesis project was proposed by Siemens, where since day one it was clear that there was a big interest in the success of this project. Moreover, one of the company statements is innovation. The company is also present in a variety of sectors like Oil and Gas, Utilities and Energy, transportation, amongst others. The presence of the company across these sectors will be beneficial to achieve what this project proposes. In addition, this opportunity allowed me to learn and interact with an entire new world and brought me the possibility to improve security related aspects in ICS world.

1.1 Motivation

Industrial Control Systems (ICS) have distinctive performance and reliability requirements. These infrastructures are structured a little different from a normal organization due to a variety of factors and it is important to keep some points in mind. For instance, the availability of this type of systems is a critical point since they are normally responsible for providing basic needs like power supply, water, controlling nuclear facilities and others. Also, the lifetime cycle of these systems is usually between 10 and 20 years, increasing the maintenance difficulty, a lot of them are built without any security feature. Similarly, the equipment patching routine is typically none, upgrading can be very difficult due to the need of high availability at all times and the extreme difficulty to get authorization to take these systems offline for security related maintenance. Finally, performing vulnerability assessment and security evaluation on a running industrial operation is technically difficult. Auditing such environments without compromising the availability, reliability and performance can be very tricky. Taking all these concerns into consideration, it becomes clear that the best way to perform these type of tests is to first, attempt to recreate an ICS in an isolated environment (testbed), allowing to safely perform the security experiments on that isolated environment. However, constructing a system like this can be very challenging and a tedious process due to the difficulty in obtaining a realistic scale and configurations like a real ICS.

1.2 Goals

The main goal of the work presented in this master thesis is to develop a the PICSEL framework for security testing in ICS. Since the techniques used to attack normal network infrastructures (e.g. corporations) are similar to those that are targeting ICS, they can also be applied to perform security tests with different types of equipment and environments, exploring known vulnerabilities or, if performed correctly, find new ones. Usually these security tests are performed in real environments. However, with ICS there are some differences, as performing these tests on a real scenario could impact the day-to-day operations. Because of that, PICSEL brings an opportunity for security researchers to experiment and demonstrate potential problems that could't be done in a real case scenario without jeopardizing the chain of processes. The objective was to build a testbed using equipment deployed in real industrial situations and able to support communication through a variety of different protocols, for testing a large variety of attack vectors. With that, sophisticated attacks can be conducted in a safe environment. With this we can improve our understanding of how these attacks work allowing the assessment of potential impacts in ICS environments.

1.3 Contributions

The main contribution of this work was PICSEL which was capable of simulating a real ICS environment with multiple equipment and architecture. Using PICSEL, it becomes possible to conduct new vulnerability assessments in communication protocols, firmware versions and hardware. The modular approach adopted in the design of PICSEL provides the possibility of adding different equipment to perform more extensive security assessments.

1.4 Timeline

The initial work plan for this project was divided in six main tasks (Figure 1.1). Therefore, we provide a brief description of what was intended to do in each of these tasks during the development of the project.

- **Understanding ICS world:** firstly integrate and interact with the new team. After that, starting to get in touch with industrial equipment and have a better understanding of the main differences between IT and OT. Understand how HMI, PLC, actuators and other equipment work and what is the main function of each one. Also have an idea of the most used communication protocols and in which scenario they are implemented. At the end understand the different architecture that industrial systems have.

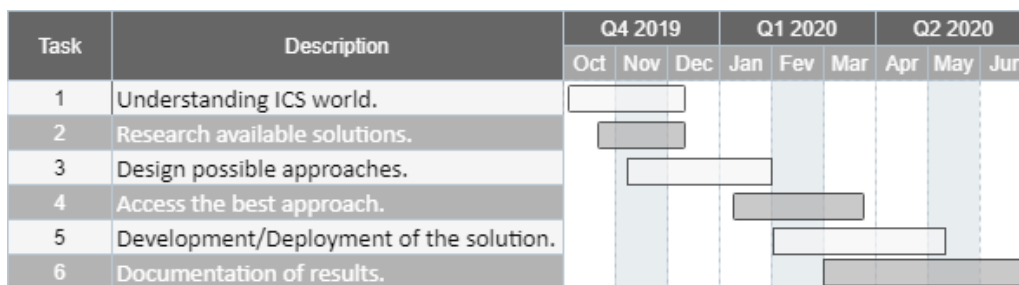


Figure 1.1: Gantt Chart

- **Research available solutions:** research of solutions/approaches available and analyze these implementations to identify the challenges that they encountered and the workarounds. After that, understand what can be useful along with my project goal and try to find a solution for a portable ICS/SCADA testbed.
- **Design possible approaches:** the analysis of hardware available and the compatible communication protocols in order to try to come up with a solution that allows for the conception of this modular testbed allowing for other equipment beyond Siemens's. Meanwhile, get in touch with TIA portal, primary setup tool and other tools needed for the deployment/testing of these type of projects.
- **Assess the best approach:** the research for disclosed exploits, understand them and figure out how they can be deployed in the testbed, also, analyse the typical attack vectors to recreate them in the testbed.
- **Development/Deployment of the solution:** the design of final architecture thus deploying/configuring the same. Also, the developing of the esthetic design of the solutions that makes the portability of this testbed possible.
- **Documentation of results:** is to further analyze how the solution performs and to document the results.

1.5 Thesis Outline

In **Section 2** we will be introducing what an ICS is, alongside the different components and their critical function, as well as some communication protocols. Afterwards, it will be presented why these architectures are different from typical information systems, together with some standards to keep in mind when designing these systems and some security guidelines. Finally, an overview of some related work and a brief assessment of the most interesting works.

Section 3 is where the problems and testbed architecture are carefully explained along with the type of equipment and protocols planned to use. Later, a good explanation of ev-

ery decision made in the development of the system is presented along with the solutions encountered.

Section 4 is dedicated to the implementation and evaluation of the testbed. It includes a detailed explanation about the use cases employed. Also, attack demonstrations are described and performed alongside the implications and effects that these cause to the normal behavior of the testbed.

Section 5 is the discussion with an overview about this project and the results followed by some future work.

Chapter 2

Context and Related Work

In this chapter it will be explained what is Information Technology (IT), Operational technology (OT) and the differences between them. Also, some important ICS aspects are explained like Programmable Logic Controller (PLC), Human Machine Interface (HMI) and communication protocols between devices. Since this type of systems has a slightly different architecture and concepts, it is important to understand and explain some of the general topics. Finally, it will be explained the testbed concept and an overall assessment of related work.

2.1 IT vs OT

Firstly it is important to understand some crucial differences between IT and OT. Initially the main differences were: IT is used to deal with information in a corporate level and OT is responsible for monitoring and/or controlling some physical processes in an industrial setting. In the following Table 2.1 are presented some of the key differences between them [8].

Item	IT	OT
Function	Storage, recovery, transmission, manipulating and protecting data.	Control processes and monitoring.
Access	Connected with the outside world.	Restricted, only people with certain privileges.
Environment	Constantly changing (new devices, new employees).	Very static.
Main priority	Data security.	Availability and information integrity.
Updates	Constant due to software updates. Service interruptions are tolerable and can be done outside working hours.	Updates must be tested carefully in advance and usually requires restarting or stopping the machines.
Life cycle	Shorter life cycles (3-5 years).	Have a longer life cycles (15-20 years).

Table 2.1: IT vs OT [8]

Nowadays, these differences are becoming smaller than ever. The need for controlling

every step of the process from anywhere in the World pushed the industrial infrastructures to a more open connectivity. With this shift comes a need of additional requirements and equipments for these environments and, with that, a higher dependency on these two worlds that raises new security concerns.

2.2 Industrial Control Systems

An ICS is a collection of control components that are interconnected to control and manage a typical industrial environment or more important critical infrastructures. The more common components present in ICS are PLC, HMI, Sensors and actuators. These equipments are always in constant communication with each other. The function of each equipment is going to be explained, starting perhaps with the most important, the PLC, responsible for controlling an industrial process. To make such decisions, they have a Control Processing Unit (CPU) that contains a user program that tells the PLC how to perform based on received information through an input (e.g. sensor). That information, is processed and based on the program that is running it can send instructions to an actuator (e.g. if the sensors say that the reservoir is under a lot of pressure, the PLC can send an instruction to an actuator to open the valve) Figure 2.1 [13].



Figure 2.1: PLC Scheme

In Figure 2.2 is represented the information exchange between devices. Between the PLC and HMI connection, the information flows in both ways because the HMI is receiving information about the process and can also send commands to the PLC.

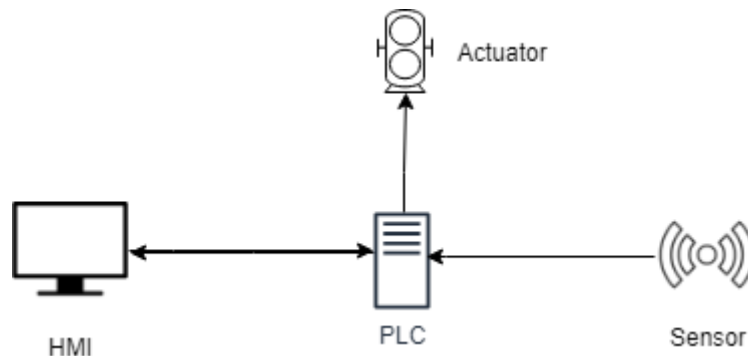


Figure 2.2: ICS

While the communication between sensors and PLC is being exchanged, some worker in higher levels (e.g. control rooms) needs to have information about the process. Typically, to monitor the process state HMI devices are used, allowing the connection between system controllers, monitoring of inputs/outputs and instructions can be sent.

2.3 SCADA System

For monitoring and controlling geographically dispersed assets, typically a SCADA systems is used. These systems are computers interconnected with each other through some specific network communication protocols that, along with some graphical interface, it can provide to an operator centralized monitoring of multiple processes. SCADA systems are planned to collect field information, transfer and display that information to an operator, allowing him to monitor and control the processes from a unique location in real time.

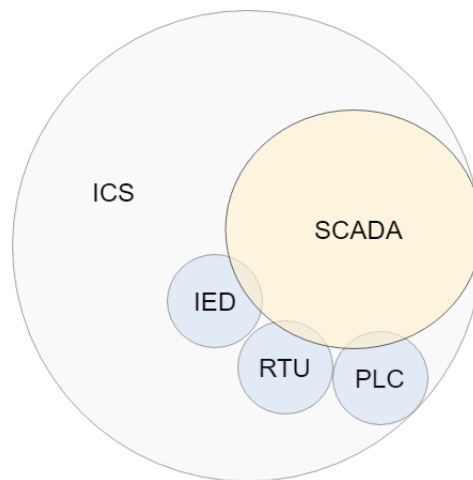


Figure 2.3: SCADA System

Figure 2.3 shows the components and some possible sources of information of a SCADA system. The ICS is the entire industrial process and houses a SCADA system. All industrial equipment is contained inside the SCADA system. Typically, the SCADA system is connected to the control room where equipment such as HMI, engineering workstations and data historian is all connected. The control center collects all information gathered by multiple field sites and is displayed through, for example, multiple HMI devices. The control center is also responsible for centralized alarming, state of the entire chain of processes, and report generation. The field substations perform local control of actuators and monitor sensors in a specific site with PLC or IED (Intelligent Electronic Device). Field sites are often equipped with a remote access capability like a RTU to allow operators to get information, remote access and repairs [19].

2.4 ICS Components

For better understanding, this section shows all typical equipment that an ICS/SCADA system has. To simplify this section, we divided all equipment into two major groups of equipment: Control Components and Network Components [19].

2.4.1 Control Components

Control components refer to all the equipment that directly or indirectly interacts with the process. These equipments usually deal with interactions such as monitoring and controlling the chain of events that are constantly happening in an industrial environment. Some of them are now described:

- **PLC** A higher portion of an ICS includes hardware devices. But, perhaps, the most important part is a computer driven system, which provides a specific set of functions based on the information that is gathered. Remote or distributed devices such as PLCs are operating under the computers command. These commands are programmed (automated) inside the device and based on inputs they are capable to perform a specific set of actions, usually through outputs or actuators [17]. In Figure 2.4 is presented a real PLC.



Figure 2.4: SIMATIC S7-1200 PLC

There are several programming languages are used to operate PLC. The most commonly used are ladder logic, SCL (Structured Control Language) and FBD (Function Block Diagram). Ladder logic Figure 2.6 symbols are very similar to the ones used in electrical circuits and it can be easily learned by professionals with background on electrical circuits. SCL is a high-level text based language that is easy to understand. The FBD Figure 2.5 is a graphical language that can describe a function between input variables and output variables.

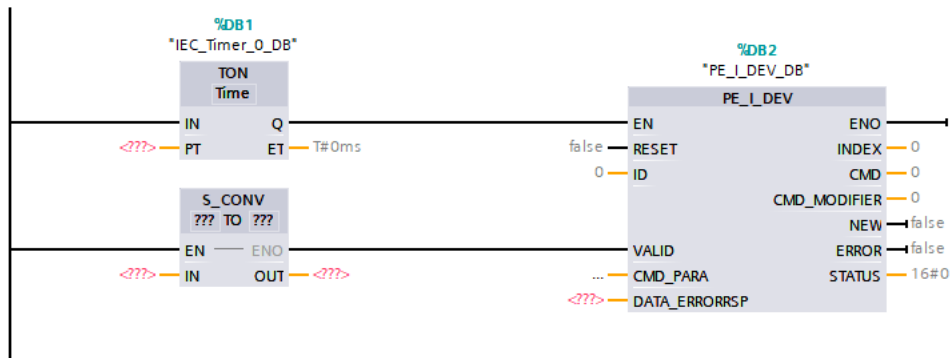


Figure 2.5: Block Language

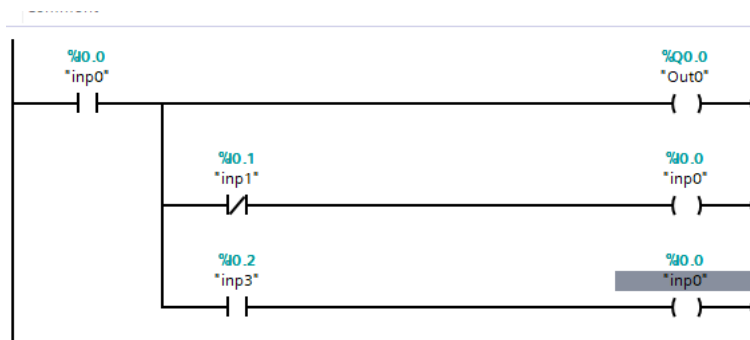


Figure 2.6: Ladder Language

- **Remote Terminal Unit (RTU)** or Remote Telemetry Units are electronic devices controlled by a microprocessor just as a PLC. The main function of these RTU is to interface the SCADA to the objects physically. The interface between objects and SCADA takes place by transmitting to a control station all the telemetry data.
- **Intelligent Electronic Devices (IED)** is a device for advanced power automation. IEDs are used in many industrial processes like control circuit breakers, transformers and capacitor banks.
- **HMI** is where the operator can monitor and supervise the system. It presents all information about the process to the operators. This information can be presented graphically by images, diagrams, alarm and event logging panels. The HMI is a smaller SCADA system responsible for a specific sub-process.
- **I/Os Devices** Input/output (I/O) is the communication between an information processing system, such as a PLC, and the outside world or live operation. Inputs are the signals or data received by the system and outputs are the signals or data sent from it. In Figure 2.8 is represented a simple binary switch connected to the PLC's digital inputs.

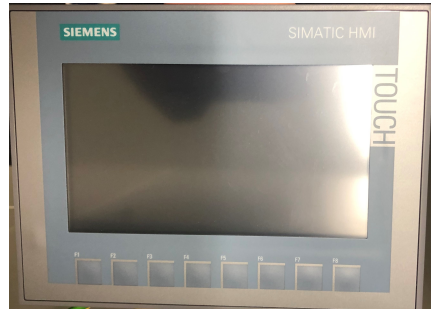


Figure 2.7: Siemens HMI

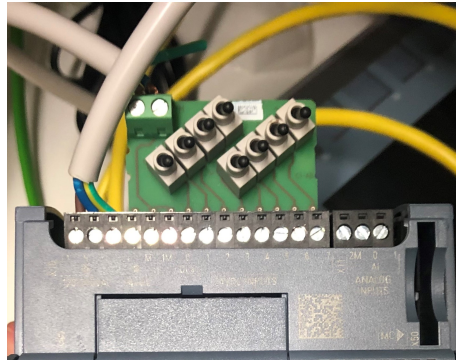


Figure 2.8: Input Switch

2.4.2 Network Components

There are different network characteristics for each layer within a control system hierarchy. Network topologies across different ICS implementations vary with modern systems using Internet-based IT and enterprise integration strategies. Control networks have merged with corporate networks to facilitate and allow control engineers to monitor and control systems from anywhere. This equipment although, has similar functions to the one used in IT world with some different requirements. These requirements are now discussed:

- **Robust design.** Industrial Ethernet devices are exposed to extreme physical conditions and therefore have a hardened enclosure, which is generally made of special plastic and metal. These devices must withstand chemicals, vibrations, or increased electromagnetic loads.
- **Extended temperature range.** Unlike in IT, OT components are not installed in rooms specifically designed for that purpose, instead, they are installed directly on the machinery or control cabinet. To fulfill these environment conditions the components can operate between -40°C and $+85^{\circ}\text{C}$.
- **Increased degree of protection.** These devices are used in harsh environments. As a result, they need to have a increased degree of protection, which starts at IP20

(protection against access by finger) and ranges up to IP69 (full protection against contact, protection against water high-pressure/steam cleaning).

The connection, as already explained, can be between enterprise network and control network. Next, is a list of the major components of an ICS network, regardless of the network topology in use:

- **Firewall.** A firewall protects devices or networks against malicious or unauthorized traffic based on rules or filtering policies.
- **Fieldbus Network.** The fieldbus network links sensors and other devices to a PLC or other controller. This eliminates the need to have one controller for each sensor in case they are dispersed, creating a communication between sensors and controller.
- **Ethernet Hub.** In IE network, it is usually used a star topology and normally they are all connected to an Ethernet hub. This Hub has some characteristics: improvement of the signal quality by restoring the signal amplitude and the signal timing, or by isolating a defective segment and decreasing the collision domain by the principle of frame forwarding.

2.5 ICS Architecture

In this section it will be presented the logical architecture for an ICS network and will be identified some security controls and patterns. Traditionally, the Purdue model is the reference for this type of systems. This model uses a concept of zones to subdivide the Control Network and Corporate Network into logical segments and each segment puts together equipment that performs similar functions or has similar requirements. The Purdue logical framework defines the Control Network at Level 0 to Level 2 and Corporate IT Network from Level 3 to Level 5. These altogether gives a total of six levels Figure 2.9 [13].

Corporate Network:

- **Level 5 - Enterprise:** is where Corporate IT infrastructure systems and applications exist. In this level is gather data from subordinate systems and normally is where we can find things like VPN and corporate Internet access services.
- **Level 4 - Site Business Planning and Logistics Network:** is a lot like Level 5. IT systems like reporting, scheduling, inventory management, capacity services, operational and maintenance management, e-mail, phone, printing services and others can be found here.



Figure 2.9: Purdue levels

- **Level 3 - Site Manufacturing Operations and Control:** is where all the systems that support, control and monitor the plan stand. At this level the operator deals with plant historian, production reporting system, production scheduling system, reliability assurance, engineering workstations. Level 3 also communicates with the above Levels through a DMZ (not directly) and can also communicate with lower Levels.

