UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE INFORMÁTICA

Security Monitoring in Production Areas

Ricardo Nuno Silveira Lopes de Aguiar

Mestrado em Segurança Informática

Dissertação orientada por:

Professor Doutor Mário João Barata Calha

2022

Acknowledgments

Sometimes our life must be done outside the order we hold as expected. Our individual consciousness, decisions, and events shape us throughout our existence. And wisdom lies in taking the best opportunities to realize our aspirations. The chance for this master's degree was a result of simply that. A long-time desire that came up at the right time and could hardly be at any other moment.

My most enormous thanks go to my family. First and foremost, to my beloved wife for her encouragement and daily support so that I would have all the conditions to accomplish this strenuous task. Without her, who never doubted me, I would never have had the chance to do it, and as the song goes:

"And so, today my world it smiles,
Your hand in mine, we walk the miles
Thanks to you, it will be done
For you to me are the only one"

Then, to my dear children, for the time I was not with them. May this serve as an example to you of how the goals we set ourselves throughout our lives can be accomplished, regardless of age and effort.
And then, to my parents, just as I ask my children, they have always asked me only two things; "be happy and make us proud".
So, I say to all of you, "One more step accomplished, and you are part of it. Thank you."

To my former manager for his personal and institutional support when I told him that I would pursue a master's degree in information security and would like to accomplish it in the context of the company to which I have dedicated myself professionally for the past 13 years. Only an altruistic person would be able to give his support. There aren't many leaders like that, so it was a privilege to be part of his team.

Finally, to Prof. Mário Calha, who was more than a dissertation advisor, was a motivator. Even when I asked myself if I would be able to carry out my dissertation, he always motivated me to continue moving forward, giving me valuable insights into the challenges of this work.

Resumo

Desde dos finais de anos 60 do século passado que um conjunto separado de tecnologias foram concebidas e implementadas para assistir na automatização dos processos industriais e manufatura. Estes sistemas, criados de forma paralela ás tecnologias IT (Information Technologies), ficaram conhecidos como, tecnologias OT (Operational Technologies).

De forma distinta das tecnologias IT, estas foram desenvolvidas com um conjunto de requisitos diferentes. Com um foco na resiliência sobre condições ambientais adversas – como temperatura, humidade, interferência eletromagnética –, uma necessidade de disponibilidade elevada e uma performance em quase tempo-real, estas tecnologias foram deixando para segundo plano outros requisitos. Como a integridade da informação ou a sua confidencialidade.

Mas a necessidade de automatizar processos, foi aumentando e nos dias de hoje, não são somente as áreas industriais – como a produção metalúrgica pesada, industria do petróleo e gás, redes elétricas, processos de distribuição de água, ou tratamento de esgotos – que tem a necessidade de aumentar a sua eficácia. As áreas de produção de uma empresa manufaturadora, também beneficiam destes dois tipos de tecnologias – IT e OT.

E é no chão-de-fabrica – i.e. numa área de produção – que as duas em encontram e se fundem e que interligam as duas redes de forma a se tornarem num sistema misto. Por vezes os requisitos para o funcionamento para uma tecnologia é o ponto fraco da outra. Um bom exemplo é a cada vez maior necessidade dos dispositivos IT terem de se ligar á Internet. Por outro lado, os dispositivos OT que têm frequentes limitações nos processos de autenticação e autorização, são expostos a redes não confiáveis, como a Internet por definição.

Nos últimos anos, e agravado pela mudanças sociopolíticas no mundo, tem-se verificado incidentes nas áreas industriais e de produção cada vez maiores e mais frequentes. Porque, estes incidentes têm um enorme potencial impacto, empresas e organizações governamentais estão cada vez mais disponíveis para implementar medidas de segurança que as defendam. Para a segurança de informação, este é um terreno fértil para o desenvolvimento de metodologias novas ou experimentação e validação de outras já existentes.

Este trabalho final de mestrado, segue a denominada “Four Step Framework” ilustrada na figura 1 mencionada por Adam Shostack [Shostack, 2014] – que pretende responder ás perguntas abaixo visando aplicar um modelo de ameaças no contexto de uma área de produção, obtendo assim um conjunto das ameaças mais relevantes. Com este ponto de partida, será analisada a aplicabilidade e o valor de duas soluções de monitorização de eventos de segurança para as áreas de produção.

Questões a serem respondidas por um modelo de ameaças:

- O que é que deve ser construído?

- O que pode correr mal, uma vez construído?

- O que deve ser feito em relação às coisas que podem correr mal?

- Foi feito um trabalho de análise decente?

Assim, este trabalho pretende responder às seguintes questões:

1. Com as muitas restrições e particularidades existentes nas áreas de produção e aplicando um modelo de ameaças com as metodologias mais adequadas, quais são os controlos ou medidas mais adequadas a implementar?

2. Um sistema com a tripla funcionalidade de identificar os dispositivos e suas vulnerabilidades, monitorar em tempo real o fluxo de dados na rede e alertar sobre eventos de segurança pode ser uma medida de segurança adequada para mitigar as ameaças identificadas?

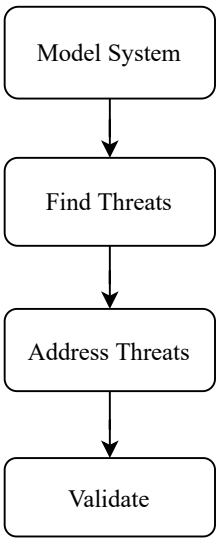


Figura 1: Four-Step Framework

Na primeira parte desta dissertação é realizada uma contextualização do que é uma área de produção, recorrendo a dois exemplos. Posteriormente, utilizando os conhecimentos adquiridos com os artigos, relatórios e literatura específica revistos, foi possível identificar os principais tópicos que têm impacto na segurança das áreas de produção. Com base nisto, foram assinaladas as medidas de segurança mais promissoras mencionadas no estado-da-arte estudado. Adicionalmente foi realizada uma avaliação de impacto da sua implementação, do esforço e do custo, resumidas no quadro 1 abaixo:

Medida de segurança	Melhoria	Esforço Impleme.	Esforço Operação	Custo
Identificação dos ativos	Médio	Baixo	Baixo	Médio
Vulnerabilidade dos dispositivos	Alto	Baixo	Médio	Médio
Segmentação da rede	Alto	Alto	Alto	(Muito) Alto
Proteção contra malware	Médio	Alto	Médio	Alto
Deteção de intrusões (usa. regras)	Médio	Baixo	Baixo	Médio
Deteção de intrusões (usa. anomalias)	Médio	Baixo	Médio	Médio

Tabela 1: Melhoria das medidas de segurança, esforço e avaliação dos custos financeiros.

De seguida, usando um modelo de uma área de produção, concebido com base na observação realizada no decorrer deste trabalho, foi criado um modelo de ameaça usando uma metodologia STRIDE para a identificação e classificação das potenciais ameaças, e a metodologia DREAD para a avaliação de risco.

Este modelo de ameaças tem em consideração três exemplos de máquinas de produção, um servidor com a função de MES (*Manufacturing Execution System*) e duas formas de acesso remoto para realização de assistência e suporte ás máquinas de produção. No diagrama DFD (*Data Flow Diagram*) são tidos em conta os ativos ou *assets*, as comunicações entre si (*data flows*) e as linhas de fronteira (*boundary lines*) que estes atravessam. Com base em três diferentes tipos de adversários – com distintos tipos de acesso – a analise deste diagrama resultou na identificação de 43 ameaças classificadas usando a metodologia STRIDE. E recorrendo á metodologia DREAD foi realizada a avaliação quantitativa do risco.

Usando as dez ameaças com maior risco expostas na tabela 2, desenvolveu-se uma árvore de ataque para mostrar como estas poderão ser encadeadas de forma atingir o objetivo de criar uma disrupção numa área de produção.

ID	Descrição	Risco
T01	Atacantes têm acesso ao sistema ou a dados não autorizados que exploram uma vulnerabilidade conhecida (e.g. falta de *patch*)	48
T02	Atacantes ganham acesso ao sistema explorando características de segurança insuficientes ou mal configuradas	42
T03	Atacantes tentam obter informações pelos *banners* através das portas abertas para descobrir potenciais vulnerabilidades	36
T04	Um adversário pode obter credenciais de autenticação por força-bruta	35
T05	Um adversário adivinha o utilizador e palavra-passe por omissão (e.g. admin:admin)	32
T06	Um adversário pode ligar um dispositivo ilícito para realizar ataques *man-in-the-middle*	30
T07	Os atacantes podem obter acesso a dados sensíveis através de um ataque *man-in-the-middle*	30

ID	Descrição	Risco
T08	Um adversário pode instalar um *malware* ou *backdoor* num dispositivo sem que o utilizador saiba (nenhum anti-vírus)	30
T09	Um adversário pode executar código remotamente	28
T10	Um adversário pode realizar um ataque DoS fazendo sucessivos pedidos de autorização	28

Tabela 2: As dez mais relevantes ameaças

De seguida usando este conjunto de dez ameaças, foi feita a sua correspondência com os tipos de medidas de segurança mais mencionadas pelo estado-da-arte revisto. O qual resultou que a mesma ameaça pode ser mitigada por mais que uma medida. Desta forma, foi feito um estudo para identificar qual a medida que melhor se adequa.

A tabela abaixo resume esta análise:

	Principal medida de mitigação		**Medida adicional de mitigação**	
ID	**Medida**	**Impacto**	**Medida**	**Impacto**
T01	Vulnerabilidade dos dispositivos	Alto	Deteção de intrusões (usa. regras)	Médio
T02	Vulnerabilidade dos dispositivos	Alto	Segmentação da rede	Médio
T03	Deteção de intrusões (usa. anomalias)	Médio	Segmentação da rede	Médio
T04	Deteção de intrusões (usa. regras)	Médio	Segmentação da rede	Médio
T05	Deteção de intrusões (usa. regras)	Médio	Segmentação da rede	Médio
T06	Identificação dos ativos	Baixo		
T07	Deteção de intrusões (usa. regras)	Médio	Segmentação da rede	Médio
T08	Proteção contra malware	Médio		
T09	Deteção de intrusões (usa. anomalias)	Médio	Proteção contra malware	Médio
T10	Deteção de intrusões (usa. regras)	Médio	Segmentação da rede	Médio

Tabela 3: As dez principais ameaças com as medidas de mitigação aplicáveis

A monitorização de eventos de segurança envolve a recolha e análise de informação para detetar comportamentos suspeitos ou alterações não autorizadas do sistema. E a posterior definição de padrões que devem desencadear alertas. Assim como, que ações necessárias tem de ser tomadas para responder a estes alertas.