Control Network:

- **Level 2 - Area Supervisory Control** includes the manufacturing equipment for an individual production area like HMI, Alarms/Alert Systems and Control Room Workstation. Normally actions of Start/Stop on a specific machine or Skid-specific thresholds are performed here.
- **Level 1 - Basic Control:** is where controlling equipment lives, the main function is to receive inputs from the sensors and upon the user program logic. It outputs instructions to an actuator. (Figure 2.2).
- **Level 0 - Process:** represents all equipment that is sending information for all higher levels. This information could be temperature, pressure, motor speed or others. With that information PLCs, Operators and higher levels can control and monitor all supply chain.

We will specially take a closer look at Level 0, 1 and 2 because it is where industrial components are more vulnerable and critical to all supply chain. Accordingly, to an inquire the companies that suffered at least one ICS cybersecurity incident, one of the main consequences with 54% of votes was “Damage to the product/services quality” [18], this means that the majority of incidents involve breaking the production line, meaning that somehow the intruder was able to access the lower levels of ICS architecture and was able to disrupt them.

2.6 ICS Network

According to the National Institute of Standards and Technology (NIST), when designing an ICS infrastructure, it is recommended that both control network and corporate network should be segmented [19]. Traffic such as Internet access, remote access and email should only be allowed in the corporate network.

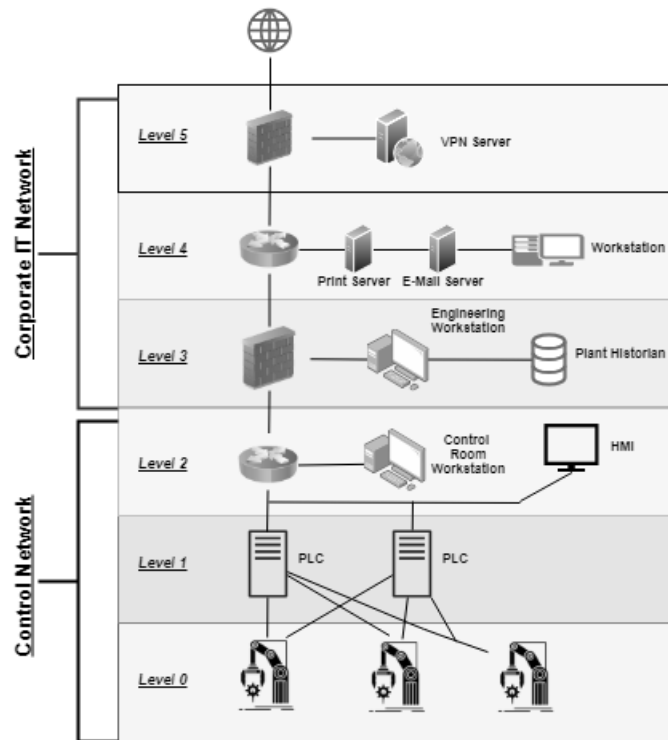


Figure 2.10: Purdue model with NIST Recommended Architecture

This segmentation is basically partitioning the network into smaller sections, minimizing the access to sensitive information by creating layers of defense to protect critical segments from the outside World. The aim is similar to the concept of least privileges, in this concept an user is only able to access resources or information that are necessary to complete the task. Another important question is boundary protection that is “entry level”, usually called demilitarized zones (DMZ). This DMZ is located at the middle of the two domains and the main purpose is to act like a control flow between these domains in order to protect the lower levels of an ICS against malicious adversaries and non-intentional errors that may occur at these layers. These DMZ are important because it is the boundary for the ICS and it is important to add some security features like honey pots, firewalls, network-based malicious code analysis and intrusion detection system (IDS). Based on the decision of these security mechanism, they should allow only data allowed which is to the control network.

2.7 ICS Communication Protocols

Depending upon the ICS environment, different types of communication protocols are used. In this section will be explained some of the most important for this project. Meanwhile, after doing some research, it becomes clear that there is a huge quantity of protocols in the wild. Therefore, to filter all these protocols some meetings were conducted with some experts from multiple vertical markets to assess each one. Some technologies such as fieldbus are being gradually decreasing [1], and others such as Ethernet-based that are assuming a central role in automation environment. Consequently, there are some protocols to be considered and they will be explained in the following sub-sections. As described above, due to ICS being different from IT systems in many aspects, traditional IT protocols cannot be used in ICS. All the systems, interfaces and instruments in an ICS use different protocols for real-time communication and data transfer. These protocols were first designed for serial connection but, with time, they have evolved to support and run on TCP/IP protocols over Ethernet networks.

2.7.1 Profinet

Profinet is a standard based on Profibus that instead of serial interface uses Ethernet. Equipment using Profinet normally are oriented to reliability and real-time communication, along with all TCP/IP functionalities for data transmission. In Figure 2.11 it is possible to see Profinet architecture; this protocol also includes two different methods for addressing MAC-based addressing Layer 2 and IP-based Layer 3 [12].

In Profinet, the protocol is distributed as follows:

- (1) PROFINET IO real-time communication Layer 2-based.
- (2) PROFINET services Layer 2-based (e.g., DCP or LLPD).
- (3) PROFINET services Layer 3-based (e.g., read/write data objects during device parameter assignment).
- (4) Network management and application-related services Layer 7-based, (e.g., SNMP).

Security

As well as with other protocols originally created for communication through Fieldbus, the security features are basic and weak (e.g. FrameID, CycleCounter) [12]. The absence of authentication and lack of security in this protocol is normal and requires other methods

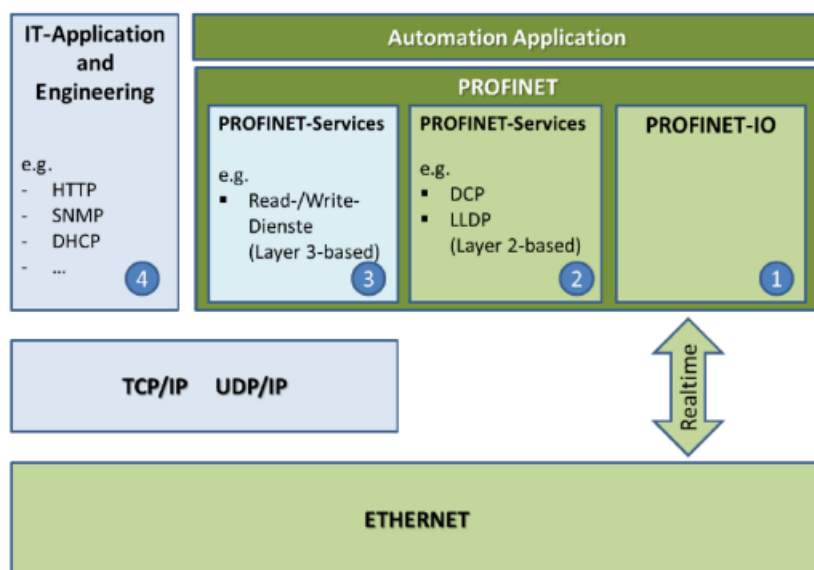


Figure 2.11: The Architecture of Profinet [12]

to try to remediate, one of which is security by obscurity. This method is based on the concept of hiding the system to make it less likely that they will be exploited. Profinet organization recommends precisely that kind of methods, such as restricted security perimeter to avoid any unauthorized or suspicious traffic inside the segmented network [9].

2.7.2 S7comm

S7comm is the backbone of the Siemens communication protocol, its an Ethernet implementation mainly used to connect the PLCs to the PC stations. The S7 protocol TCP/IP implementation relies on the block oriented ISO transport service. This communication protocol is typically found between TIA portal and S7 PLCs or among S7 PLC models [3].

Security

Siemens PLC protocol has 3 versions, S7Comm protocol, early S7CommPlus protocol and new S7CommPlus protocol. S7Comm protocol is used in the communication among S7-200, S7-300 and S7-400 PLCs. This protocol did not involve any anti-replay attack mechanism and can be easily exploited by attackers. The current S7CommPlus protocol implementing encryption has been used in S7-1200 V4.0 and above, as well as S7-1500, to prevent attackers from controlling and damaging the PLC devices. The early S7CommPlus protocol used in the communication among S7-1200 v3.0 is more complicated than S7Comm protocol and uses a two-byte field called session ID for anti-replay attack. However, the session ID is too easy to calculate. The new S7CommPlus protocol used in the communication among S7-1200 v4.0 and S7-1500 has a complex encryption

part to prevent attacks [10].

2.7.3 Modbus

Modbus is another example coming from serial interface to Ethernet. This is one of the oldest industrial protocol and one of the most used in industrial control system. This protocol is located in the application layer, and thus permits different physical means to be used for transport. It offers communication client-server mode with differing sorts of equipment and manufactures on lower layers, which include but are not limited to, the TCP/IP protocol layer [9].

Address	Function	Data	CRC Check
8 BITS	8 BITS	n x BITS	16 BITS

Table 2.2: Modbus Frame

In subsection 2.7.3 is represented a frame from modbus that is pretty simple and straight forward ClientID, Function code, Data, CRC error check. Address is the destination address, Function code is Read/Write and the type of object Table 2.3 and finally a cyclic redundancy check that is an error detecting code that checks if there were any accidental changes to the data.

Object type	Access	Addresses	Size
Coil	Read-write	00001-09999	1 bit
Discrete input	Read-only	10001-19999	1 bit
Input register	Read-only	30001-39999	16 bits
Holding register	Read-write	40001-49999	16 bits

Table 2.3: Modbus object types.

Security

Modbus was designed to be used in highly controlled environments and it does not include any security mechanism on application layer. To perform an attack all that it takes is to form a valid specially crafted packet. This information can be easily obtained or inferred over the Internet using a network sniffer. It is possible to apply generic measures for the TCP/IP like generic IDS solutions that already are specially adapted for Modbus which are highly advisable for enhancing security in this protocol.

2.7.4 IEC 61850

The International Electrotechnical Commission (IEC) defines, IEC 61850 standard has been the most widely industry-accepted standard, which provides a comprehensive data

modeling and abstraction method that unifies data structure definitions across equipment from different manufacturers. IEC 61850 is an international standard defining communication protocols for IED (intelligent electronic devices) at electrical substations.

The IEC 61850 standard is well fit for decentralized and distributed control architectures, because it abstracts the definition of the service and data items to be independent from the underlying protocols. The abstracted data items and services can thus be mapped into any other communication protocol.

Each communication protocol has a specific structure for messages. IEC 61850 maps the data to three different protocols, according to the application. The protocols are:

- **GOOSE** are part of Generic Substation Event (GSE) services that are associated with time-critical activities such as fast and reliable communication between IED used for Protection purposes. In the IEC 61850, one of the messages associated with the GSE services are the GOOSE messages that allow for the broadcast of multicast messages across the Local Area Network (LAN).
- **MMS** is a messaging system for modeling real devices and functions and for exchanging information about the real device, and exchanging process data - under real-time conditions - and supervisory control information between networked devices and/or computer applications.
- **SMV** that is utilized in continuous real-time monitoring and control.

Security

Implementation of security mechanisms in systems often has negative impacts on speed given greater processing requirements. This can prove to be quite complicated in industrial environments where time requirements are very adjusted and specific. For instance, for applications that have time requirements inferior to 4 milliseconds such as GOOSE and SV (Sample Values), encryption is not recommended.

One possible solution to reduce encryption time could be the incorporation of specific chips designed to perform the corresponding mathematical operations. However, this would significantly increase solution costs and would make them unfeasible. On the other hand, it could also be of interest to utilise an encryption that does not require as many calculations, such as one with a symmetric encryption. This would significantly reduce the time required.

Security features in protocols are only a small part of substations protection. Other types of security controls are also recommended:

- Network separation on at least two levels.
- Remote connections with VPNs and TLS.

- Role-based access control using replicated systems and LDAP. Account management should be established from a centralised location that allows for two-factor authentication.
- Patch management.
- Use of firewalls, routers and devices oriented towards perimeter security.

2.7.5 IEC 60870-5-104

The IEC 60870 standard stands for remote control (supervisory control and data acquisition) in electrical engineering and power system automation applications. Part 5 provides a communication profile for sending basic remote messages between a central station and an outstation, which uses permanent directly connected data circuits between the central station and individual outstations. This protocol has two types of messages:

- **APCI format** (Application Protocol Control Information) starts with a start byte with value 0x68 followed by the 8-bit length of APDU (Application Protocol Data Unit) and four 8-bit control fields (CF).
- **ASDU format** (Application Service Data Unit) contains two main sections: the data unit identifier (with the fixed length of six bytes), and the data itself, made up of one or more information objects. The data unit identifier defines the specific type of data, providing addresses to identify the specific identity of the data, and includes additional information as cause of transmission.

Security

This forms a serious vulnerability against IEC 104 communication, especially when transmitted over insecure IP layer. Possible attacks on IEC 104 communication may include:

- Changing the value of an ASDU transmitted in the IEC 104 packet.
- Inserting spoofed ASDU messages into the network.
- Providing DDoS attacks on IEC 104 master or slave stations.
- Inserting a rogue control station into the network.
- Interception of the transmitted data.

2.8 Testbed Concept

Testbeds are test platforms containing specific environments based on the applications that are being replicated. Platforms like these can be used for new concept ideas, new features, optimization improvements, security research, etc., all these with the same goal - to perform a variety of tests and validate the results without the need of a real environment.

2.9 Information Gathering

One of the main goals of PICSEL is to facilitate the testing of new security solutions. These security solutions most often use information retrieved from the environment, and to get this kind of information there are some methods that we could use. To gather this information we can use two different methods, passive and active reconnaissance:

- **Passive** information gathering: the attacker tries to be non-intrusive as possible to stay out of the target's radar. Public domain data is the main source of information in this stage of an attack. The key of a passive reconnaissance is to identify the attack surface without triggering any alert.
- **Active** information gathering involves direct engagement with the target organization through techniques like social engineering or nmap scans. Since it makes direct contact to the target, active information gathering would trigger the target's IDS, IPS if there were any and this is where we draw the line between passive and active information gatherings.

2.10 Related Work

Considering the existing ICS/SCADA testbeds, this section will discuss some of the most interesting and relevant projects in this scope. It becomes clear that testbeds can have a variety of approaches, purposes and designs. Therefore, according to this research, it is clear that these testbeds need to have some categories like those in the survey of SCADA testbed [15]. As a result, it was decided to divide these projects in the following categories: large-scale, small-scale, hybrid and software simulated systems. Finally, after a brief description of all the projects, there is an assessment of the most interesting ones for this research.

2.10.1 Large-Scale Physical Systems

One project was developed by the Idaho National Laboratory National SCADA Testbed [7], they actually implemented a real electrical power grid, with a special concern and focus on electrical substations. They studied the energy industry and had identified emerg-

ing substations automation technologies and configurations. They implemented and determined which substations were vulnerable to cyber-attacks and provided recommendations for mitigation.

2.10.2 Small-Scale Physical Systems

A small scale testbed is the one made by the Indian Institute of technology in Kanpur [2]. This solution is capable of emulating multiple critical infrastructures by supporting different equipment, architecture and configurations. This diversity of equipment like PLC, HMI, sensors and actuators adds the possibility for simulating a good variety of ICS protocols like IEC 61850, MODBUS TCP, DNP3 and others. The architecture follows the layers as in Purdue model with the network segmented, recreating an environment very close to a real process. To enhance this project, they were able to uncover a significant privilege escalation vulnerability, achieving one of the testbed purposes and bringing more possibilities to discover new vulnerabilities and improve security research for industrial systems.

Another one, not so complex, was built in the University of New Orleans [5] - a small scale testbed where they implemented three different scenarios (Power Distribution, Gas Pipeline and Wastewater Treatment). To implement these scenarios they used PLCs, sensors, actuators and for communication they implemented three common protocols EtherNET, Modbus and PROFINET. The PLCs were connected to a software simulated HMI. Finally, they managed to successfully put together a small operation which recreates an industrial environment convenient for an academical purpose like PLC programming, forensics and security research.

An additional work that is not ICS related but the concept idea is very interesting, is the example of PASTA from TOYOTA [20]. Essentially, they created a portable testbed for automotive security because, like in a lot of protocols in ICS world, the CAN protocol was developed with no concerns about cybersecurity. Modern cars have multiple electronic control units (ECU) communicating with each other by CAN protocol, therefore, combining that with delay and lack in development of cybersecurity technologies in the automotive industry, it becomes a real problem. An attacker can simply interfere with these messages and take control over the car.

From this philosophy they came up with a portable solution that recreates a real car, allowing for researches to test different approaches without the need of a real car. Since it has a portable format, this testbed can be easily transported or sent to a researcher to test or develop a proof-of-concept.



Figure 2.12: PASTA in attaché case [20]

2.10.3 Hybrid Systems

Other interesting publication is SCADASim. It is a simulated testbed that is also capable for real world hardware through a concept of gates [16]. A gate is an object that links to an external environment like equipment (e.g. PLC, RTU) and simulation environment. This simulation environment is done with OMNET++, which is usually used to build and simulate network environments. However, they are capable of integrate a variety of protocols either in TCP or Serial. This solution is very interesting because it supports a simulated environment with real equipment, allowing to add real world devices communicating through simulated networks.

2.10.4 Software Simulated Systems

Another idea is from University of Alabama in Huntsville [6] - they propose a framework model that replicates complex SCADA systems entirely on a virtual simulation, making this project very cost-efficient and portable. They achieve this by dividing a SCADA system strategically and logically like Purdue model, virtualizing each part with virtual components through open-source tools (e.g. OpenPLC, Simulink). This approach proved efficiency and portability.

2.10.5 Assessment

After analyzing some works already done in this field, it is interesting to see solutions either being completely software simulated or real complex ICS. The National testbed in Idaho is a good example of a complex ICS testbed where they have real equipment in a real scale scenario. It is a very interesting project, although this project objectives are a bit different (a portable testbed with real communications and devices representing an ICS process). Another good example is the one made by the Indian Institute of technology, in which they had a good diversity of equipment and communication protocols. Never-

theless, their portability was not so good. Although, it was well developed and perfectly filled their purpose and they, while performing some tests to the testbed, were able to discover security issues and vulnerabilities. For other solutions like software simulated, they are especially interesting because they can accomplish some results without any kind of physical equipment, being this one of the most challenging obstacles recreating such scenarios.

Culminating all these ideas, the most interesting project is the one from Indian Institute and PASTA. With such ideas, it is possible to achieve a large portion of this project purpose leading to very interesting results, because in one side we have a complete small ICS testbed and, on the other, we have the portability concept, with the fact that both projects aim for security-related topics. With these two projects combined comes the possibility to transport a testbed to wherever we want along with an entire ICS structure that together is indisputably important for proof of concepts and security researchers.

One of the advantages is that the project is being developed at Siemens. Therefore, it was possible to get access to real equipment and the know-how about how these systems are implemented nowadays. It is relevant to know and understand which equipment and protocols are being deployed so that is possible to identify the most common Siemens devices in the ICS world and communication protocols that are in use. Based on these two factors the next step was trying to find an architecture that enables the use of multiple communication protocols and equipment, offering the security community a portable ICS testbed that recreates a multiple infrastructures for security research.

Chapter 3

PICSEL

In this Chapter it will be explained all the challenges inherent to the development of this project. However, to accomplish the goals of PICSEL there are some concerns and questions that need to be addressed before recreating a real industrial system, specially with a portable format. Meaning, there are questions to be defined in order to achieve a system that represents a real life industrial system. Also, some key functionalities are presented along with an explanation of why they are so relevant for this project.

3.1 Problem Definition

One of the challenges of deploying a portable testbed like PICSEL is to prepare an environment that can be adapted/ready for the changes that different industries can bring.

There are some requirements that are crucial for this project, specially when the recreation of certain environments depends on them. They are key to distinguish the different test scenarios and, through that, enabling different attack vectors between them.

Analysing all relevant industries will help determine the different characteristics that each one has and also what features PICSEL must have in order to be able to handle scenario changes and changes within a scenario in an easy way.

To begin with, it is necessary to select the scenario in which the portable testbed will be used. Then, it is necessary to analyse the network architecture in that industry (or scenario) to identify its characteristics. If these characteristics present some similarities with other scenarios for which layouts already exist, then these layouts may be reused. Otherwise, they must be implemented.

It is also necessary to analyse the security mechanisms that are usually implemented on that specific industry or if there is some common pattern between industries. Once again, this may allow the reuse or the deployment of standard mechanisms. Finally, knowing the specific services and type of equipment along with protocols and which device communicates with whom.

In order to address the scenario that the portable testbed will recreate there are some

of things to keep in mind. This task, that seemed an easy one and turned out to be a very complex one, specially if we want scenarios that are in PICSEL scope. This means that they should meet a real life situation and at the same time can be implemented with a portable format.

There are multiple ways to select which environment we want to see represented. One of them, and the most obvious, is by industry (e.g Energy, Health, Transportation) since they have plenty of unique characteristics that distinguish them. Siemens is present in a variety of business areas (Figure 3.1), with years of experience with clients and specific requirements.

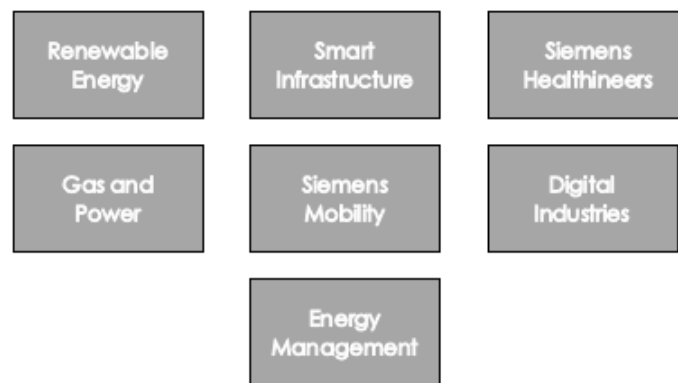


Figure 3.1: Siemens Business Areas

Another approach would be to try to design a functional environment, based on the most used protocols, that fits our goals.

Each industry has a different type of architecture, communication protocols, equipment, concepts and security features. Meanwhile, when an industry is chosen we have to take all of this into consideration since, in the end, the goal is to have the closest recreation as possible. Although, since this project is being developed on Siemens, it makes sense to analyse industries that are in a common scope with the Siemens business areas.

On the other hand, Siemens is a huge company and since the development of this project is time limited, there is no time to deeply analyse all business areas. Doing that would be time and resource consuming. Therefore, we must choose the most important ones and then try to recreate an identical scenario.

Secondly, there are different types of network architectures. Typically, a flat network is the most common, because it is the cheapest, the easiest to maintain and until now the security topic wasn't a big concern. Having the vast majority of security recommendations being about network segmentation and considering that different industries require different architectures makes it clear that this is an important characteristic in order to maintain similar attack vectors in the network along with the security configurations of each scenario.

Finally, there are services that are used on the system that are also important to im-

plement. These type of services along with the functionalities they provide, can bring different types of vulnerabilities.

3.2 Equipment

The perks of developing this project at Siemens include the possibility to work with real equipment even though there are hundreds of industrial products, it was decided to come up with a list of equipment that is commonly used in multiple industries.

Table 3.1 presents the list with the different equipment divided by the categories already explained on Section 2.4. The power category represents the power supplies for this project.

ID	Name	Category
1	SIMATIC S7-1200	PLC
2	SIMATIC S7-1500	PLC
3	Power Supply 1500	Power
4	LOGO Power Supply	Power
5	LOGO Controller	Controller
6	LOGO HMI	HMI
7	KTP700 Basic Panel	HMI
8	SCALANCE S615 LAN router	Network
9	SCALANCE SC642-2C	Network
10	SCALANCE X005	Network
11	SCALANCE XB208	Network

Table 3.1: Available Equipment

In the beginning of this project all the equipment was packet in their original boxes. This is important to refer because, previously I didn't have any experience on working with these equipment or any idea of how they work or which functionalities they had. These difficulties, forced me to dive into OT environments and learned concepts that I never heard before. After researching, and with the help of my team, I started to understand the different functionalities and how all these equipment work together. It was a very important step of the project, specially to visualise the final solution.

3.3 Requirements

The main objective of this project relies on the recreation of some variables that are crucial in a specific industrial environment. These variables (i.e., Communication Protocols) will define the characteristics of a specific scenario which, as previously mentioned, will be the core for the PICSEL framework.

We abstracted these types of characteristics in which PICSEL will rely to better understand its main functions and role in the system:

- **Test Scenarios** after analysing and digging more in OT environments, it is clear that each industry has its specific characteristics, so, each test scenario should be a perfect mimic of them:
 - **Protocols:** with the amount of OT equipment and manufacturers came a enormous quantity of communication protocols for these environments. Some specific protocols are used in specific environments and types of industries. Therefore, it is important to know the specific protocols that are used in each environment because each protocol brings an entirely different attack surface. Industrial network protocols form the basis for communication between industrial network devices and a large number of industry specific network protocols that have been developed over the past decades, each designed for specific purposes and environments.
 - **Services,** OT environments as in IT environments depend on secondary services (e.g. NTP, email server, redundancy systems) and it becomes a very important variable in a specific environment. Since the goal of PICSEL is to recreate real processes, it is very important to implement all these details of the system, resulting in the expansion of the attack surface and allowing for different attack combinations.
 - **Network Scheme** after doing some research on different industries, testbed projects and security recommendations, it becomes clear that the network is another must have feature. This feature is due to all these types of environments having different requirements, such as old infrastructures, due to long update cycles, new implementations with external connectivity for remote management, poor security features on communication protocols and more. There are a couple of points that better describe the importance of this feature. Network Layers, as already discussed, are used in a vast majority of these types of environments/industries. Most of them still with a very primordial network infrastructure, meaning, networks without segmentation or any kind of security control like firewalls. Others already show some type of control like the separation between OT network and IT network. Finally, the most recent ones already comply with NIST Cybersecurity Framework and follow all security recommendations. Having these different typologies is important in order to have the conditions to recreate these different environments. Also, researchers could leverage on analysing how an attacker deals with different situations and how it could be prevented.

- **System Monitoring** is another essential feature, because in any system it is important to collect information. Gathering information of all equipment can be used to analyse, detect different behaviors and correlate all information within the different events that may occur. Based on the different types of equipment in these environments, there are different sources of information that are very crucial to gather. These different sources are based on the types of equipment previously discussed:
 - **Control equipment** these sources of information are important because they give a better perception of the process. For instance, information about the events happening on the PLC (e.g. Stop, Start, Running, Reset). Information like that can be a huge help to correlate data with other sources and facilitate the search of anomalies in the entire industrial process.
 - **Network equipment** another important source of information. In OT environments is a bit more reliable, because, most of these environments have a very static behaviour so its easier to spot an anomaly than in IT environments, since they usually have a more unpredictable behaviour. Monitoring this information, can help mitigate problems sooner and preventing others from future development.

3.4 Design Decisions

This section presents some possible solutions taken into account for the development and implementation of PICSEL. Also, some of these options are discussed and explained. In the end, the electrical scheme of PICSEL is presented and discussed.

3.4.1 Test Scenarios

In reality, test scenarios will be the main requirement for PICSEL.

In a small scale, PICSEL will have to show how a real industrial process works, meaning that it should explain how all equipment, protocols and services interact with each other in each specific scenario. As previously mentioned, there are many different types of industries and each one with different requirements and necessities.

There are different types of industries (e.g. Energy, Smart Buildings) and it is impossible to implement all. Here, some facts are going to be presented and discussed in order to choose what type of industries are going to be implemented. This step of the project is important because future decisions will depend on it.

With so many industries available and, since this project is inclined to security topics, it is important to analyse some annual reports about ICS from some major security firms. One of them, "The State of Industrial Cybersecurity 2018" from Kaspersky did a good job pointing out some relevant industrial sub-segments. Not forgetting that these segments

are all interconnected to various degrees. The most relevant ones for OT/ICS, with the percentage of the importance to an organization, are smart energy (51%), Industry 4.0 (48%), smart transportation (44%), smart metering (43%), and smart cities (43%) [18].

In these metrics it is easier to have an idea of the industries that are struggling more with security problems, not forgetting that they are all very important. Each of these industries have different specifications and, when implementing a specific scenario on PICSEL, difficulties can appear due to a variety of reasons, for example specific equipment, redundancy protocols or not compatible protocols. Besides these potential difficulties, it is important to analyse and experiment in different ways in order to recreate a specific industry.

3.4.2 Network Architecture

Network segmentation or subnetting, is the process of dividing a network into two or more networks, improving performance and its security. Considering a typical industry with different processes running, sometimes there are sections that are more critical than others, or sections that don't need to communicate with each other. If the same LAN is segmented for different sections, then the performance would increase as the unnecessary traffic would not move on each network segment, neither would an attacker, that somehow gained access to one of the segments.

This segmentation can be done in a variety of ways. It can be done at a physical level, where the two networks are divided physically at a wire-level, one for each LAN segment. Another way to achieve this is by the logical way, with software running in the router or by hardware using a switch. Finally, it can be achieved in the application level, for instance segmentation with firewalls. Some of these methods are further explained below:

- **Routers** are intelligent network devices, they can be configured to use the most efficient route to transmit the data. They work on layer 3 of OSI model, network layer, and the software can route data packets from one network to another based on their IP address. Each port of a router can be a separated segment, and routers are usually used to segment fairly large networks, in terms of geography or very high volume networks.
- **Switches** are data link layer devices that allow multiple LAN segments to be interconnected into a single larger network. Switches perform on hardware instead of software and therefore they are much faster than routers. Switches forward and flood traffic based on MAC addresses, layer 2. They learn the MAC address of the requester and the port or the location of the device which responded to the request, almost instantly. Switches can also be used to create VLANs, virtual segments instead of physically segmenting the network. The packets in a VLAN are sent only to the ports that are a part of the same VLAN.

- **Firewalls** can be used not only to segment the network but can also monitor the traffic that is passing applying policies. There are different types of firewalls:
 - Stateless firewalls watch network traffic and restrict or block packets based on source and destination addresses or other static values.
 - Stateful firewall monitor the full state of active network connections. This means that they are constantly analysing the context and data packets. Also, it allows to approve or restrict certain kinds of traffic.

Having all these possibilities, during the development of the electrical scheme presented at Section 3.5, it was considered that these options have identical power requirements. The fact of PICSEL is already handling some of these types of equipment it becomes possible to support another type of equipment.

3.4.3 System Monitoring

To maintain a normal operation and security of an industrial process, it is very crucial to collect data from all equipment. During operation, in order to detect and prevent unusual anomalies, it is very important to collect a good variety of information, that is why this module is also so important.

System monitoring, depending on the size and complexity of the environment, normally is responsible for controlling the technology in a process, such as hardware, networks, protocols, services, among others.

All that information can be used to: detect and alert about possible errors, correlate different types of data, forensics and help security solutions with data.

Meanwhile, to collect this type of information it is important to analyse all equipment and understand the information that is important to collect. Bellow is presented some of the most important information that should be gathered from the different types of equipment:

- **Network devices information** obviously gathering information about the network is very important. This information contains critical information about the status, errors, warning and configuration logs of the network devices.

This information provides details about the events, errors or any serious problems which can happen in a normal network infrastructure. Normally, to get this kind of information there are some standards and the most common ones are Syslog and Simple Network Management Protocol (SNMP).

- **Control devices information** as important as the network information. This information should give us a perspective about what is happening on the lower levels of the Purdue model. There is a lot of information that can be carved from these

devices such as alarms, errors, downtime, configurations, firmware version and others.

During the selection of the equipment, not all equipment allowed for this type of options and functions. To support all these functions, the equipment must support them.

3.5 Electrical Scheme

In this section, the core of PICSEL is explained and presented. One of the project main goals is portability, so it is very important to design a static solution for the equipment layout. Meaning that, all electrical components need a power supply and these components, especially in an industrial setting, have different requirements and specifications. Since these equipment is extremely expensive, it is very important to make an electrical diagram in order to everyone understand and be able to easily assemble it. This task requires that we follow each device specifications, and finally, PICSEL electrical scheme is described below.

Before further explanations about the electrical circuit, some labels were created in order to better understand all schemes. These labels are explained below:

- **PS** - Power Supply
- **C** - Controller
- **N** - Network Equipment
- **V** - Visualization Equipment
- **F** - Circuit Breakers

Figure 3.2 presents part 1 of the electrical scheme. Siemens 6EP1332-4BA00 is a power supply and it requires 230VAC of input, also, it comes with two outputs of 24VDC with a total of 3A. These outputs are going to supply the PLC S7-1500 that requires 1.9A and the other output is going to supply all network equipment, SCALANCE S615 0.3A, SC642-2C 0.38A, X005 0.5A and XB208 0.17A.

In Figure 3.3 is part 2 of the electrical scheme. Siemens 6EP1331-6SB00-0AY0 is the second power supply and requires 230VAC of input and has two outputs with 1.3A. The first output is going to supply the LOGO Controller that requires 0.165A, LOGO HMI 0.23A and KTP700 Basic Panel 0.23A.

Figure 3.4 is part 3 and the final of the electrical scheme. Siemens S7-1200 PLC has a built-in power supply, meaning, it can be connected directly to 230V.

This electrical scheme represents all of physical equipment that PICSEL is intended to run. All equipment must be properly connected to be used in safe conditions. With

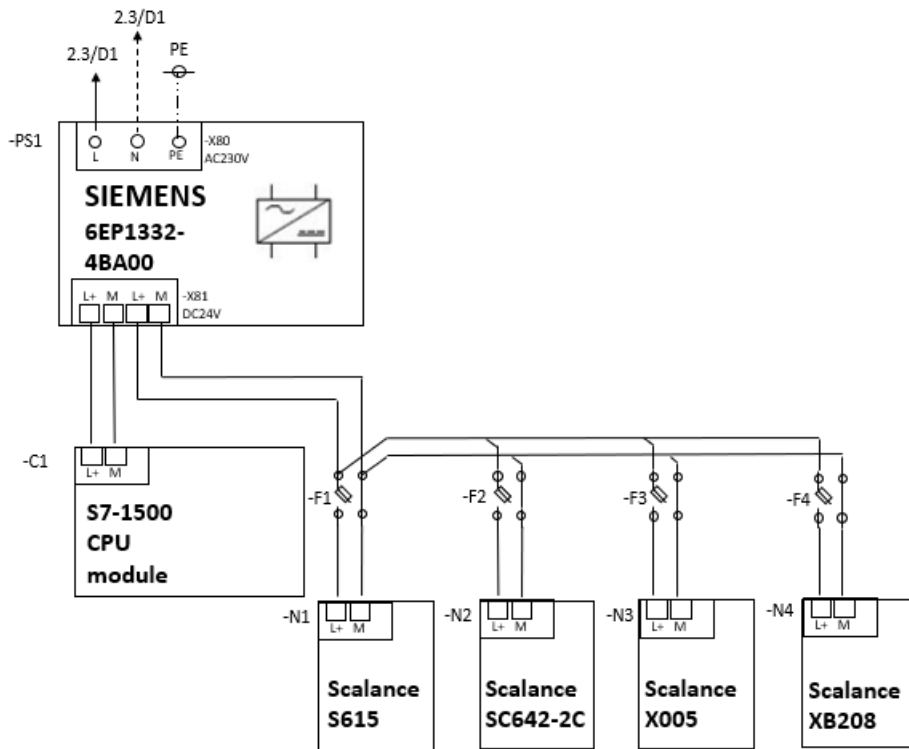


Figure 3.2: Electrical Installation Part 1

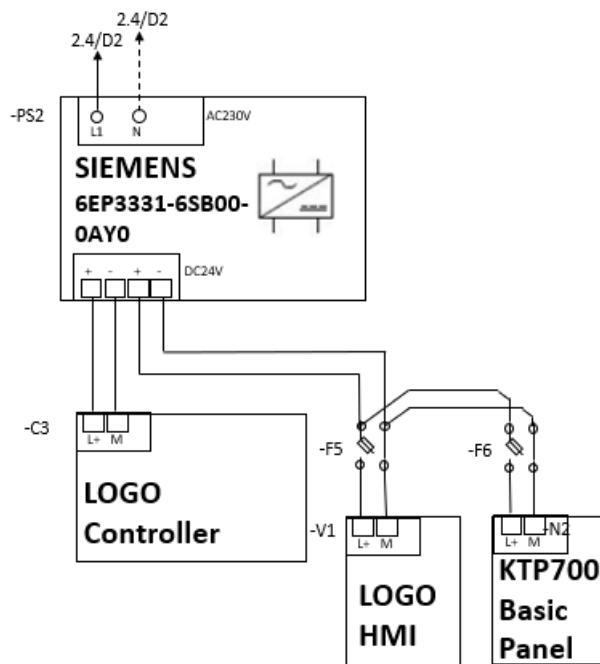


Figure 3.3: Electrical Installation Part 2

the physical core of the project deployed and working properly, it is possible to start testing some simple projects and start to understand how the equipment works and what is possible to do. In Figure 3.5 is all equipment connected and properly working.

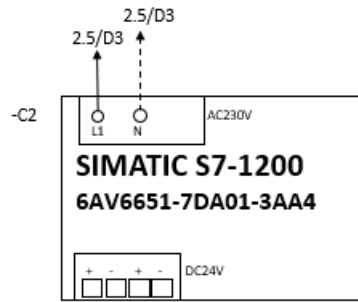


Figure 3.4: Electrical Installation Part 3

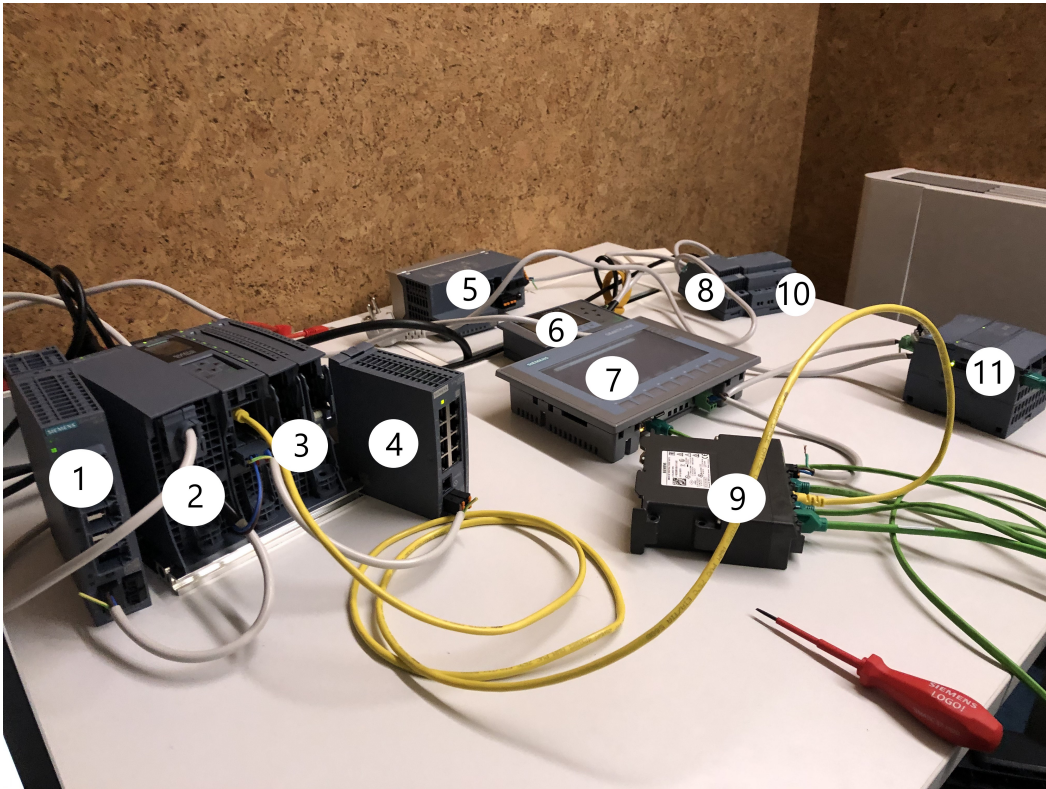


Figure 3.5: PICSEL equipment

- Label 1 - SCALANCE S6150
- Label 2 - Siemens 6EP1332-4BA00
- Label 3 - S7-1500
- Label 4 - SCALANCE XB208
- Label 5 - SCALANCE SC642-2C
- Label 6 - LOGO HMI
- Label 7 - KTP700 Basic Panel

- Label 8 - Siemens 6EP1331-6SB00-0AY0
- Label 9 - SCALANCE X005
- Label 10 - LOGO Controller
- Label 11 - S7-1200

Chapter 4

Implementation

In this Chapter, all final decisions are explained along with all configurations. It includes all the choices made, device configurations and explanations behind each decision. Also, the structure follows the previously discussed requirements.