Analisou-se assim, duas soluções com as funcionalidades de detetar e identificar os dispositivos conectados e as suas vulnerabilidades, e com a monitorização e identificação de eventos de segurança, utilizando o tráfego de rede observado numa área de produção real. Esta análise tem como objetivo verificar o efetivo valor destas ferramentas em mitigar as ameaças mencionadas anteriormente. O resultado é que a implementação de uma ferramenta como esta pode ser uma solução de segurança capaz de mitigar as ameaças de forma eficaz.

No entanto, não deve ser considerada como uma solução que preenche todos os requisitos. Como é demonstrado, não pode oferecer a garantia de identificar todos os dispositivos, nem automatizar completamente o processo de gestão do inventário destes. Além disso, estas ferramentas não podem atribuir a propriedade do quão critico é um dispositivo, tendo em conta todo o sistema. Por todas estas razões, a intervenção humana continua a ser necessária, dissipando assim a ideia de que a tecnologia, por si só, é suficiente para tornar uma área de produção completamente segura.

No final deste trabalho foi possível concluir que apesar das especificidades e restrições de uma área de produção, a realização de um modelo de ameaça utilizando as metodologias STRIDE para a classificação de ameaças e a metodologia DREAD para a sua avaliação de risco produz um conjunto válido. A partir daqui, as medidas ou controlos mais adequados podem ser nomeados. É também possível afirmar que, utilizando soluções semelhantes às analisadas, com a capacidade de identificar os dispositivos e as suas vulnerabilidades e a monitorização em tempo real do fluxo de dados na rede e o alerta para eventos de segurança, é possível mitigar as dez ameaças identificadas.
O resultado é um aumento da maturidade de segurança de uma área de produção, obtendo o correspondente aumento da sua resiliência contra as ameaças constantemente emergentes.

Palavras Chave: modelo de ameaça, vulnerabilidade, identificação, monitorização, área de produção

Abstract

Since the late 1960s, a different set of technologies has been designed and implemented in parallel to assist in automating industrial and manufacturing processes. These systems, created parallel to IT (Information Technologies), became known as OT (Operational Technologies).

Unlike IT technologies, these were developed with a different set of requirements. With a focus on resilience to adverse environmental conditions – such as temperature, humidity, and electromagnetic interference – and a need for high availability and near-real-time performance, these technologies took a back seat to other requirements. Such as information integrity and confidentiality. However, the need to automate processes has developed. Today, it is not only industrial areas – such as heavy manufacturing, oil and gas industries, electrical networks, water distribution processes, or sewage treatment – that need to increase their efficiency. The production areas of a manufacturing company also benefit from these two types of technologies – IT and OT. Furthermore, it is on the shop floor – i.e., in a production area – that the two meet and merge and interconnect the two networks to become a blended system.

Often the requirements for the operation of one technology are the weak point of the other. A good example is an increasing need for IT devices to connect to the Internet. On the other hand, OT devices that often have inherent difficulty with authentication and authorization processes are exposed to untrusted networks.

In recent years, and aggravated by the socio-political changes in the world, incidents in industrial and production areas have become larger and more frequent. As the impact of incidents in these areas has the potential to be immense, companies and government organizations are increasingly willing to implement measures to defend them. For information security, this is fertile ground for developing new methodologies or experimenting and validating existing ones.

This master's work aims to apply a threat model in the context of a production area, thus obtaining a set of the most relevant threats. With the starting point of these threats, the applicability and value of two security monitoring solutions for production areas will be analyzed.

In this dissertation's first part, and after reviewing state-of-the-art with the result of identifying the most mentioned security measures for industrial and manufacturing areas, a contextualization of what a production area will be performed—followed by an example, based on what was observed in the course of this work. After giving this background, a threat model will be created using a STRIDE methodology for identifying and classifying potential threats and using the DREAD methodology for risk assessment. The presentation of an attack tree will show how the identified threats can be linked to achieving the goal of disrupting a production area. After this, a study will be made on which security measures mentioned initially best mitigate the threats identified.

In the final part, the two solutions will be analyzed with the functionalities of detecting connected devices and their vulnerabilities and monitoring and identifying security events using network traffic observed in an actual production area. This observation aims to verify the practical value of these tools in mitigating the threats mentioned above.

During this work, a set of lessons learned were identified, which are presented as recommendations in a separate chapter.

Keywords: threat model, vulnerability, identification, monitoring, production area

Contents

List of Figures

List of Tables

Acronyms

API Application Programming Interface. 79

CISA Cybersecurity and Infrastructure Security Agency. 54, 72

CVSS Common Vulnerability Scoring System. 7, 10, 13

DCS Distributed Control System. 3

DFD Data Flow Diagram. XV, 26, 35, 39, 80, 85

DoS Denial of Service. 44–50, 52, 78

ENISA European Network and Information Security Agency. 5, 8, 9, 72

HMI Human Machine Interface. 3, 13

ICS Industrial Control System. 5, 6, 8, 54, 61, 68

IDS Intrusion Detection System. 6

IEC International Electrotechnical Commission. 7, 72

IEEE Institute of Electrical and Electronics Engineers. 5

IIoT Industrial Internet of Things. 34

IoT Internet of Things. 6

IPC Industrial PC. 35, 36, 39, 43, 53, 54

ISA International Society of Automation. 7, 72

ISMS Information Security management System. 74

ISO International Organization for Standardization. 6, 7, 22, 72

IT Information Technology. 2, 6

M2M Machine to Machine. 37, 56

MES Manufacturing Execution System. V, 33, 34, 36–38, 40, 54–56

NIST National Institute of Standards and Technology. XVIII, 11, 62, 72, 76

OPC-UA Open Platform Communications United Architecture. 37

OT Operational Technology. 2, 3, 6, 62, 68

OWASP Open Web Application Security Project. 21, 26, 27

PLC Programmable Logic Controller. 1, 3, 13, 34–39, 42, 47, 50, 53, 54, 68, 79

SCADA Supervisory Control and Data Acquisition. 3

SDN Software-Defined Networking. 6, 9, 10, 12–14

SIEM Security Information and Event Management. 2, 70, 79

SIS Safety Instrumented System. 7

SOAR Security Orchestration, Automation, and Response. 79

SOC Security Operation Center. 75, 79

TTP Tactics, Techniques and Procedures. 8

VLAN Virtual LAN. XVIII, 65, 66

Chapter 1

Introduction

The production areas of a company have always been regarded with special attention and care. They produce the products with which a company wants to satisfy its customers. Without the production of the products, there is no sale and eventual profit, and therefore no satisfied customers.
The search for efficiency and safety in production processes is reflected in a continuous increase in automation, either with the acquisition of specific machinery, the use of robotics, or, more recently, with the widespread introduction of sensors throughout the factory floor to obtain realtime information on the status of the entire production process.

Although automation began with the industrial revolution in the mid-19th century, it was not until the late 1960s[Barkalov et al., 2019] with the implementation of the first programmable logic controllers (PLCs) that the progressive digitization of manufacturing processes took place. More recently, in 2011, as part of a more comprehensive strategy, ”Industry 4.0” generalized computing on the shop floor, introducing machine-to-machine communication and broad use of embedded sensors that exchange information with other devices over the internet. These new trends have increased diversity, with controllers of higher processing and communication capacity, robots, or sensors directly affecting physical processes. This more significant number and diversity of devices have increased the volume of information generated and exchanged, often using proprietary protocols, thus allowing a permanent and detailed knowledge of the state of the whole system. All this information aims at more effective and flexible management, allowing the implementation of production strategies called ”Just-In-Time”.

The specific nature of these types of equipment is reflected in a high acquisition cost, which intends to make them as profitable as possible through their operation for long periods – often measured in decades, with low maintenance levels or updating. Supported by rigid contracts, the consequent maintenance, and guarantee services heavily depend on the equipment manufacturers, leaving few options to the buyer if later security measures are needed. Thus, a simultaneously diverse and specialized set of various costly and vendor-dependent subsystems, where there is a coexistence of technologies, where the new works alongside the old, where standard-based technologies work alongside proprietary ones, and where the integrity or confidentiality of the data used, is preempted by availability. Another distinctive aspect of these areas is their focus on manufacturing processes, leaving awareness of information management security processes in the background. The engineers responsible for designing these environments seek

to maximize availability, ensure that systems keep running, and avoid interruptions or unexpected stoppages.
It is then essential to reflect on the challenges these environments bring for information security. This information has to translate into efficient operation, which often contains the intellectual property of the manufacturing processes that must be protected.
Perhaps the most disruptive factor for security in production areas is the need for remote access dedicated to equipment maintenance, either by other company areas or, more worryingly, by their respective manufacturers. These accesses are inseparable from the acquisition and are often considered an equipment characteristics. Therefore, they cannot be changed or deactivated when explicitly mentioned, which may often be active through imperceptible accesses. Because they are entirely independent of the preexisting infrastructure, using, for example, mobile broadband network access.

Those responsible for the shopfloor security are faced with the limitations of not being able to interfere directly with the devices, such as installing software agents or implementing security strategies and measures that eliminate remote access or completely block data flow at the network boundaries.
What measures can then be implemented directly on the local network infrastructure to improve the identification of devices and their known vulnerabilities or that bring visibility to the patterns of network traffic generated and the consequent identification of possible attacks on the systems implemented in the production areas?
Only a detailed and comprehensive knowledge of what a production area represents allows us to have a holistic view of the security threats they have to withstand. It is more commonly referred to as OT (Operational technology) instead of the more well-known and studied IT (Information technology).

This work aims to start by studying the structure of a threat model that uses the best methodologies for this context. This threat model will allow other companies or organizations to reuse it as a solid starting point for their specific case. Based on the result of the model mentioned above and as one of the possible resulting measures, the validity of the applicability of a system with the threefold functionality of identifying equipment and other connected elements, their security vulnerabilities, along with a realtime monitoring of the network traffic and the consequent creation of alerts for security events. Like any tool used in an organization, it is more valuable the better it is integrated with other existing tools and processes.Therefore, it is necessary to look at the potential for integration, with a SIEM, for example, or the export of identified assets to a ‘ Configuration Management Data Base”.

This dissertation was based on the production areas of the German group company Carl Zeiss AG. Founded in Jenna in 1846, this world-leading optical systems company currently has approximately 40 production sites worldwide, from Edinburgh in the north to Randburg in South Africa and from Tijuana in Mexico to Tokyo in Japan. Although most known for consumer products such as camera lenses or binoculars, its product portfolio is much more complete today. From optical, X-ray, and scanning electron microscopes, semiconductor lithography systems, ophthalmic medical solutions, eyeglass lenses, and much more. Despite differences in their production areas due to different manufacturing processes and degrees of automation, they all share how the various devices have to be connected.