4.1 Test Scenarios

This section will provide more insight on the chosen test scenarios: explanations of why a specific scenario was selected and an analysis of each specific environment in real-world situations in order to identify important characteristics so it can be implemented, the chosen scenarios were the two most relevant industries from Kaspersky report: smart energy and industry 4.0.

4.1.1 Industry 4.0

The first scenario, Industry 4.0 was the most interesting choice. Industry 4.0 refers to the current transformation of traditional manufacturing and industrial practises with the latest technology. This topic applies to typical industries such as food and beverage, water and waste water, among others.

In this case, it was decided to implement a simple process that recreates a beverage industry. This industrial process consists of two different lines of production, filling and labelling. This first process will transport an empty bottle through a conveyor belt. This bottle arrives at the first station, filling. When the bottle is filled, the conveyor belt takes the bottle to the next step. Moving on to the next station, if all relevant conditions are met, the labelling process begins. Once it is done, it leaves the production stage.

In Figure 4.1, is represented the architecture of this project. The architecture follows the Purdue model level 0, with sensors and actuators, level 1, control equipment (PLC), level 2, a NTP server and finally a control room that in this case contains windows machine running TIA Portal, and to supervise all operation an HMI is also present. The industrial protocols that we decided to implement was S7comm and Profinet.

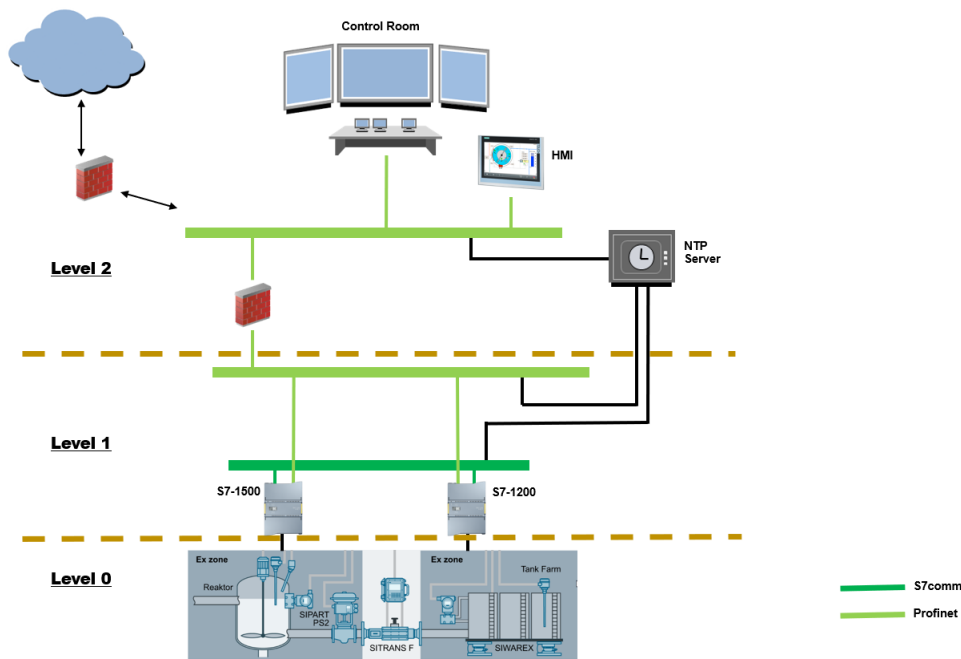


Figure 4.1: Industry Architecture

To implement this project, after all equipment is correctly assembled, the next phase will be performed in the TIA Portal, we will need to create a new project and add all equipment accordingly with the MLFB (Machine-Readable Product Designation) of each equipment. In Figure 4.2 is displayed the TIA portal interface where it is possible to do a variety of different tasks of which, but at the current time, "Create new project" is the only relevant one for us. After creating a new project, the devices must be added according to the MLFB of each equipment that is going to be used on the current project.

After adding all devices, we move to the project tree menu which is represented in Figure 4.3. On this menu, it is possible to check all devices along with other options. The most important options, for now, are: "Online access", used to check if the TIA Portal can reach the devices, "Add new device", at any time it is possible to add a new device to the project, "Devices & networks", all network configurations such as IP, web-server, ports, connections, protocols and much more.

In Figure 4.4, we can see the drop-down menu for each device, in this case, S7-1200. Inside this menu is a very important folder, "Program blocks". Inside it resides all the code that this specific device is going to run. Normally, this folder only contains the "Add new block" and "Main[OB1]" but in this case, it was already programmed. The "Main[OB1]" is like a normal main in another programming language. This main has all the code needed for the project and it can be divided by networks, like code sections. The CPU will perform a cyclic execution very typical in real-time devices such as this.

In the option "Add new block", the menu that appears is on Figure 4.5. There are 4 types of blocks: OB (Organization block), is the main; FB (Function Block), used

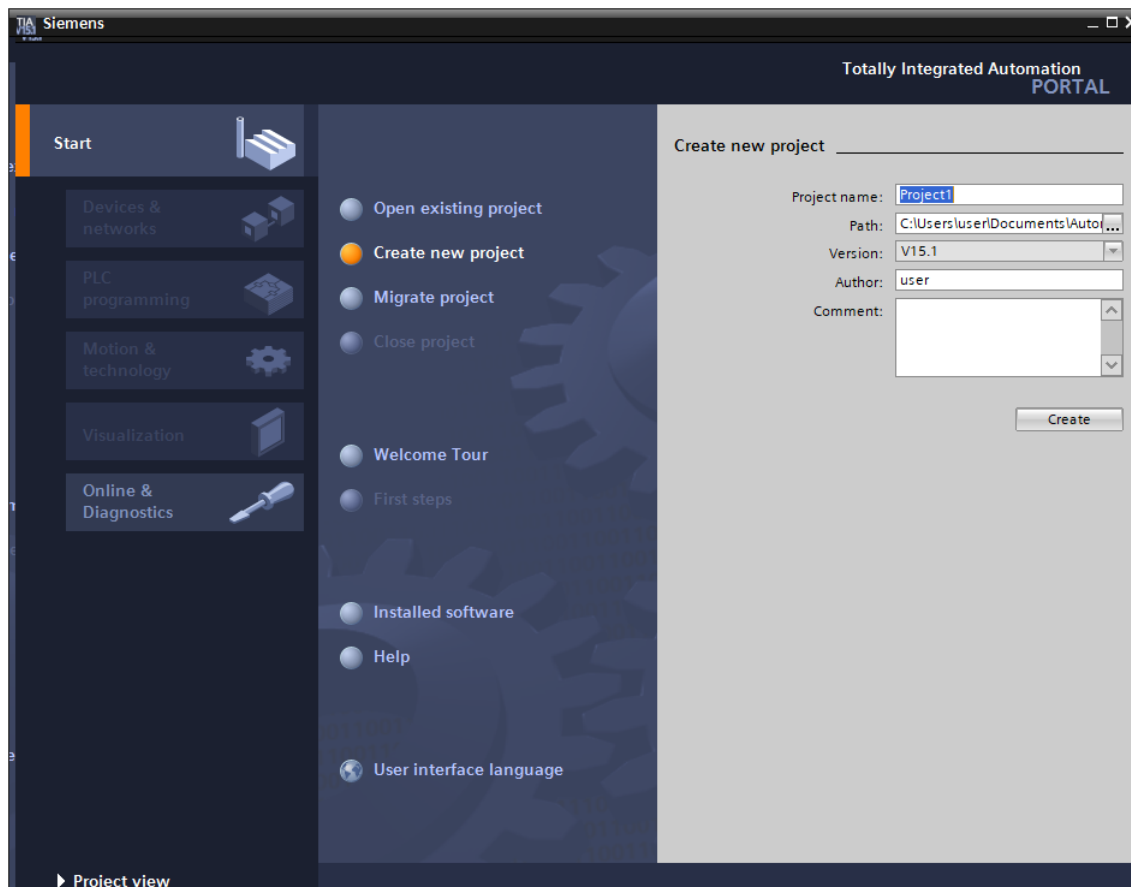


Figure 4.2: TIA Portal interface

to create functions with I/O and have a memory block; FC (Function), used to create functions with I/O and doesn't have a memory block; DB (Data block), is used to save information. Each of these types of blocks other than the DB can be programmed in the languages already discussed in Chapter 2.

In Figure 4.3 is the option "Devices & network". This option is where the communication between the equipment is configured. The first step is to configure the IP address of each equipment. Since we are going to use profinet and all devices have a profinet interface, we just have to enable the interface and specify the devices that are going to communicate. In Figure 4.6 is the network view of the project and the green line means that they are connected with each other.

After all the equipment is correctly configured and communicating with each other through Profinet, we can start implementing S7comm protocol. The implementation of S7comm is a bit different from Profinet, in Figure 4.7 is displayed a diagram for this specific environment. The communication will be between S7-1200 and S7-1500 - this type of communication works by blocks and requires a manual implementation on the user program. For the S7 connection, the S7-1200 acts as a client and the S7-1500 as a server. This means that the S7-1200 actively establishes the connection. The function

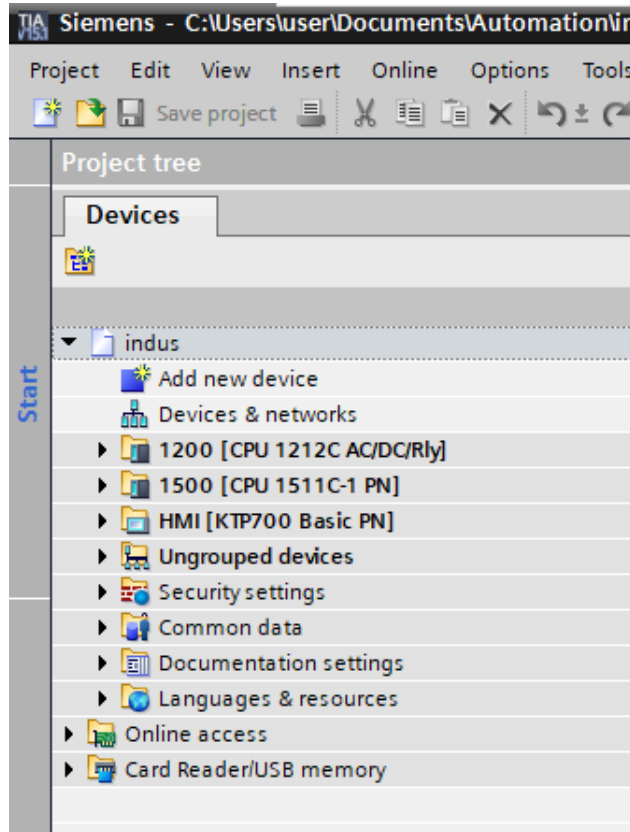


Figure 4.3: TIA Portal project tree

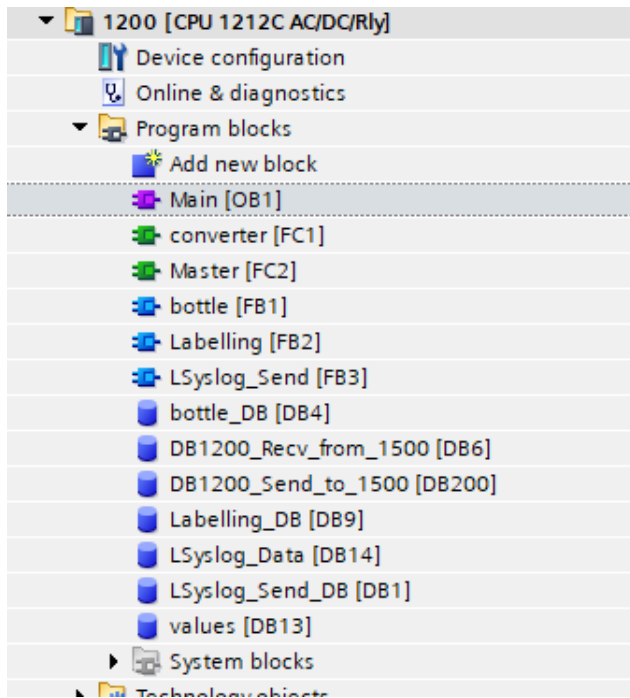


Figure 4.4: TIA Portal equipment menu

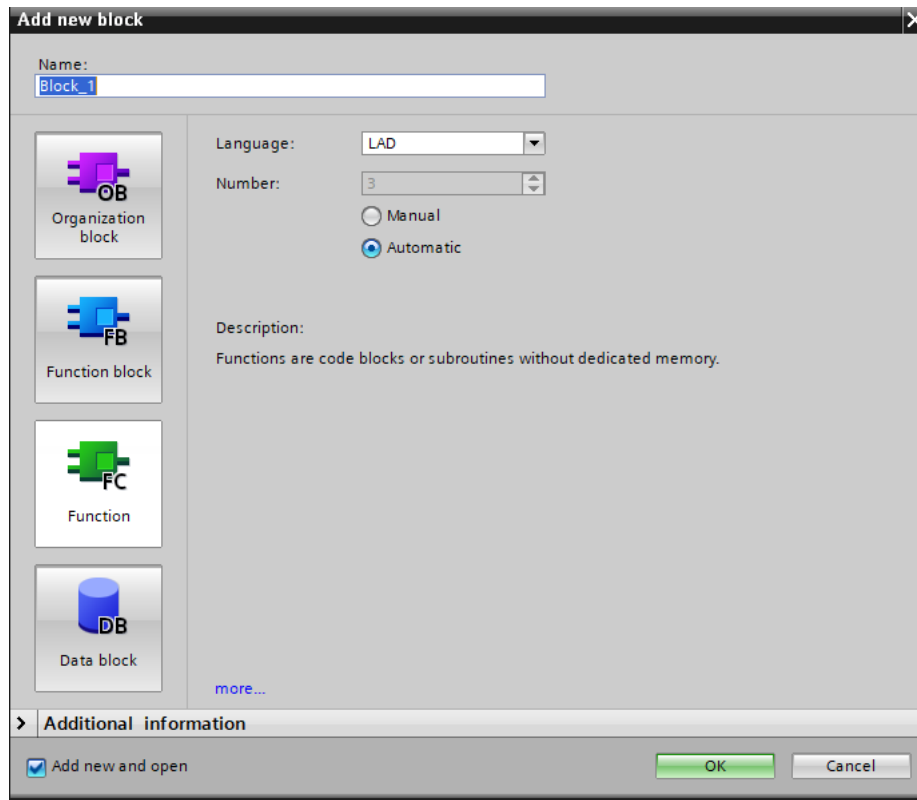


Figure 4.5: TIA Portal adding new block

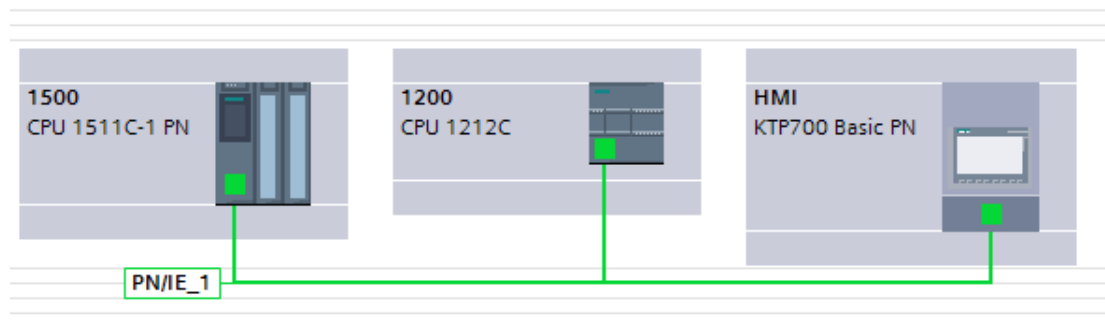


Figure 4.6: TIA Portal network view

blocks "PUT" and "GET" are called in the user program of the S7-1500 to read data from the S7-1200 and write data to the S7-1200. In order to implement such communication we follow all the documentation and standards of the function blocks.

After all connections are properly established, we can start to write the program for the industrial process. This process consists of two different lines of production filling and labelling that are controlled by the S7-1200. The filling line is waiting for an empty bottle to arrive through the conveyor belt, when an empty bottle arrives at the filling machine, the conveyor belt is set to OFF and the filling process starts if the bottle is on the right place and the reservoir has enough capacity to fill the bottle. Meanwhile, if these conditions are

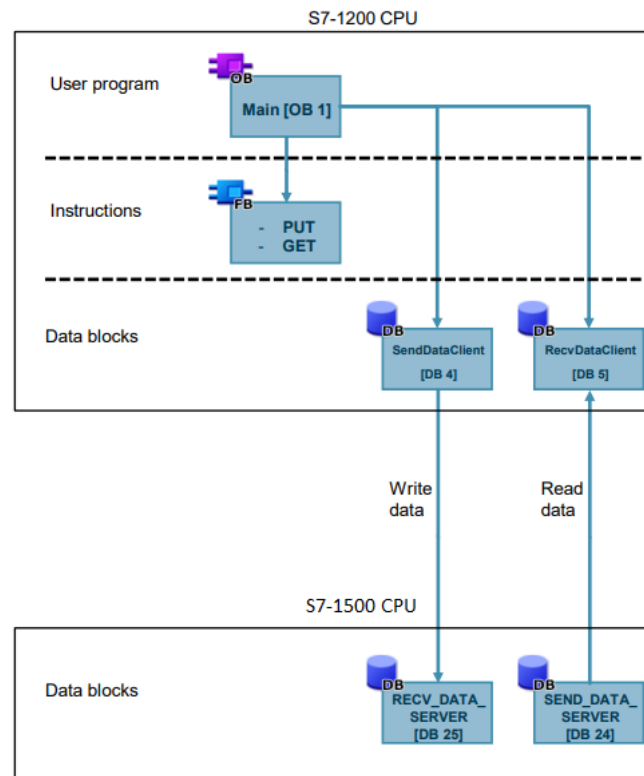


Figure 4.7: S7comm Communication

met, the bottle starts to be filled and when the process is completed, the conveyor belt is set to ON and the bottle moves to the next process line, labelling. In this stage, when the bottle arrives at the labelling machine, the conveyor belt is set to OFF and the labelling process starts. When the bottle is well labelled, the conveyor belt is set to ON and the bottle leaves that production line. The Filling station is connected to a reservoir that is being controlled by the S7-1500. This reservoir has a threshold to the minimum level and when this condition is triggered, it sends a command to S7-1200 to stop and the S7-1500 starts the filling process. When the tank is full, it sends a command to the S7-1200 to start the normal operation, filling. The communication between the HMI and PLC's is done by profinet and the communication between S7-1200 and S7-1500 with S7comm.