The chapter 4 on the validation of security measures, where two security monitoring solutions for production areas are validated, was carried out in a production area of the SMT[1] business group in Oberkochen. This group is dedicated to developing state-of-the-art optical systems integrated into the world-leading ASML, the sole supplier to major processor manufacturers such as TSMC, Samsung, and Intel. The results of this work, being the consequence of the analysis of these solutions in the context of the production area where it will be carried out, allow the establishment of this concept's validity and effectiveness, not only in this area but also in any other production area.

1.1 Motivation

The most conventional industrial areas, such as electricity or power and water supply, are designed around a SCADA system composed of a reduced set of devices such as PLC, DCS, or HMI. More commonly referred to as OT technologies instead of the more well-known and studied IT technologies. Nevertheless, the reality is that the overwhelming majority of production or manufacturing areas in companies present a mixture of both worlds, with OT technologies side by side with IT technologies, accumulating and combining the possible security threats. These areas can thus be called "production areas".
As explained in the introduction, in these networks formed by devices with low processing capacity, difficult updating, and often obsolete, the application of measures to reduce or prevent security risks directly on them, can already be envisioned as very difficult or even impractical. Therefore, the option foreseen is to apply the local network-level measures without ever interfering with the necessary network flows between them.

1.2 Objectives

A threat model is the most commonly used methodical framework for conducting a threat survey and proposing security controls to mitigate or prevent the associated risks. Although this structure applies to various systems, it is mainly used in software development contexts. In this context, the concept of "security risk management" is a fundamental element in developing the business model. In this way, this work intends to answer the following questions:

1. With the many restrictions and particularities existing in the production areas and applying a threat model with the most appropriate methodologies, what are the most adequate controls or measures to implement?

2. Can a system with the threefold functionality of identifying the devices and their vulnerabilities, real-time monitoring of data flow on the network, and alerting to security events be an appropriate security measure to mitigate the threats identified?

[1] Semiconductor Manufacturing Technology

1.3 Methodology

To answer the questions above, the methodology followed will start by reviewing and summarizing the state-of-the-art papers, reports, and published articles related to the topic under study. Next, a threat model will be made to determine the adequate measures for the identified threats, answering the first question. Finally, two security events monitoring solutions will be analyzed to validate their effectiveness in mitigating the identified threats to answer the second question.

1.4 Document Structure

In additionally to the present introductory chapter, the remainder of this document is structured in four chapters as follows:

- **Chapter 2** – Related Work – State-of-the-art analysis through reports and papers relevant to the context of this work in the last five years. At the end of this chapter, the most promising security measures recommended by the reading will be identified.
- **Chapter 3** – Threat Modeling – A threat model will be executed, starting by explaining the steps and objectives for its realization. Using as an example a production area that results from the observation made throughout the study for this work. Ending in a correspondence between the threats identified in the threat model and the security measures mentioned in chapter 2.
- **Chapter 4** – Security Monitoring – Two monitoring solutions will be used to validate that they are a measure that can mitigate the threats identified above.
- **Chapter 5** – Recommendation – Discuss further actions that can be taken and the benefit learned during the execution of this work.
- **Chapter 6** – Conclusion – The conclusions from this work will be discussed, as well as possibilities for future work.

Chapter 2

Related Work

This chapter will review the most relevant articles published in the last five years related to security in production areas or industrial environments. This review serves to create a solid knowledge base to support the rest of the work.

Articles from various sources, such as IEEE[1], ENISA[2], and multiple academic institutions, were chosen to have a representative set. In discussing the topic of information security, these articles present us with relevant concepts that are important to define. The first of these concepts, and perhaps the cornerstone, is that of vulnerability, defined by ISO 27005 [for Standardization, 2018b] as *"A weakness of an asset or group of assets that can be exploited by one or more threats (...)."* Moreover, a threat is *"a person or thing likely to cause damage or danger."* These definitions are directly related to *attack surface*, a system set of interaction points where a threat can be realized through a vulnerability.

Threat model is another key concept for this work. Although Adam Shostack [Shostack, 2014] avoids defining it clearly, it can be seen as a process of identifying and dealing with possible threats that can cause damage to a system.

2.1 Topic group articles

The articles reviewed have been grouped by the topics most relevant to this paper to help better understand them and summarize the information and knowledge they provide.

2.1.1 Threat Modeling

Mashkina I. and Garipov I. [Mashkina and Garipov, 2018] briefly focus on the attack vectors targeting an ICS (Industrial Control System). The most exciting part is how they model these threats through a cognitive map (as the authors call it), where it is possible to visualize the path an attacker would have to follow to gain access to the various components of the system. This cognitive map clearly shows a big

[1] Institute of Electrical and Electronics Engineers
[2] European Network and Information Security Agency

difference in the risk of an attack being carried out by an adversary with access to the LAN network or access to the ICS systems (insider threat) of another with remote access. However, how the cascading effect can affect the various components is unclear.

Nweke L and Wolthusen S [Nweke and Wolthusen, 2020] have reviewed the state-of-the-art asset-centric threat modeling approaches. Pointing out that DREAD, Trike, OCTAVE, and PASTA are the most widely used asset-centric threat modeling approaches. Also, describe the features of the asset-centric threat modeling approaches to discuss their similarities and differences.

Tatam M, Shanmugam Bm, Azam S, et al. [Tatam et al., 2021], review the limitations of different threat modeling methodologies, strengths, and any perceived gaps. In this paper, the authors emphasize an asset-focused approach (referred to as asset-centric). These asset-centric approaches can be used to model both technical and non-technical threats and provide explicit information that can be used for assessing risk.

Peter Danielis, Moritz Beckmann, Jan Skodzik [Danielis et al., 2020] present an Excel and VBA-based tool that uses the STRIDE methodology aligned with ISO 27001 and ISO 27005. The latter is for risk management. Nevertheless, it is impossible to validate the tool because it is unavailable. Another shortcoming is that the article analyses IoT device use-case without further context.

Makhdoom I Abolhasan M Lipman J et al. [Makhdoom et al., 2019] offer a comprehensive article listing the vulnerabilities and measures to reduce threat occurrence probability. Not all mentioned measures can be applied to an industrial or production area. There is no comparison of the most effective measures. Another shortcoming is that the whole article analyses IoT always outside the context of a production area. Moreover, always as an autonomous system and using a separate network infrastructure. Nevertheless, measures like IDS and SDN networks are indicated as possible measures that can be used in a production area.

Zografopoulos I, Ospina J, Liu X, et al. [Zografopoulos et al., 2021] propose to develop a framework that bridges theoretical and simulation-based security case studies and evaluates cyber-physical systems behavior leveraging testbed environments, leading to more secure cyber-physical electric systems architectures. Although this article uses an electrical distribution network as a case study, it highlights some exciting ideas characteristic of the OT environment, which has strong similarities to production areas. For example, the reference is that this type of system is much more complex and simultaneously with specific device types compared to the more traditional IT systems. Moreover, an adversary needs a certain level of access to the system – which the author names Adversary Model Formulation – to gain the specific knowledge to carry out a successful attack.

Tuma K, Sandberg C, Thorsson U, et al. [Tuma et al., 2021] investigate the benefits and shortcomings of performing a threat model where a threat risk assessment is completed a priori (eSTRIDE), compared with the traditional STRIDE methodology where the evaluation is done at a later step. Although the study did not show a difference in productivity or timeline, the eSTRIDE methodology identified twice as many threats of higher priority. In contrast, STRIDE identified more threats of medium or low priority. An important aspect also identified was that the information security experience of the teams had a significant

impact.

Khan R, McLaughlin K, Laverty D, et al. [Khan et al., 2017], in this 2017 paper, present an example of using a threat model using STRIDE methodology. Not for software design but for a system part of an electrical distribution network. Although this work is a bit old, it can offer an example of how to model threats in an industrial context.

Xiong W, Lagerström R, et al. [Xiong and Lagerström, 2019], review the available literature until 2019 on threat modeling. However, all articles concerning industry or CPS (cyber-physical systems) were written before 2016. The primary conclusion is that threat modeling is a diverse field lacking common ground, and the definitions are numerous and used in many different ways. Also, the threat modeling work remains to be done manually, which can be time-consuming and error-prone. Moreover, threat modeling is flexible (graphical, formal, qualitative, quantitative), sometimes focusing on general and other times more specific (both in terms of threats and application domain), and validation methods vary.

Hollerer S, Kastner W, and Sauter T [Hollerer et al., 2021] in this short article, discuss the need to perform a threat model to cover the security and safety of a production environment. Arguing that if the SIS (Safety Instrumental System) system is the target of an attack, it can jeopardize the welfare of the people working on the factory floor and the availability of other connected devices. The most important aspect of this article is to relate the CVSS score of vulnerabilities to the Security Layers mentioned in the IEC/ISA62443-3-2 standard.

Messe N, Chiprianov V, et al. [Messe et al., 2020] describe an asset identification process to help participants collaboratively identify significant assets for business stakeholders, product team members, and security experts. This article presents the compelling idea of the definition of an asset. The authors extend the definition (for example, ISO 27001 *"Something that has value to the organization"*) to something more concrete. They have different interpretations for Domain experts or Security experts, part of the threat modeling process. As the authors mention, to a Domain expert, an asset is *"anything that has value for them, towards the fulfillment of the function and goal of the system, together with the assurance of its properties,"* and to a Security expert is *"Anything that has value for them. It has vulnerabilities that can be menaced by threats"*. However, this article lacks a more concrete example to materialize the described concepts.

Reading these articles, it becomes clear that a threat model is a process that allows identifying and effectively dealing with them. Furthermore, there are several possible methodologies to perform them. Thus the most promising approach, which will be used in this paper, is the STRIDE methodology to help identify threats and DREAD for risk assessment.

2.1.2 Threat Detection

Moustafa N., Turnbull B., et al. [Moustafa et al., 2018] propose an interesting idea to improve IDS beyond their existing signature based or anomaly detection models. By proposing a new threat intelligence scheme based on Beta Mixture and Hidden Markov Models (MHMM) for discovering adversaries that

attempt to expose physical and network layers of Industry 4.0 systems. However, as the authors also point out, furthermore work is needed to test this in a real industrial environment.