When the program is completed, we need to compile the project and send the information from each project to the specific equipment, this process is called Download. To download the project to the device, on TIA Portal project tree right-click on the device and select the option "Download". In Figure 4.8 is the download interface: first, we must choose the connection interface and then start the search. If everything is working fine the device will appear and we can load the project to the device.

Figure 4.9 we presents a print screen of the HMI. The objective of HMI is to give a perspective of the entire process to an operator. In this case, since everything is turned off, the sensors are red. The operator at level 2 can control what is happening at level 1.

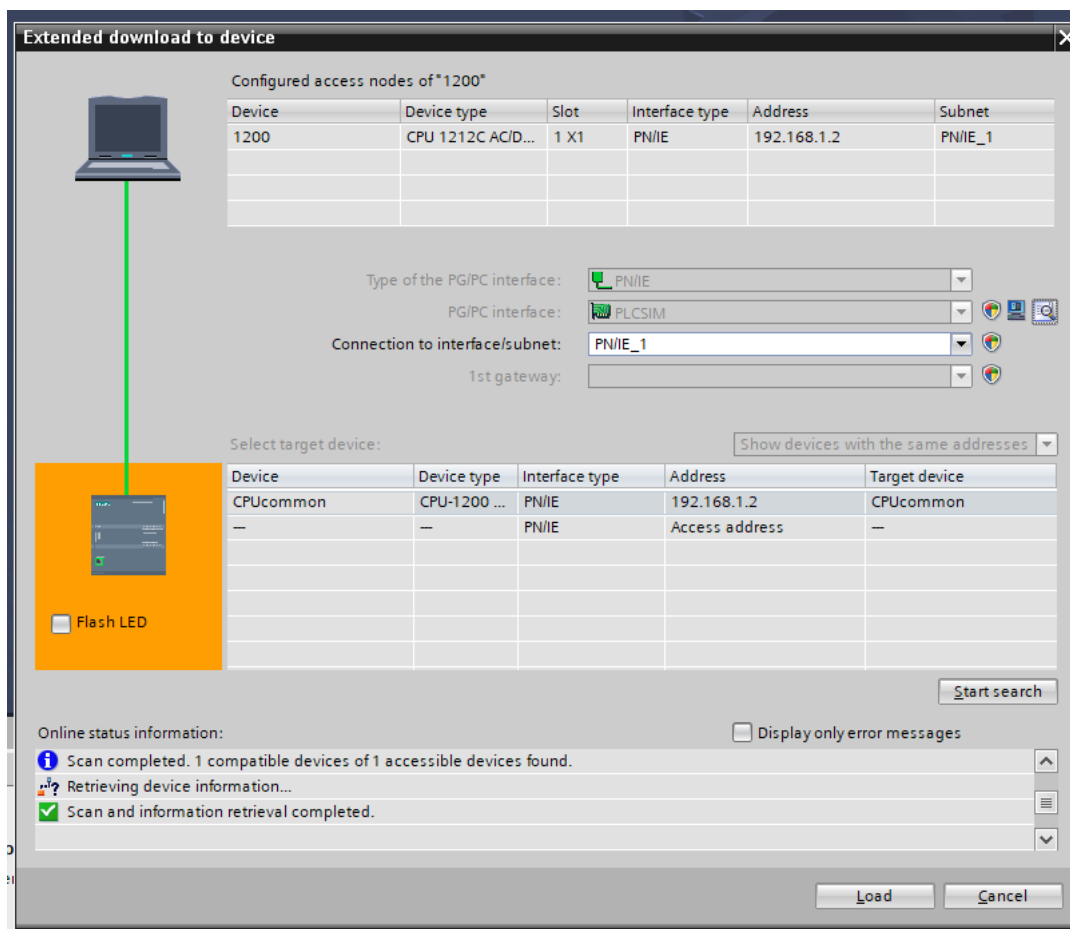


Figure 4.8: Download interface

The network configuration for this environment is shown in Figure 4.10. At the network diagram is represented Level 1 and 2 of the Figure 4.1. That firewall is dividing levels 1 and 2 following the NIST guide. In this case, three of the switches have a firewall, but for testing purposes, the equipment is all connected to the same network without any type of firewall rule.

4.1.2 Energy Management

Energy environments are responsible to ensure a secure and trustworthy flow of energy to a variety of places, such as homes, businesses, industries, and critical infrastructures. Having all these systems dependable on this sector, is probably one of the reasons to be one of the most critical ones. Even with this value, sometimes it is very difficult to improve the security of control systems hardware, software, communications and control infrastructure.

Normally, in a simple way, these environments are structured in stations and substations. Stations, or control centres, are where the operators receive and send commands. They can also control multiple substations through it. The larger amount of control equip-

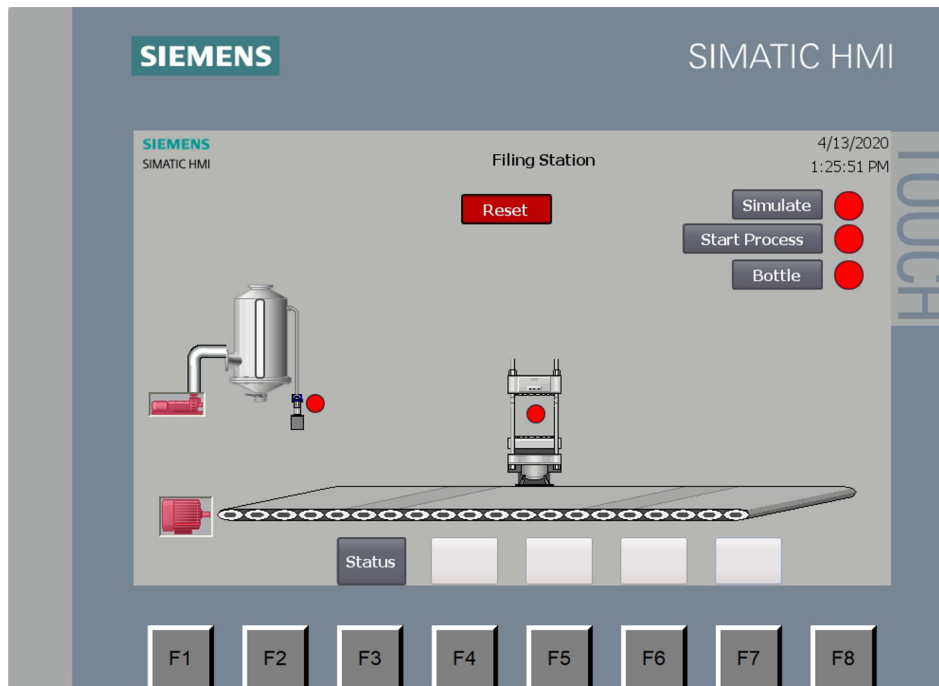


Figure 4.9: HMI print screen

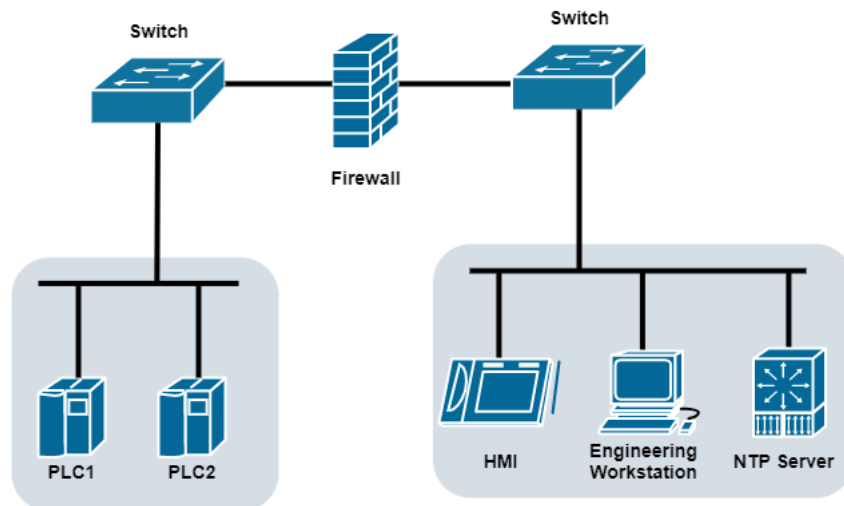


Figure 4.10: Network Diagram

ment is located within unmanned substations that must receive commands and send data about the process. Substations are usually located both in remote rural and urban locations and exist in a much higher quantity than the control centres.

While compromises to these control systems can have immediate impacts on electric grid operations, significant grid impacts can also occur when compromised remote and unmanned substations are used to access and manipulate local or networked systems and equipment. Evaluation of this substation control system equipment from a security perspective is deemed necessary in order to gather all information supporting this industry

sector. PICSEL can help improving the security regarding the energy sector, especially for SCADA systems, through research, system assessments, testing, validation, training, and outreach of new discovery.

Normally, this type of critical infrastructures follows some industrial standards. In order to better understand these type of environments, there were several meetings with energy infrastructures experts from Siemens. These meetings, combined with some research, allowed the understanding of the key components and how they work together to make it possible to start designing and implementing the test scenario.

Typically, these environments (Figure 4.11) have a regional control centre and substation control system. Taking this into consideration, multiple substation control systems can be communicating with the regional control centre. This communication between stations follows the standard IEC 60870 and, inside each substation, the communication standard is IEC 61850.

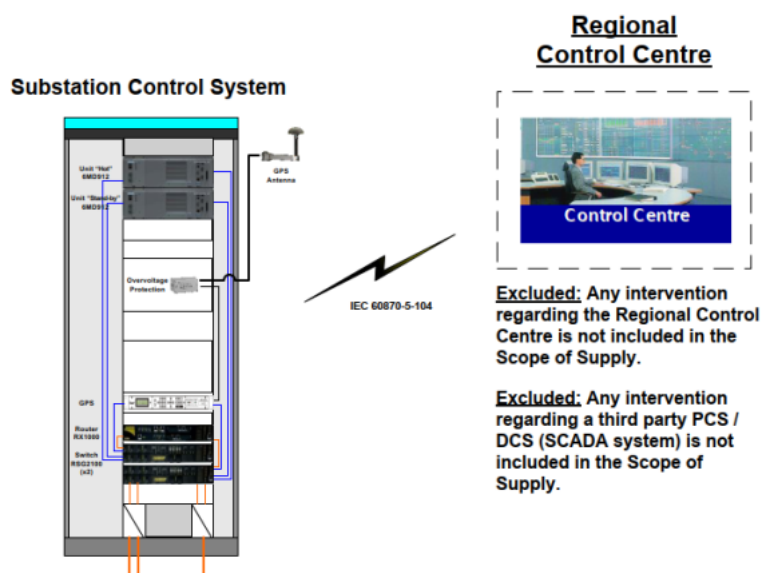


Figure 4.11: Energy topology

While researching more about this kind of infrastructures, we went to a pre-production facility from a Siemens client. This client provides equipment and infrastructures for some energy providers in Portugal, allowing to ask some questions about network layout, communicating ports, network protocols, services and equipment. With all those questions answered, it was possible to start designing all the environment. In Figure 4.12 is displayed the typical energy architecture according with the information gathered about these environments.

It is important to understand that this environment in Figure 4.12 represents a very simple architecture from the real process. During the analysis of such environment, different security systems, redundancy protocols and equipment were detected. Being impossible to handle all these variables, a simple approach was developed.

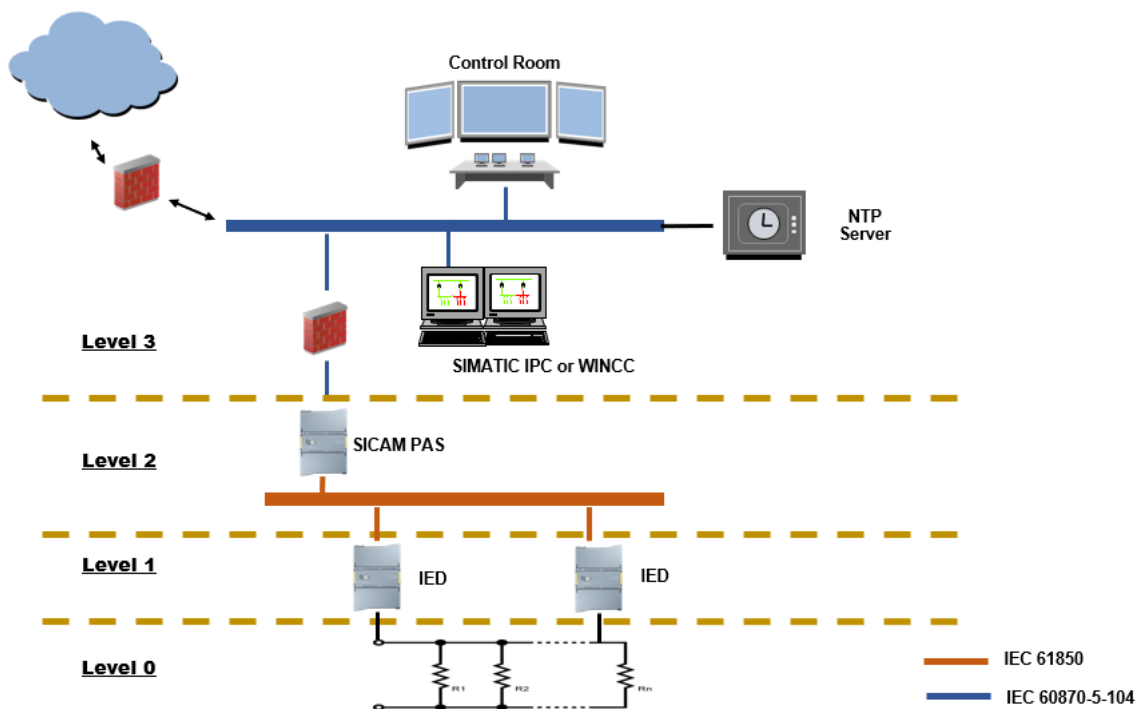


Figure 4.12: Real Environment

There are some particular differences in this particular environment. Level 0 is represented with an electrical scheme, level 1 instead of PLC there are IEDs that in this case represent SIPROTEC equipment - this is the Siemens product family for protection relays for digital substations. They are very common in field devices and are used for protection, control, monitoring, and measuring applications in electrical energy systems.

For some years now, power generation and distribution have been undergoing major changes. The innovation cycles are getting even shorter, and the market is becoming increasingly deregulated due to the number of new manufactures and different equipment. The systems used to monitor power supply equipment and processes must take into account these changes. At level 2, is the SICAM PAS (Power Automation System) that allows manifold interfacing to various communication media and extensibility supporting multiple communication protocols and different types of equipment.

At level 3 is the Control Room where the management of all the substations takes place. It also includes the main NTP server. Another Siemens product is the WinCC that is a scalable process visualization system with numerous functions for monitoring automated processes, the Siemens SCADA system. Whether in a single-user system or a distributed multi-user system, it offers complete functionality for all industries and for highly complex visualization tasks for SCADA applications. During research, one of the very important points for them to use the WinCC was the highly configurable framework it offers. In Figure 4.13 is presented a real scenario of the lab, data visualization is very important because it is where the operators will supervise and control the entire system.

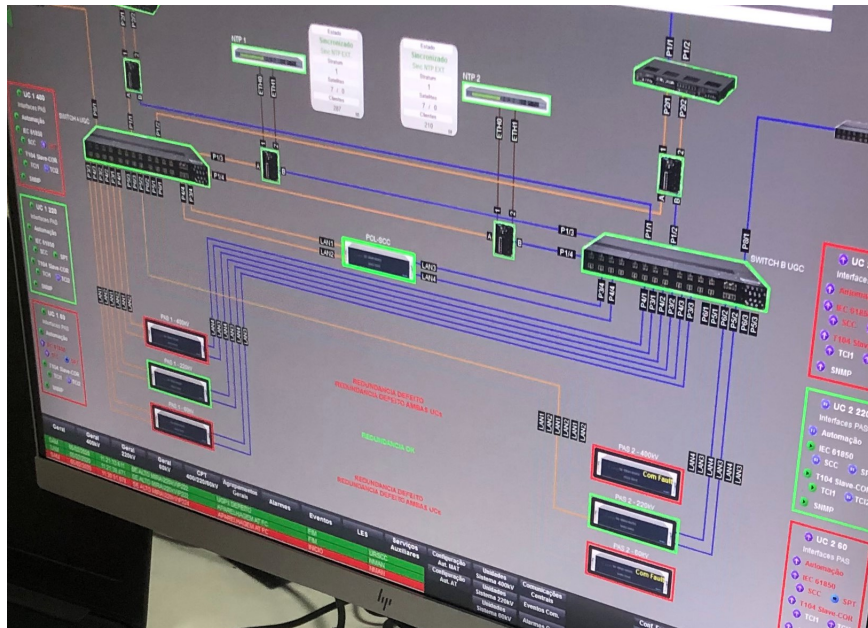


Figure 4.13: WinCC of the Pre-Production Lab

In Figure 4.14, is an energy architecture that follows the Purdue model. Levels 0, 1 and 2 are where the substation control system is located. This substation consists on a server of the IEC 61850 that is collecting information about the process, Level 0, and sending it to the client at level 2. At Level 2, is the client of IEC 60870-5-104. This client is located at the substation and is responsible to bridge all information to the IEC 60870-5-104 server at Level 3, regional control center.

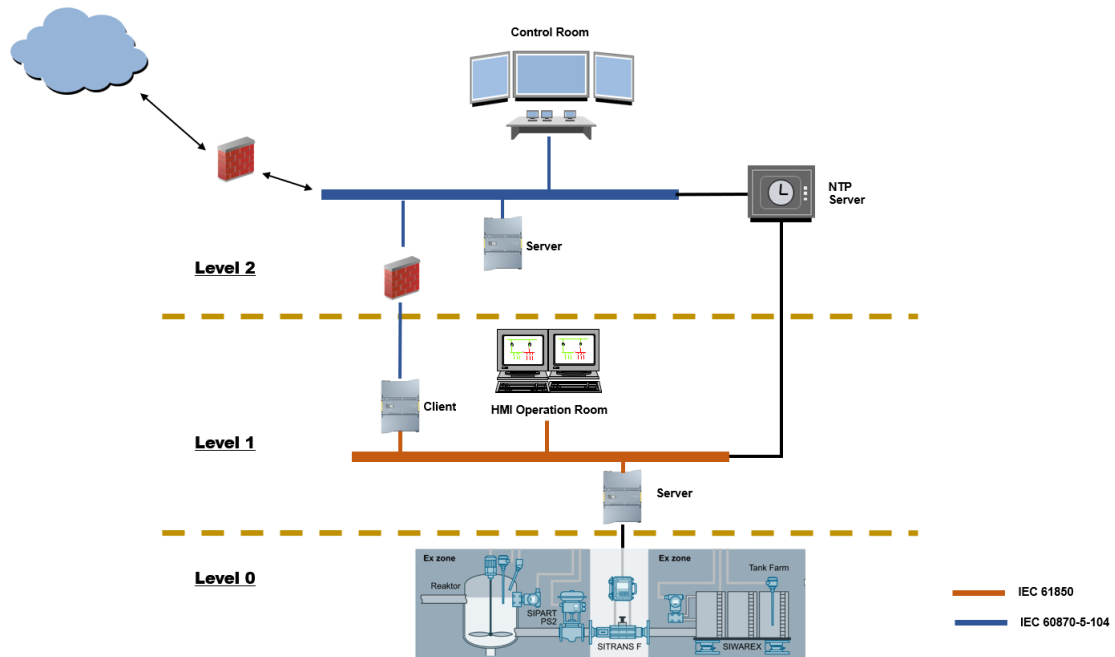


Figure 4.14: Approach to an Energy Architecture

This communication allows all clients to send and receive commands and data. The type of messages configured in each protocol were based on the data gathered on the pre-production facility. Based on all information, the same type of messages, communication ports and the network layout follows the same structure.

During the deployment of this particular scenario, some difficulties were encountered. These difficulties were mainly caused by us not having all the equipment used in real scenarios. Such a limitation didn't allow the full implementation of the scenario and communication protocols between the devices. To mitigate the situation, one possible solution was to implement the protocols using the available PLC's (S7-1200 and s7-1500). Yet another problems surfaced, the PLC's don't support these types of communication protocols and require additional CP (Communication Processors) for example in order to the S7-1200 to communicate IEC 60870-5-104 it requires a CP MLFB:6GK7243-1PX30-0XE0 just like the S7-1500. Buying all this additional material would be very difficult and extremely expensive.

By this point, the research was already finished, it was decided to implement both IEC 60870-5-104 and 61850 with an open-source library that provides a high-level API. This allowed the implementation of the client and server of each protocol in order to understand how these protocols work. During the implementation, we follow the same type of messages used in the real scenario. Each client and server of the different protocols were tested in different Linux machines with the PICSEL network equipment.

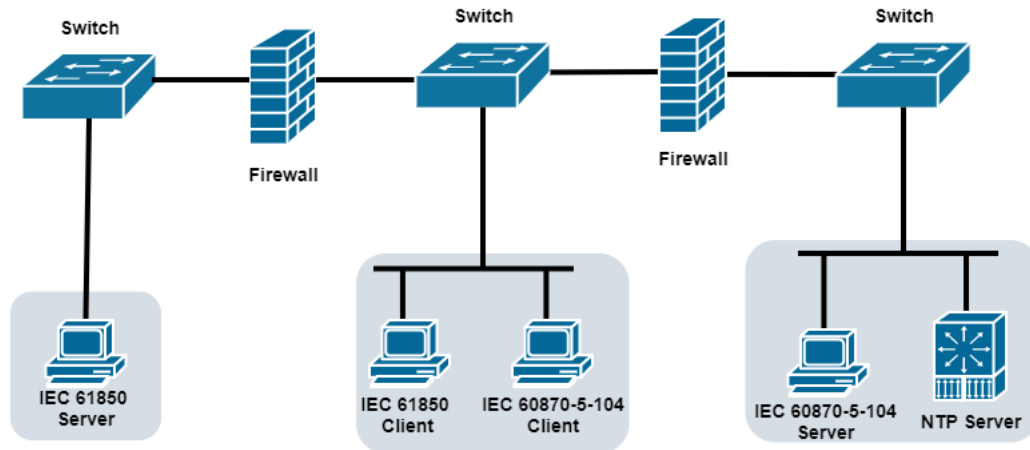


Figure 4.15: Network diagram

Figure 4.15 presents the network diagram. The network segmentation is very important in these type of environments, hence this diagram is very similar to a real one. Since our available equipment does not include router devices but it does include switches with firewall capabilities, the segmentation was done through the firewall. There are some differences between the pre-production environment and this one, even if the network layout is pretty much the same, there are crucial differences. They have a strong hardware redundancy for every single network equipment and redundancy protocols are also used. It

is very important to remind ourselves of this fact because this project is trying to recreate an entire industrial environment with low resources.

4.2 Network Configuration

Since both scenarios require the same network configuration, this section is dedicated to understanding how the network equipment is configured.

In the first stage, after the network equipment is turned ON it needs to be configured. The easiest way to do so is with Primary Setup Tool, a Siemens tool that is presented at Figure 4.16, this tool allows to set an IP address of the device. In Figure 4.16, the IP address was set to 192.168.0.30 which allows access to the web-server. In the web-server, it is possible to configure everything, from creating VLAN's to configure firewall rules.

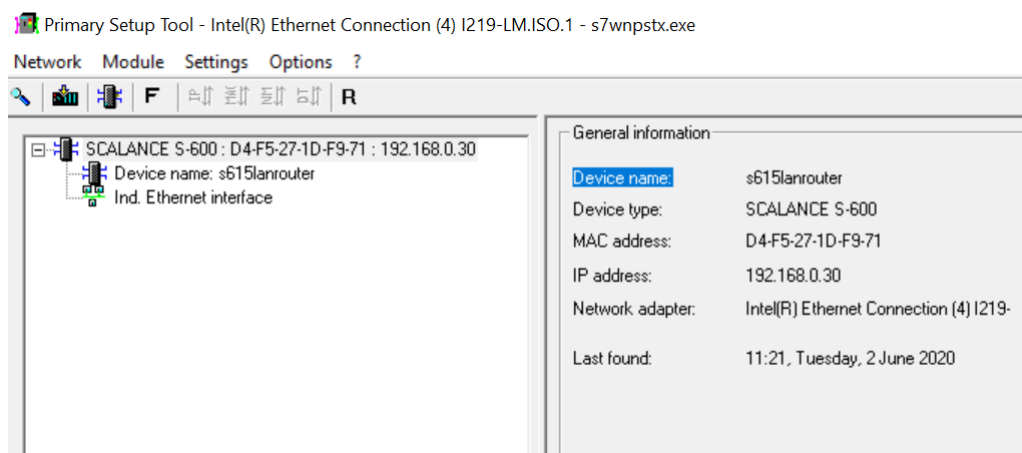


Figure 4.16: Primary setup tool

After properly configuring the IP address of each equipment we should have access to the web-server. To do so, we need an internet browser and access to the already configured IP address, in this case, <http://192.168.0.30>.

For the network configuration, it was decided not to implement any firewall rule or to segment the network to test everything without any type of problem or miss-configuration. Having a flat network allowed us to have full control and access to all equipment connected in order to perform tests without any kind of restriction or problems.

4.3 System Monitoring

Another important objective of PICSEL is to collect as much information as possible about the network and equipment. This information can be useful to correlate information in order to detect anomalies in the system.

Anomalies can be originated from malicious behaviour or normal problems. These sources of information can be very important to security solution in order to parse and correlate the received information with e.g. network traffic. In the following subsections, based on the two types of industrial equipment, an explanation of each type of information gathered and the mechanism used to send it.

4.3.1 Control Equipment

Control equipment is present at levels 1 and 2 of the Purdue model, the most critical levels. It is important to keep track of everything, especially in those levels. After doing some research, we noticed that there wasn't a lot of information to collect from these devices.

In order to solve this problem, we started checking what type of information a PLC was able to generate and the most important information. There were two very interesting options, checksum functions and diagnostic buffer - both functions represent important data from the PLC.

The diagnostic buffer data in a PLC is a buffer that includes all events of the operating system and information about the running user program. Unfortunately, we were not able to collect this type of data, after digging into all kinds of functions there were only two ways to read this data, by the web server if during the development of the project this option was properly configured, and with the TIA Portal program.

Another option was to get the diagnostic buffer information of the modules connected to the controller, we found a way to get this information from the PLC modules. Normally there are different types of modules such as CP, additional I/O, robotic arms, safety equipment and others. When connecting these modules to the PLC, the PLC creates a global variable for that specific module and through that it is possible to use that hardware variable to map "GET_DIAG" FB. In order to configure this FB besides the DB it is also necessary a very specific data structure to save this information.

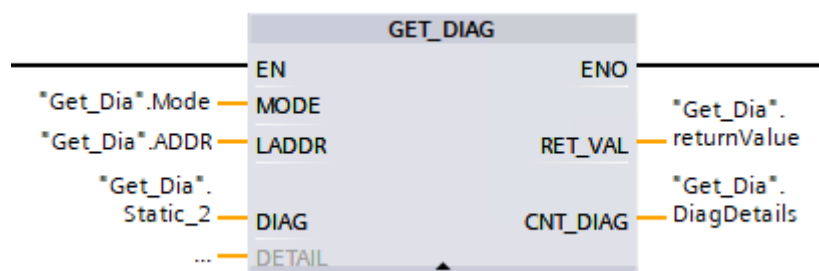


Figure 4.17: GET_DIAG FB

To understand the diagnostic information, the hex values must be converted to binary code. At Figure 4.18 is represented the DB for the "GET_DIAG" FB. At DIAG parameter is the DIS data type that is represented by that specific structure. When deploying a project

it is possible to monitor the values in TIA Portal via the "Online Mode" and it is possible to see the current value at the "Monitor Value" column. The meaning of the current values can be found at the function documentation, but in this case indicates: MaintenanceState: According to the value "0", the CPU requires no maintenance. ComponentStateDetail: According to the hex value "0000_8000", bit 15 is active by default. OwnState: According to the value "0", no fault has occurred. IOState: According to the hex value "0001", there is no maintenance required. OperatingState: Outputs "0" because we don't have any additional module and we were using a I/O and the OperatingState always has the value "0" for I/O.

My_gDB_GET_DIAG				
	Name	Data type	Start value	Monitor value
1	Static			
2	diagMODE	UInt	1	1
3	myLADDR	HW_ANY	50	50
4	returnValue	Int	0	0
5	CountDiagDetails	UInt	0	0
6	myDIAG	DIS		
7	MaintainanceState	DWord	16#0	16#0000_0000
8	ComponentStateDetail	DWord	16#0	16#0000_8000
9	OwnState	UInt	0	0
10	IOState	Word	16#0	16#0001
11	OperatingState	UInt	0	0

Figure 4.18: DB from GET_DIAG FB

Other curious information is the checksum value of each PLC. To retrieve that information, we used the Checksum function, which outputs a hexadecimal value, corresponding to the checksum of the project running on the PLC. This means that, if a different project is running on that PLC, the checksum value is going to be different. In Figure 4.19 is the checksum FB with the created values in a DB after the implementation is properly configured by the manual.

Not having any modules connected to the PLC's the "GET_DIAG" FB isn't very useful. So we decided to go with the "GetChecksum_Instance" FB because the output is only a hexadecimal number and it is easier to process and send.

For the information to be parsed and analysed it is important for the PLC's to be able to send it. To do so, we use the Syslog protocol for a couple of reasons. First, the newest firmware versions of PLC's just add support for this protocol, second, it is a simple protocol and can be easily parsed by any tool allowing any security solution to use it. Figure 4.20 represents a checksum function block.

4.3.2 Network Equipment

Network equipment is spread through a variety of different Purdue levels and can supply very important information. By analysing the possible configurations it is possible to

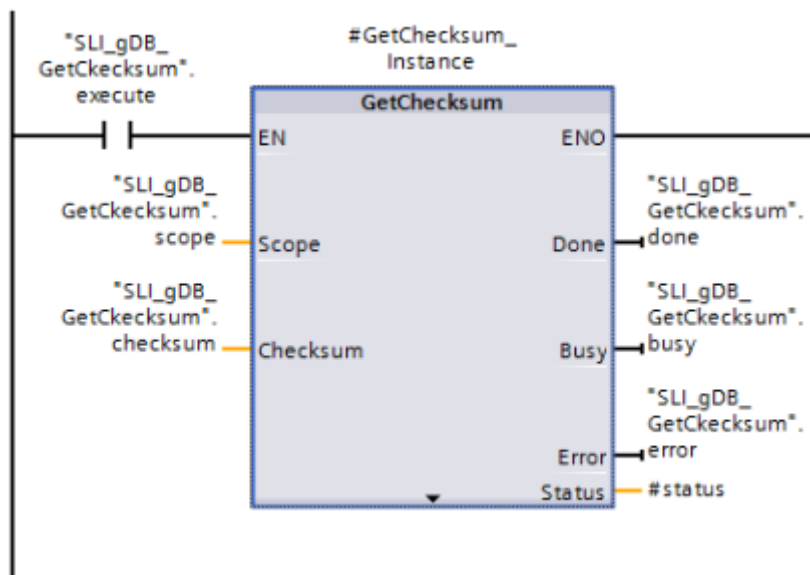


Figure 4.19: Checksum function block

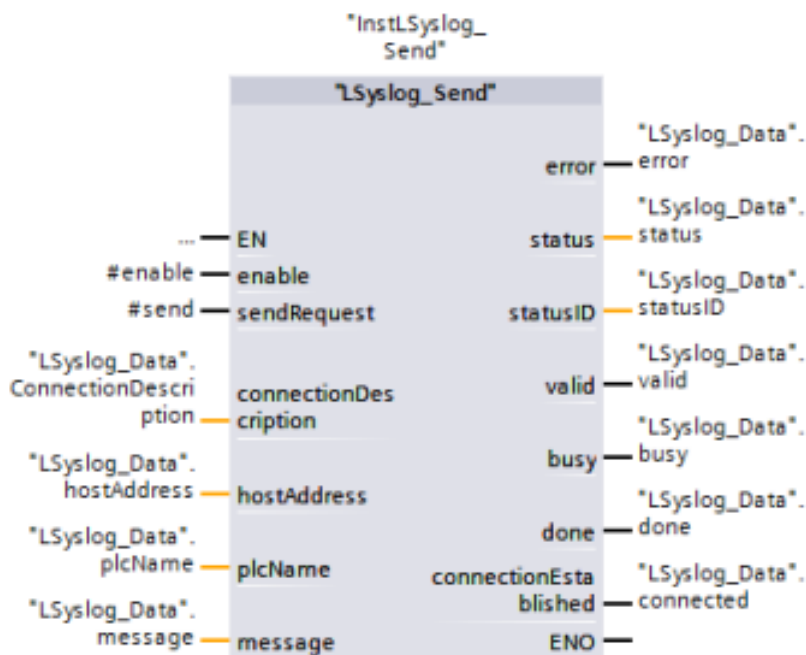


Figure 4.20: LSyslog_Send function block

collect events with some very useful information. Events, such as:

- **Link Change** - This event occurs when the port status is changed.
- **Authentication Failure** - This event occurs when access is attempted with an incorrect password.
- **Power Change** - This event occurs only when power supply lines had a change to line 1 or line 2 (Some equipment may have more than one input).

- **Loop detection** - A loop was detected in the network segment.

Following the same line of thoughts from the control equipment, the information should be sent all to the same place. To do so, the Syslog option was set and configured. To do these configurations only required access to the equipment web server and set "Syslog Client" options, and set the IP and PORT address for the Syslog server. With all configurations done, each equipment redirects all chosen events.

Also, another PICSEL goal is to test possible security solutions. Another solution could be an IDS, for example. In this case, the IDS should have access to all ports and VLAN's traffic in order to be analysed. This problem was solved with the strategic placement of a port mirroring. Port mirroring allows the switch to send a copy of network packets from a specific switch port to another switch port. Giving full capabilities for a security solution to monitor all traffic.

4.3.3 PICSEL Layout for System Monitoring

Figure 4.21 presents a diagram of the information flowing at PICSEL. The information will be sent to a server and there it can be parsed and analysed. A good central point to test a security solution that correlates information about the network and the information received from all equipment.

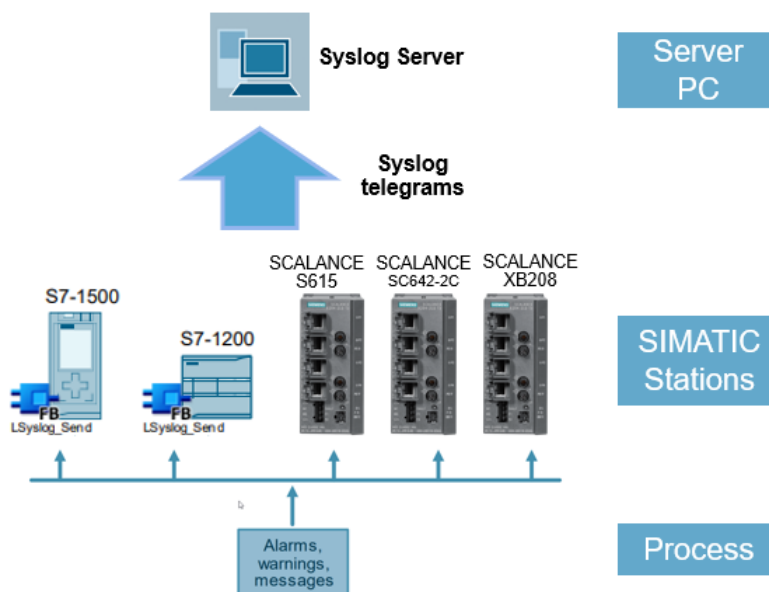


Figure 4.21: Information Gathering

Chapter 5

Experimental Evaluation

In this chapter, the potentials of PICSEL will be presented and the following goals will be evaluated. These goals: testing new security solutions, perform security tests and test possible vulnerabilities. Also, an assessment of potential impacts is crucial and can also give us a better understanding of how everything works. Meaning, performing these tasks can be useful for addressing real-world situation.

5.1 Experimental Objectives

The objective of PICSEL is to help to: perform new exploits and study how protocols and equipment communicate with each other, demonstrate the effects on a production line, and the possibility to mitigate and detect these attacks. These types of occurrences can be caused by flaws in various components, such as hardware, firmware, communication protocols, and possibly multiple other variables related to the underlying environment. The objectives of these experimental evaluations are to show that PICSEL's objectives are fulfilled.

5.2 Evaluation Criteria

The main criteria for evaluating the experimental results and conclude about the achievement of the defined objective are the following. For the first goal, an experiment to be considered a success must allow the analysis of the environment and, if possible, develop or follow a POC that affects the expected behavior. For the second goal, the conditions must be met in order to effectively test exploits already in the wild and assess if they represent a real threat to the environment. Finally, to mitigate or detect some of these attacks, based on the information gathered from PICSEL, an appliance must detect some anomalies caused by malicious behavior.

5.3 Experiments

In this section, all experiments will be described and thoroughly explained. We are going to prove that the PICSEL goals were accomplished and also provide a discussion about the overall results.

5.3.1 Control Command Injection Attack on IEC 60870-5-104

This first experiment will be performed against the test scenario Energy Management. The type of attack is a command injection, this attack was based on a proof-of-concept [14]. This was considered into the experiment to demonstrate the impact of the lack of authentication and encryption in industrial communication protocols, in this case, IEC60870-5-104 protocol. The objective of this experiment is to evaluate a PICSEL goal, namely the possibility of injecting an attack and observing its impacts on a production line.

Packet injection is a computer network attack that refers to the process of interfering with an established network connection, through constructing and injecting packets with malicious payloads into a network, in such a way that these specially crafted packets appear as part of the normal communication stream.

This attack consists of five main steps, that are:

- Break into the SCADA system (phishing, physical breaking into the place, internet exposed systems, etc);
- Analyse the environment (i.e identify all equipment communicating IEC 60870-5-104 and also the type of messages being exchanged);
- Hijack the connection between the two devices;
- Start a new valid connection;
- Send crafted packet to start communication for IEC 60870-5-104 protocol;

First, only for this project goal, there are a few preconditions for the experiment and one of them is assuming that the attacker already had gained access to the SCADA system Figure 5.1. These type of breaches can happen in a variety of ways and one of the most common is by phishing campaigns, but for this example, the way that the attacker gained access is not relevant.