Bhamare D et al. [Bhamare et al., 2020] review the work produced on cybersecurity for ICS. They highlight that the use of Machine Learning algorithms is not applied on a larger scale in IDS systems due to the lack of a data set based on actual traffic that guarantees adequate accuracy. They have demonstrated that machine learning techniques need significant rework to perform satisfactorily in the context of anomaly detection in ICSs. The major challenge in applying machine learning methods is obtaining real-time and unbiased datasets.
After reading these articles, it is clear that the machine learning algorithms used for identifying ongoing threats, while promising, are not yet in the mainstream.

2.1.3 Supply Chain Threats

The ENISA Threat landscape for supply chain attacks [Lella et al., 2021] report aims to map and study the supply chain attacks discovered from January 2020 to early July 2021. It offers a practical guide with measures to be taken by organizations to increase the resilience of their infrastructure and minimize the impact of an eventual attack through a supplier. It also attempts to define a taxonomy for this type of attack, which is intended to complement Mitre Att&ck's taxonomy.

Yeboah-Ofori et al. [Yeboah-Ofori and Islam, 2019] present a cybersecurity threat model with integrated threat intelligence concepts for the supply chain, such as threat, attack vector, TTP, and control. Plus, with concepts from the goal modeling languages, including actor, goal, and requirement from supply chain context, including inbound and outbound, also they consider widely used industry procedures such as the internet security control and STIX threat model to analyze the threats in the supply chain context. Finally, they used a running example from a smart grid system to study the proposed approach and demonstrate its applicability. Interestingly, one of the proposed controls is an active discovery tool to identify devices connected to the network and update the hardware asset inventory.

These articles reveal a recent concern for supply chain security and strongly encourage all organizations to take steps to identify and control these threats – e.g., through a threat model.

2.1.4 Risk Management

Kure H et al. [Kure et al., 2018] show in this article a comprehensive, integrated cybersecurity risk management framework that explicitly evaluates risk from a holistic viewpoint of the stakeholder model, cross functions risks, and existing risk management frameworks, plus the integration of the cascading effect from interdependent CPS components considering vulnerability, threats, and risks to an asset; and an evaluation of the proposed integrated risk management approach into a real cyber-physical system. Although exhaustive, this is a complicated process that, as the article uses, is more suitable for critical infrastructures such as an electricity distribution network. Also, there is a need to use the described approach in other case studies to generalize the findings and validate the applicability.

This article reinforces the idea that risk management, with its identification, assessment, and treatment of risks, should be used by all organizations as a structural tool to underpin their information security strategy. This very idea is in itself a motivation for this final work.

2.1.5 Industry 4.0

The ENISA Industry 4.0 Cybersecurity Challenges & Recommendations [Malatras et al., 2019] report is not a technical article but provides an excellent overview of the security challenges in the production/manufacturing areas. The audience of this report is much broader, covering the organizations themself, security experts and standardization community, regulators, academia, and R&D institutions. Although the paper has Industry 4.0 in mind, all the mentioned challenges and recommendations apply to all current production areas that have integrated new devices into an existing environment.

This report draws attention to the notion that integrating new devices with the capacity and needs to exchange large amounts of data, often over the Internet, presents significant security challenges to production areas with their devices deployed over the past decades. This notion will have to be considered in identifying threats later in this dissertation.

2.1.6 ICS Survey

Asghar M. et al. [Asghar et al., 2019] present an extensive review of other published papers regarding IDS solutions. Interestingly, according to the authors, all the solutions analyzed have high implementation and maintenance costs. They do not point out any solution as more viable from the financial point of view. However, it highlights in the ”Future research direction” an exciting path where the IDS can provide information to a software-defined network (SDN) to micro-segment the network to isolate the devices, thus minimizing the impact of one compromised device, compromising others.

In this work, the implementation and maintenance costs must be considered when evaluating the measures to be mitigated.

2.2 Security measures presented

Reviewing the literature specific to industrial areas or production environments where the implementation of cyber-physical devices is used, allows the identification of a common set of security solutions to mitigate the risks associated with threats. The following security measures are recurrently mentioned in the reviewed articles:

- Asset identification – solution that collects information from networked assets in an automatic or assisted manner providing a complete view of the various subsystems that make up a production area. Information such as operating system, version, or network configuration is fundamental to knowing the scope of other consequent security measures.

- Device vulnerability and risk evaluation – solution that identifies existing vulnerabilities in the various endpoints connected to the network. Since these devices in production environments are particularly sensitive, this solution should infer the existing vulnerabilities from the observed network traffic. Furthermore, it should also perform risk analysis using the information of the vulnerabilities identified, using, for example, the CVSS score to prioritize the intervention needed to fix the vulnerabilities, e.g., patch installation or configuration change.

- Network segmentation – Segmenting the network into subnets to minimize the scope of a possible threat. For this measure to be effective, traffic control and limitation must be performed through a firewall or access policies using Software Defined Networks (SDN).

- Malware Endpoint Protection – Solution for protecting devices from malware. Also known more commonly as Anti-Malware or Anti-Virus.

- Intrusion Detection (static rules) – Network traffic monitoring solution for intrusion detection, using a defined set of rules, i.e., static rules.

- Intrusion Detection (anomaly-based) – Network traffic monitoring solution for intrusion detection, using a comparative process for identifying intrusions between a regular traffic pattern – i.e., baseline – and observed abnormal traffic.

The articles mentioning each solution are shown in table 2.1.