The attacker, already inside the network, starts sniffing the network in order to analyse it and to identify possible targets. As shown in Figure 5.2, the attacker is seeing the packets being exchanged between two IP addresses. Furthermore, these two IP's are also communicating in IEC 60870-5-104 and by analysing the protocol it is possible to identify both client and server. The normal behaviour of the protocol, is that, initially, they always start a TCP connection and when the client wants to start communicating by IEC protocol

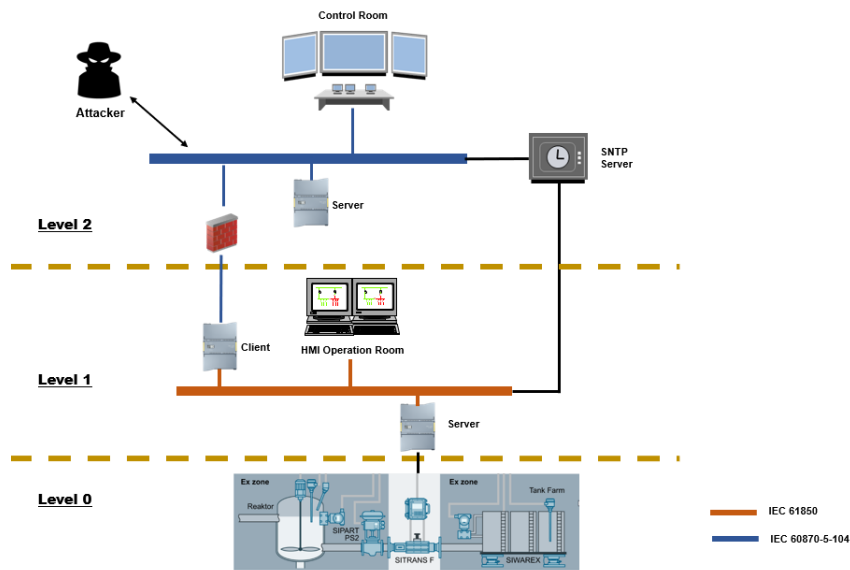


Figure 5.1: Attacker inside the network

he must send an IEC packet because, data transfer is not automatically enabled in server-side, this state is called STOPDPT and is the server default state. Following this line of thoughts it is possible to identify the server IP=192.168.94 and the client IP=192.168.74.

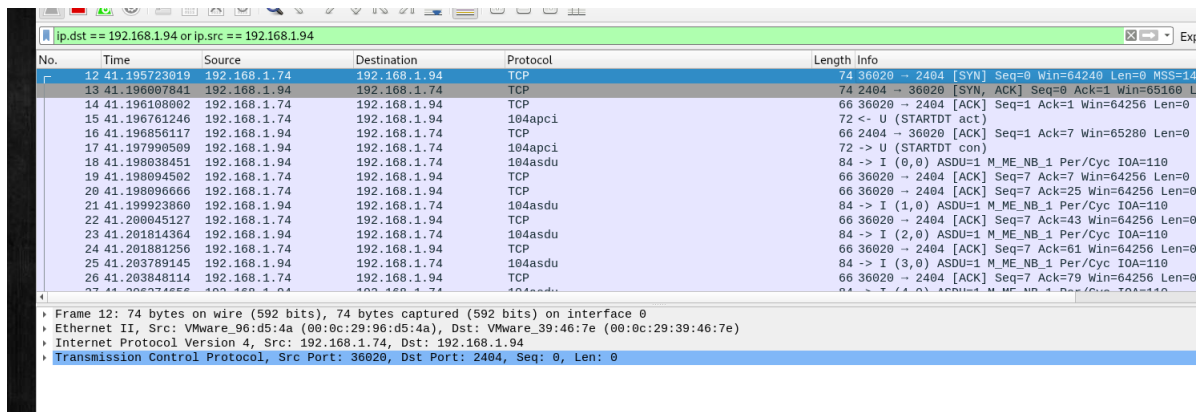


Figure 5.2: Wireshark showing the network communication

The client must activate the data transfer by sending a "STARTDT act" (activate) packet to the server. This is important because when the attacker establishes the new TCP connection the first packet that he must send is this "STARTDT act" message. In Figure 5.3, shows the "STARTDT act" packet that allows the attacker to easily identify and recreate the payload to successfully enable data transfer to the server.

After "STARTDT act" packet the communication follows a specific standard as shown in Figure 5.4. In order to the attacker be successful he must understand how it works. Thus, both client and server have two counters a V(S), send state variable and a V(R) receive state variable. To inform that the communication is going smoothly, they keep

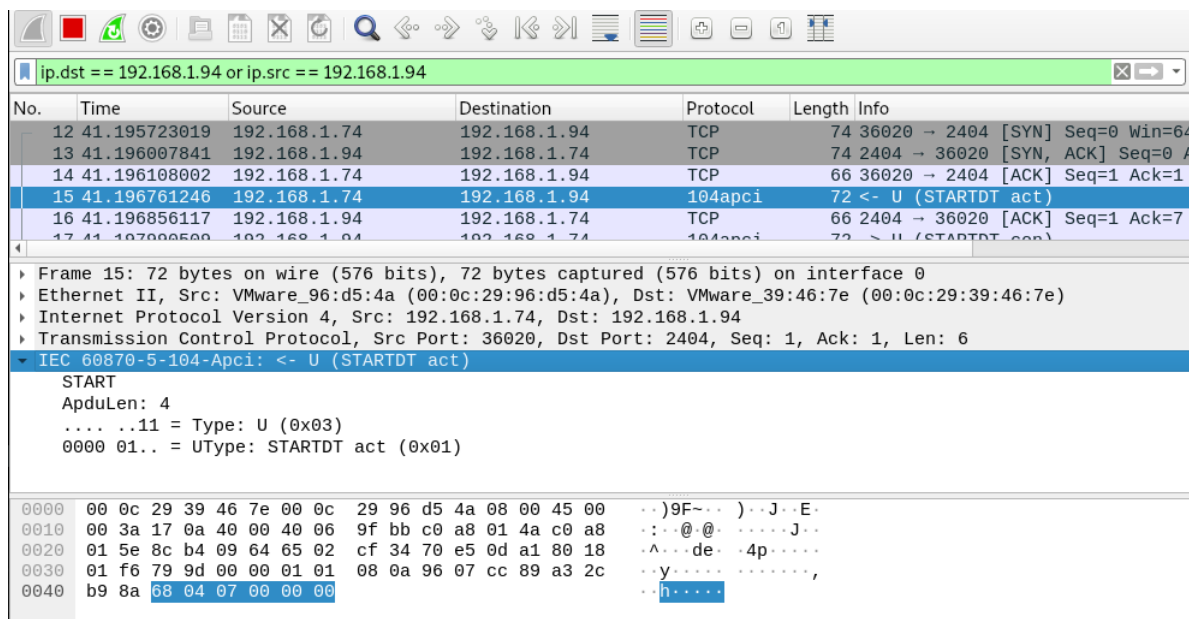


Figure 5.3: Payload for the "STARTDT act"

exchanging the counters information and, to do so, they use a message following format, I(a,b) is the information format of APDU, meaning, a = send sequence number V(S) and b = received sequence number V(R). This is important because the attacker must follow this structure for the packet payload, otherwise the packet will be dropped.

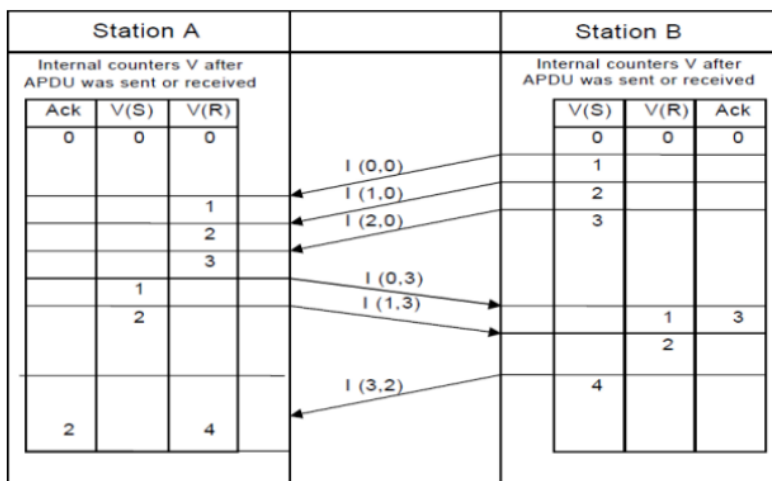


Figure 5.4: Undisturbed sequences of numbered I format APDUs

Finally, after this brief explanation about IEC 60870-5-104 protocol and some network analysis, the attacker can start building the exploit. The exploit in Figure 5.5, consists of three steps breaking the connection between the IEC devices, starting a new connection and finally sending the malicious command.

Based on the previous steps, the first stage is to break the connection between the

```

root@kali [~/Documents/Attacks/IEC 104] master + python3.7 final.py
TCP reset attack finish
Connecting....
###[ Raw ]###
load = 'h\x04\x07\x00\x00\x00'
Sending starter packet!!!!
Sending payload!!!!
root@kali [~/Documents/Attacks/IEC 104] master +

```

Figure 5.5: Exploit

client and the server. To do that, we used a technique called TCP reset attack, basically it is a way to tamper and terminate the connection by sending a malicious packet containing a TCP reset flag, allowing to interrupt the connections between two parties. In Figure 5.6 is represented the network traffic at the time of the attack with Wireshark. Where it is possible to see the normal connection and when the attack begins the red packets it is possible to see the RST flag, closing the connection between client and server.

222	37.572648590	192.168.1.94	192.168.1.74	104asdu	84 -> I (108,1) ASDU=1 M_ME_NB_1 Per/Cyc IOA=116
223	37.572777679	192.168.1.74	192.168.1.94	TCP	66 60926 -> 2404 [ACK] Seq=101 Ack=1986 Win=64256
224	38.571677596	192.168.1.94	192.168.1.74	104asdu	84 -> I (109,1) ASDU=1 M_ME_NB_1 Per/Cyc IOA=116
225	38.571713663	192.168.1.74	192.168.1.94	TCP	66 60926 -> 2404 [ACK] Seq=101 Ack=2004 Win=64256
226	38.620896133				42 <Ignored>
227	38.621403175				60 <Ignored>
228	38.656127439	192.168.1.74	192.168.1.94	TCP	54 60926 -> 2404 [RST] Seq=101 Win=65536 Len=0
229	38.663864984				82 <Ignored>
230	38.663956934	192.168.1.94	192.168.1.74	TCP	60 2404 -> 60926 [RST] Seq=2004 Win=0 Len=0
231	38.692952936	192.168.1.74	192.168.1.94	TCP	54 60926 -> 2404 [RST] Seq=357 Win=65536 Len=0
232	38.740701822	192.168.1.74	192.168.1.94	TCP	54 60926 -> 2404 [RST] Seq=613 Win=65536 Len=0

Figure 5.6: Breaking connection between client and server

At this time, the connection is already down between the client and the server. The next step is to establish a new TCP connection between the attacker and the server. In Figure 5.7 is the attacker IP=192.168.1.96 establishing a new connection with the server IP=192.168.1.94. Also, in order to start communicating with IEC 60870-5-104 the server needs to receive the STARTDT act message to enable data transfer between attacker and server as shown in Figure 5.7. Also, it is possible to see the server replying with the "STARTDT con" message meaning that they can start communicating IEC 60870-5-104.

250	40.021067902	IntelCor_96:...	LLDP_Multicast	LLDP	136 TTL = 20 SysName = MD21
251	42.347140537	192.168.1.96	192.168.1.94	TCP	74 57476 -> 2404 [SYN] Seq=
252	42.347594007	192.168.1.94	192.168.1.96	TCP	74 2404 -> 57476 [SYN, ACK]
253	42.347629653	192.168.1.96	192.168.1.94	TCP	66 57476 -> 2404 [ACK] Seq=
254	42.347997492	192.168.1.96	192.168.1.94	104apci	72 <- U (STARTDT act)
255	42.348718791	192.168.1.94	192.168.1.96	TCP	66 2404 -> 57476 [ACK] Seq=
256	42.354750643	192.168.1.94	192.168.1.96	104apci	72 -> U (STARTDT con)
257	42.354769213	192.168.1.96	192.168.1.94	TCP	66 57476 -> 2404 [ACK] Seq=
258	42.354802200	192.168.1.94	192.168.1.96	104asdu	84 -> I (0,0) ASDU=1 M_ME

Figure 5.7: New malicious connection

With the server already enabled for data transfer, i.e. accepting IEC 60870-5-104 commands, we can start building the payload for the command injection. Choosing the

command, when the attacker was analysing the messages between client and server, it was possible to analyse what the client was writing to the server. This information was value 0 to the IOA (information object address) 5000 writing 0 to the variable 5000. So, for this experiment, we decided to send 1 to the IOA 5000. In Figure 5.8 is the attacker sending the packet.

272	46.352853445	192.168.1.96	192.168.1.94	104asdu	82 <- I (0,6)	ASDU=49665 C_SC_NA_1 Act	IOA=5000
273	46.353099759	192.168.1.94	192.168.1.96	TCP	66 2404 → 57476	[ACK] Seq=133 Ack=23 Win=65280 Len=0	
274	46.353227738	192.168.1.94	192.168.1.96	104asdu	82 -> I (7,1)	ASDU=49665 C_SC_NA_1 ActCon	IOA=5000

Figure 5.8: Command Injection

In Figure 5.9 is presented the server-side. The first part shows a connection that was terminated with IP=192.168.1.74 without even sending any command, and a new connection with IP=192.168.1.94, the attacker, that managed to send a valid command.

```

./simple_server
File Edit View Search Terminal Help
root@kali: ~/Documents/proto/iec/lib60870/li
104 server master ./simple_server
APCI parameters:
t0: 10
t1: 15
t2: 10
t3: 20
k: 12
w: 8
New connection request from 192.168.1.74
Connection opened (0x55f7daafd3d8)
Connection activated (0x55f7daafd3d8)
Received interrogation for group 20
Connection closed (0x55f7daafd3d8)
New connection request from 192.168.1.96
Connection opened (0x55f7daafd3d8)
Connection activated (0x55f7daafd3d8)
received single command
IOA: 5000 switch to 1
Connection closed (0x55f7daafd3d8)

```

Figure 5.9: Command injection server side

In summary not only it was possible to inject the attack using PICSEL but it was also possible to observe the effects of this attack namely sending an incorrect value that could have an impact on a production system. An overall discussion of the results is presented in Section 5.4.

5.3.2 MITM Attack on IEC 61850

A MITM attack is a very common attack. In this example, we will exemplify an attack to the protocol IEC 61850. The objective of this experiment is to evaluate a PICSEL goal, namely the possibility of injecting an attack and observing its impacts on a production line.

For this experiment, the precondition is that the attacker already had gained access to the SCADA system, level 1 of the network. Figure 5.10 represents the location of the attacker for the experiment.

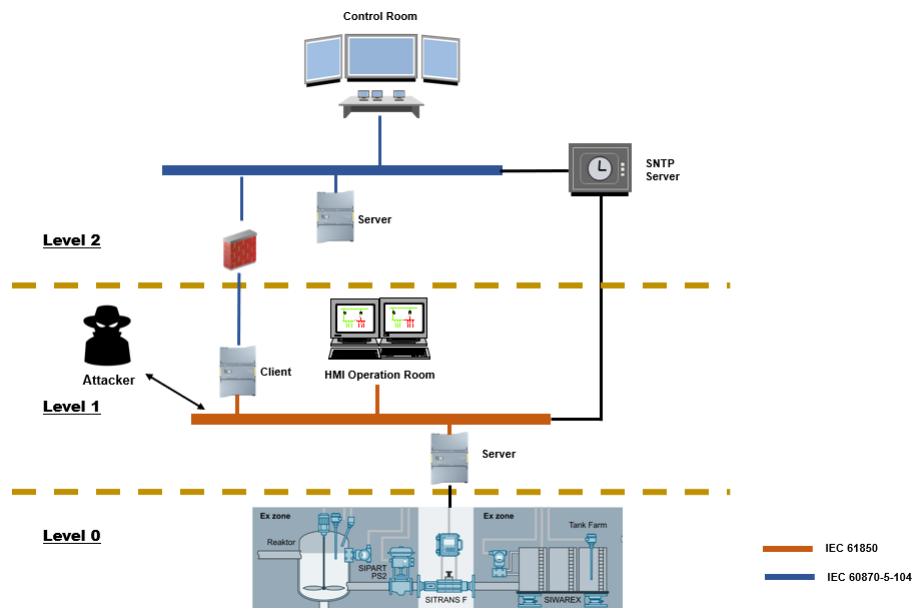


Figure 5.10: MITM Attacker

The attacker always needs to perform some type of information gathering, identify targets, communications, messages. After identifying the potential targets it is possible to launch the attack, in this case, the MITM. To do that, in this example was used as a tool called Ettercap.

This tool allows us to perform protocol analysis, traffic interception, active and passive eavesdropping. Having all these features, configuring and launching the attack is the only thing left.

First, we need to add the targets, in this case, the client and server. Meanwhile, with the targets added we are only doing passive listening, to start the MITM the attacker must launch the ARP poisoning. This will send false ARP messages through the network in order to associate the attacker MAC address to another machine. Now that the attacker is between the communication, we can start to do some real damage. Being in the middle of the connection we can do a variety of things like redirect traffic and change packets, and both the client and server will be always thinking that everything is normal. For this experiment, we decided to clock some specific messages.

In Figure 5.11 is the log console of Ettercap. Ettercap offers a tool called Etterfilter that allows the programming of filters. Etterfilter allows compiling filters to be used on Ettercap. These filters consist of 'if' and 'if/else' statements, the syntax is almost like C code. To exemplify the possibilities of this tool a filter was written, consisting on searching every packet for a specific destination IP, PORT and string. This string was set

```

kat
er ARP poisoning victims:
GROUP 1 : 192.168.1.74 00:0C:29:96:D5:4A
col
ECO GROUP 2 : 192.168.1.94 00:0C:29:39:46:7E
ant Client server IEC communication!!!!
RRP Client server IEC communication!!!!
sou Command detected!!!
g t Dropping packet!
ng Client server IEC communication!!!!
labels to real offsets done. Dropping packet!

```

Figure 5.11: Ettercap console

after analysing the communication and identifying the packet that, in this case, sets the circuit breaker to 1 or 0 and finally drops that specific packet. In other words, this filter checks if the packet meets a specific conditions searching inside the packet for a specific string and if the packet meets all conditions it drops the packet.

```

, -0.996165
Circuit breaker state = 1
Valores: 0.946300
, 0.239249
, -0.687766
, -0.982453
Valores: 0.909297
, 0.141120
, -0.756802
, -0.958924

```

Figure 5.12: MITM server side

In Figure 5.12 it is possible to see that, after deploying the filter, the message "Circuit breaker state = 1" stops appearing in the latest received values. Although, at the client side presented at Figure 5.13 everything is normal.

With this experiment, it shows how easily an attacker can deploy a MITM attack in this environment. We can observe how an operator could be blocked from sending a command to a remote location, potentially affecting the entire process.

5.3.3 Exploits in the Wild

ProductCERT team is not only in contact with the researches for responsible disclosure of vulnerabilities. There is constant monitoring of the Internet for new exploits and POC.


```

+-----+-----+
| Variables | AnalogValues() |
+-----+-----+
|   AnIn1   |      0.909297   |
|           |                 |
+-----+-----+
|   AnIn2   |      0.141120   |
|           |                 |
+-----+-----+
|   AnIn3   |     -0.756802   |
|           |                 |
+-----+-----+
|   AnIn4   |     -0.958924   |
|           |                 |
+-----+-----+

```

Figure 5.13: Client HMI

Performing some research is possible to encounter a lot of exploits for this kind of systems. This section of the evaluation is intended to show another important role of PICSEL, analyse different types of exploits in the wild.

Some of the most important phases in an attack is to enumerate all possible devices. To give an example, a script from GitHub was tested. In Figure 5.14, is presented a result of that specific enumeration script.

```

PORT      STATE SERVICE  REASON          VERSION
102/tcp   open  iso-tsap syn-ack ttl 30 Siemens S7 PLC
| s7-enumerate:
|   Module: 6ES7 214-1AG31-0XB0  \xFF\xFE
|   Basic Hardware: 6ES7 214-1AG31-0XB0  \xFF\xFE
|_  Version: 3.0.2
MAC Address: 00:1C:06:09:0C:34 (Siemens Numerical Control, Nanjing)
Service Info: Device: specialized
Final times for host: srtd: 745 rtdvar: 3803 to: 10000

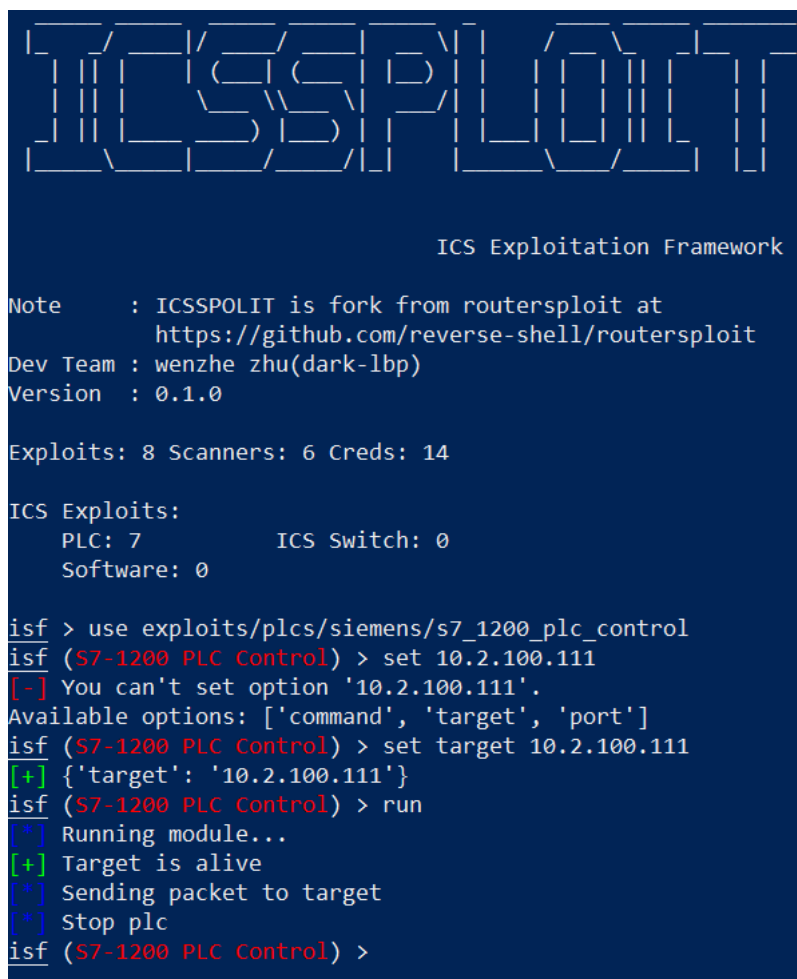
```

Figure 5.14: Result of a enumeration script

It is a Nmap script and it allows to see some relevant information of the device. This relevant information like the "Module" that corresponds to the MLFB and "Version" to the firmware version, it allows an attacker to know the specific module, main functions and if there are some recent vulnerabilities. Having the possibility to quickly test this type of scripts, it is important to flag them and to understand what kind of information an attacker can access.

Another very interesting GitHub was ISF (Industrial Exploitation Framework). This

framework has a variety of tools from scanners to exploit modules and also for different manufactures. In this case, we are going to focus on Siemens related exploits. This framework can detect communication in Profinet and S7comm, change IP addresses in a Profinet communication, S7-1200 PLC control and also for S7-300/400, which are other PLC families. All these exploits depend on the firmware version that each PLC is running. In PICSEL we have an S7-1200 so we are going to test the PLC control that allows us to start/stop/reset the module. To configure the module is pretty simple, it is like in the Metasploit framework. First, we chose the module that we want to run, next it is required to set a target, the port is by default the 102 and we run. Figure 5.15 shows the ISF framework with the module running.



```

ICS Exploitation Framework

Note      : ICSSPOLIT is fork from routersploit at
            https://github.com/reverse-shell/routersploit
Dev Team  : wenzhe zhu(dark-lbp)
Version   : 0.1.0

Exploits: 8 Scanners: 6 Creds: 14

ICS Exploits:
  PLC: 7          ICS Switch: 0
  Software: 0

isf > use exploits/plcs/siemens/s7_1200_plc_control
isf (S7-1200 PLC Control) > set 10.2.100.111
[-] You can't set option '10.2.100.111'.
Available options: ['command', 'target', 'port']
isf (S7-1200 PLC Control) > set target 10.2.100.111
[+] {'target': '10.2.100.111'}
isf (S7-1200 PLC Control) > run
[*] Running module...
[+] Target is alive
[*] Sending packet to target
[*] Stop plc
isf (S7-1200 PLC Control) >

```

Figure 5.15: ISF Framework

And in Figure 5.16 it is possible to see the result of the module.

In the PLC there are 3 lights, in this case, the most important is the "Run/Stop". If the PLC is in "Run" mode the light is green, if it is in stop the light is orange. In this case, after running the module the result is clear. In stop mode, any user program that is programmed on PLC is not running.



Figure 5.16: PLC state

Another set of exploits found was the ACSPLOIT, another GitHub with multiple modules. A particular interesting module was the `segment_smack.py`. Segment Smack (CVE-2018-5390) forces 4.9+ kernel versions of Linux to make expensive calls to some CPU intensive functions for every packet sent. This is achieved by bombarding the target machine with TCP keep-alive packets (PSH; ACK).

In order to test it, we set up the attacker with the 192.168.0.6 IP address, PLC 192.168.0.1 and the HMI 192.168.0.2. In Figure 5.17 is the Wireshark with the `segment_smack.py` running with the target set to the PLC.

No.	Time	Source	Destination	Protocol	Len	Info
829	27.723885591	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
830	27.723888859	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=664 Ack=1 Win=
831	27.724617830	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
832	27.724620638	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=666 Ack=1 Win=
833	27.725282025	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
834	27.725284409	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=668 Ack=1 Win=
835	27.725970681	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
836	27.725974724	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=670 Ack=1 Win=
837	27.726905673	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
838	27.726912370	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=672 Ack=1 Win=
839	27.727627647	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
840	27.727630120	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=674 Ack=1 Win=
841	27.728289166	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
842	27.728291543	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=676 Ack=1 Win=
843	27.729011311	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
844	27.729013785	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=678 Ack=1 Win=
845	27.729700615	192.168.0.6	192.168.0.1	TPKT	60	[TCP Previous segment not captured] Continuation
846	27.729703145	192.168.0.6	192.168.0.1	TCP	60	[TCP Keep-Alive] 20344 - 102 [PSH, ACK] Seq=680 Ack=1 Win=

Figure 5.17: Wireshark SegmentSmack Exploit

Even if the exploit is not directly targeting ICS components, this CVE is affecting some of the Siemens's equipment. With a simple program running, by sending a PLC input 1 or 0 to the HMI when we start running the exploit, the HMI immediately had lost connection from the PLC and instead of showing 0 or 1 at the HMI was "#####".

The ”#” symbol represents a broken connection between HMI and PLC.

When we tested this exploit, Siemens was already aware of the vulnerability, and a security advisory was published at the Siemens ProductCERT website www.siemens.com/cert.

5.3.4 Security Solution

Another goal for the PICSEL project is the possibility to test security controls in an industrial environment. Another project that was developed at the same time was a type of IDS with some SIEM characteristics. Meaning, it can monitor all the network through the mirror port that allows access to all VLAN’s and can collect all information about the PICSEL equipment. Having so much information, this solution, with some integrated tools, can correlate and detect malicious threats inside the network.

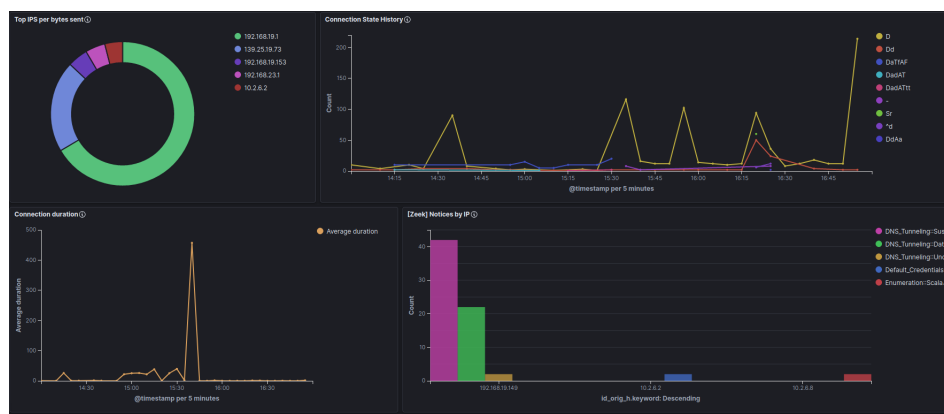


Figure 5.18: Security Solution Dashboard

For this security appliance to be able to display all the information about the network and events that may be important to show, in Figure 5.18 is shown a print screen of the dashboard. In this dashboard, it is possible to organize, filter and create specific rules that can be used to correlate all the information.

5.4 Discussion

In this section, we are going to discuss the results of the experimental evaluation and verify if the PICSEL goals were met.

The command injection attack to the IEC 60870-5-104 protocol was based on a real experiment. This vulnerability comes in the protocol that allows for a server to have multiple clients. The experiment, although we were using a limited set of real equipment, was a success - although the exploit is not publicly available and there is only limited information about it, we managed to reproduce it. We had broken the communication between client and server and successfully injected a value on the server variable. Also,

another experiment taking advantage of the lack of encryption is the MITM experiment on the IEC 61850 protocol. This experiment shows how easily an attacker can craft a filter to block selected messages from being sent. Both examples show the potentialities of having such a framework to analyse an attacker behaviour in order to try to prevent and circumvent these flaws.

Another important point is to test another type of exploits or enumeration scripts. Having these scripts tested, it gives us a better perception of what is published out there and how they behave with real equipment. Although we only have one S7-1200 and another S7-1500, we were able to test these scripts against these devices and see some interesting behaviour, namely that a significant amount of information can be obtained from the device. The information that an attacker can gather about the devices is very critical and even if the exploits are not directly targeting these industrial devices they can have an impact on them. The segment smack example is very good because it allowed us to quickly test and analyse the potential problems in a real environment. This problem can be easily identified by having a connection between the PLC and HMI broken in a real environment. This means that, although the PLC continues the normal program at level 1, the operator at level 2 doesn't have any information about what is happening and the ability to control the system with the HMI.

Having access to all that information and testing all those exploits, a different project developed at Siemens consisted on a security appliance for an industrial environment. To evaluate this appliance, PICSEL was used. The major benefits of PICSEL, besides recreating an ICS environment, was the information that it provides from control and network equipment.

One of the major goals of that external project was to be able to correlate all the information and to detect anomalies in the system. The appliance developed in that project captures all PICSEL traffic and, based on that, detect anomalies or important security events. Events such as: unknown devices in the network, someone is trying to perform a MITM attack in the network, or a login attempt into the web server with the default credentials is detected. All these events generate a warning and can be visualized at the application dashboard, Figure 5.18 like a SIEM.

Finally, based on all these experiments the potentials of PICSEL are clear. PICSEL allows the implementation of diverse industrial scenarios, on which it is possible to perform security tests. For instance, trying to inject vulnerabilities in a given configuration or trying to perform attacks to see if they are effective and to observe their effects. Furthermore, PICSEL is a small testbed that can be easily moved around to anywhere, hence be taken taken to a conference or a meeting with a client and used to show, with real equipment, how easily an attacker can disrupt a normal industrial process. This will be more effective than using simple presentations, to convince them how important it is to take security more seriously.

Chapter 6

Conclusion

In this thesis, it was intended to create a platform allowing to analyse and recreate different types of ICS environments to test different types of vulnerabilities and gain a better understanding of the impacts on these different environments. The knowledge acquired allowed me to have a completely different idea of how everything works and the problems that may arise without security features on these environments.

Current trends, such as Industry 4.0 and the Internet of Things, are an evolving industry that seeks ubiquity, moving away from the traditional monolithic and self-contained infrastructure paradigm in favor of distributed and interconnected architectures. Sometimes, these new infrastructures due to a variety of factors are implemented upon older ones, bringing an entire set of vulnerabilities that even the newest systems can't prevail.

Taking this into account, the combination of old equipment with these new requirements for critical infrastructures can be challenging due to traditional network architectures and the lack of security features, it may thus become an extremely complex process to be properly secured. Different equipment is provided by different suppliers that rely on closed management protocols and different configuration instructions or interfaces.

In this work, it was created a new solution that uses PICSEL as the primary building block. It was developed an approach based on ICS, taking into consideration the necessities of the current critical infrastructures while trying to keep the main characteristics of a real industry with the available equipment. This approach brings the possibility of having a small scale ICS environment with the ability to perform security tests and assess possible impacts.

The solution implemented allowed to recreate a small scale industry multiple many communication protocols where different types of security functions can be applied for each case. The security tests performed allowed the understanding of possible impacts in a real infrastructure and at the same time satisfying the project requirements.

Firstly, the implementation of the solution started with the assembly of the physical equipment. During the research of potential industries, it was clear that the equipment was not sufficient for all scenarios, so some choices had to be made. Specially on the

energy industry, some specific equipment was required. To circumvent this problem, that environment was deployed with an open-source library, namely MZ Automation project, which is a software version of IEC 61850 and IEC 60870-5-101/104 protocols. Multiple Linux PC's were used to test and evaluate the software version of the protocols.

In the Industry 4.0 project, we were able to recreate all conditions using the PICSEL equipment. This project consisted on the recreation of a simple beverage industry. The project was developed with TIA Portal by following the official documentation, and at the end was deployed to the real equipment. Regarding the network conditions, it was used the PICSEL network equipment in both projects.

Some aspects of the testbed environments were not sufficient. In the case of the energy industry, the results could have been affected because the protocol can behave differently when implemented using real equipment, and so, the experiment evaluations results can be different; on the other hand, having to implement them allowed a better understanding of how they work.

The COVID-19 situation had a negative impact on the possibility of testing the portability requirement for PICSEL. In fact, due to physical and logistics limitations, it was not possible to assemble all the devices in a single frame that would allow the solution to become completely portable as originally planned. Something to be done as soon as these limitations are not an issue any longer.

During the tests, for each of PICSEL's objectives, some potentials were demonstrated: 1) Prove the possibility to perform new exploits and study how protocols and equipment communicate with each other; 2) Show the effectiveness of testing exploits already in the wild and assess if they actually represent a real threat; 3) Test a security appliance showing the potentials of such a testbed. Analysing the results, it was possible to demonstrate that PICSEL is a successful project and met all requirements defined at the beginning by performing multiple tests: 1) Test exploits in the wild and assess their behaviour; 2) Replicate POC attacks to assess possible impacts; 3) Test a security solutions that can detect and prevent the above cases.

Further research can be performed to understand other types of industries and the physical equipment required for each one. Deploying different types of network configurations is really important to analyse other types of attacks (e.g. lateral movement). In this project, only the lower levels of the Purdue model were taken into consideration, in the future, higher levels could also be deployed.

Bibliography

- [1] Industrial network market shares 2019 according to hms. <https://www.hms-networks.com/news-and-insights/2019/05/07/industrial-network-market-shares-2019-according-to-hms>. [Online; accessed Nov 2019].
- [2] A scada test bed for cyber security education & research. *Indian Institute of Technology in Kanpur*.
- [3] Siemens communications overview. http://snap7.sourceforge.net/siemens_comm.html. [Online; accessed Nov 2019].
- [4] What is a plc? <https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/what-plc/>. [Online; accessed Nov 2019].
- [5] Irfan Ahmed, Vassil Roussev, William Johnson, Saranyan Senthivel, and Sneha Sudhakaran. A scada system testbed for cybersecurity and forensic research and pedagogy. In *Proceedings of the 2nd Annual Industrial Control System Security Workshop*, pages 1–9. ACM, 2016.
- [6] Thiago Alves, Rishabh Das, Aaron Werth, and Thomas Morris. Virtualization of scada testbeds for cybersecurity research: A modular approach. *Computers & Security*, 77:531–546, 2018.
- [7] Kenneth Barnes and Briam Johnson. National scada test bed substation automation evaluation report. Technical report, Idaho National Laboratory (INL), 2009.
- [8] Bywebranded. Information technologies (it) vs operational technologies (ot). <https://randed.com/information-technologies-it-vs-operational-technologies-ot/?lang=en>, Mar 2019. [Online; accessed Nov 2019].
- [9] Miguel Collantes and Antonio Padilla. Protocols and network security in ics infrastructures. *incibe*, May 2015.

- [10] Cheng Lei, Li Donghong, and Ma Liang. The spear to break the security wall of s7commplus. *Blackhat USA*, 2017.
- [11] Bernard Marr. What is industry 4.0? here's a super easy explanation for anyone. <https://www.forbes.com/sites/bernardmarr/2018/09/02/what-is-industry-4-0-heres-a-super-easy-explanation-for-anyone/#64b6836a9788>, Jul 2019. [Online; accessed Nov 2019].
- [12] PROFIBUS Nutzerorganisation. Profinet security guideline. *Guideline for PROFINET*, 7.002, Nov 2013.
- [13] Luciana Obregon. Secure architecture for industrial control systems. *SANS Institute Information Security Reading Room*, 2015.
- [14] Maslina Daud Norhamadi Ja'afar Salman Yussof Roslan Ismail Wan Azlan Wan Kamarulzaman Qais Saif Qassim, Norziana Jamil. Simulating command injection attacks on iec 60870-5-104 protocol in scada system. *International Journal of Engineering Technology*, 2018.
- [15] Qais Qassim, Norziana Jamil, Izham Zainal Abidin, Mohd Ezanee Rusli, Salman Yussof, Roslan Ismail, Fairuz Abdullah, Norhamadi Ja'afar, Hafizah Che Hasan, and Maslina Daud. A survey of scada testbed implementation approaches. *Indian Journal of Science and Technology*, 10(26):1–8, 2017.
- [16] Carlos Queiroz, Abdun Mahmood, and Zahir Tari. Scadasim—a framework for building scada simulations. *IEEE Transactions on Smart Grid*, 2(4):589–597, 2011.
- [17] GPH Sandaruwan, PS Ranaweera, and Vladimir A Oleshchuk. Plc security and critical infrastructure protection. In *2013 IEEE 8th International Conference on Industrial and Information Systems*, pages 81–85. IEEE, 2013.
- [18] Wolfgang Schwab and Mathieu Poujol. The state of industrial cybersecurity 2018. *Trend Study Kaspersky Reports*, page 33, 2018.
- [19] Keith Stouffer, Joe Falco, and Karen Scarfone. Guide to industrial control systems (ics) security. *NIST special publication*, 800(82), 2011.
- [20] Tsuyoshi Toyama, Takuya Yoshida, Hisashi Oguma, and Tsutomu Matsumoto. Pasta: Portable automotive security testbed with adaptability.