Security measure	**Articles**
Asset identification	[Malatras et al., 2019], [Yeboah-Ofori and Islam, 2019], [Lella et al., 2021]
Device vulnerability and risk evaluation	[Zografopoulos et al., 2021], [Bhamare et al., 2020], [Asghar et al., 2019], [Lella et al., 2021]
Network segmentation	[Makhdoom et al., 2019], [Yeboah-Ofori and Islam, 2019], [Zografopoulos et al., 2021], [Bhamare et al., 2020], [Asghar et al., 2019]
Malware endpoint protection	[Mashkina and Garipov, 2018], [Tatam et al., 2021], [Makhdoom et al., 2019], [Asghar et al., 2019], [Lella et al., 2021]
Intrusion Detection (static rules)	[Mashkina and Garipov, 2018], [Moustafa et al., 2018], [Makhdoom et al., 2019], [Yeboah-Ofori and Islam, 2019], [Zografopoulos et al., 2021], [Bhamare et al., 2020], [Asghar et al., 2019]
Intrusion Detection (anomaly-based)	[Moustafa et al., 2018], [Kure et al., 2018], [Tatam et al., 2021], [Zografopoulos et al., 2021], [Bhamare et al., 2020], [Asghar et al., 2019]

Table 2.1: Security measures mention in the reviewed articles.

These measures can be further grouped using one of the five core functions designated in the NIST CyberSecurity Framework 1.1. That is, by the phase at which the benefit of their implementation is maximum.
The relevant functions are:

- Identify – Develop an organizational understanding to manage cybersecurity risk to systems, people, assets, data, and capabilities. Examples of outcome Categories within this Function include asset management, business environment, governance, risk assessment, and risk management strategy.

- Protect – Develop and implement appropriate safeguards to ensure delivery of critical services. Examples of outcome Categories within this Function include identity management and access control, awareness and training, data security, information protection Processes and Procedures, Maintenance, and Protective Technology.

- Detect – Develop and implement appropriate activities to identify the occurrence of a cybersecurity event. Examples of outcome Categories within this Function include Anomalies and Events, Security Continuous Monitoring, and Detection Processes.

Appendix B has the NIST CyberSecurity complete Framework Core functions list and description.
It is also helpful to identify the domain of action of each of the security measures.

- Network – the measure is applied in the network context. This measure has the advantage of having a more comprehensive scope since it will impact all connected devices and not require the installation of agents on endpoints. The disadvantage may be less granularity in applying the security measure since it may be missing information only available on the device.

- Endpoint – the measure is applied in the context of each connected device. This measure has the advantage of using information from each device and increasing the security measure's effectiveness. However, this can be seen as a disadvantage in the production area context due to the frequent inability to install agents on endpoints. Installing extra software can have an unpredictable impact on the devices and their functionality or operation behavior due to the often limited processing power.

For a more straightforward overview, Table 2.2 summarizes the function and domain of each of the solutions.
After enumerating the security solutions reference in the articles reviewed, each correspondence with the phases of the NIST Cyber Security Framework, and finally, with the domain of action, an assessment of applicability in production areas is required. For this analysis, the following dimensions must be taken into account:

Phase	Security measure	Domain	Description
Identify	Asset identification	Endpoint or Network	Assisted/Automated asset identification of network connected devices
Identify	Device vulnerability and risk evaluation	Endpoint or Network	Assisted/Automated asset vulnerabilities and risk evaluation network connected devices
Protect	Network segmentation	Network	Use SDN to perform network (micro) segmentation to limit the devices accessibility only other required devices
Protect	Malware endpoint protection	Endpoint	Endpoint security protection software (aka. Anti-Virus or Anti-Malware)
Detect	Intrusion Detection (static rules)	Network	Intrusion Detection System based on rules
Detect	Intrusion Detection (anomaly-based)	Network	Intrusion Detection System based on anomaly detection

Table 2.2: Security measures classified by application phase and domain.

- Security Improvement – expected improvement that the security solution application will have in increasing the resilience of the production area.
- Implementation effort – expected effort to implement the security solution in a production area expected.
- Operation effort – expected effort to operate and maintain the security solution ensuring its continued effectiveness.
- Financial cost – expected financial cost required to acquire the security solution.

For this analysis, qualitative values were used to evaluate each dimension. Since a more detailed and quantitative set of values is impossible, a qualitative measure was chosen in three grades; Low, Medium, and High. These three measures simultaneously allow for subjectivity in the evaluation while maintaining the detail required to differentiate them and support an adoption security solution decision.
It is now possible to evaluate each of the security solutions using the previously mentioned dimensions:

- Asset identification
  - Security Improvement (Medium) – the knowledge of the existing assets in a production area is essential to choose and increasing the effectiveness of any other security measure. However, it is not in itself a direct threat mitigation action.

  - Implementation effort (Low) – implementation can be done in the network domain or at the endpoint. However, in a production context, installing any software or agent is impractical since this could directly impact the operation of the device or not be available to the wide variety of existing device types. That is why the option of network usage is the most feasible and easiest to implement.
  - Operation effort (Low) – the operation of this type of security solution requires infrequent configuration changes or updates. Most information collected should be automatic if a new device is identified. It can eventually be enriched with information such as the function or device criticality in the context of the production area in which it is inserted.
  - Financial cost (Medium) – the cost may be considerable because this type of solution has to fit into the context of a production area. Unusual device types characterize these environments – e.g., PLC, HMI, automatic guided vehicle – that communicate using proprietary protocols, e.g., S7comm, Profinet, or Modbus.

- Device vulnerability and risk evaluation

  - Security Improvement (High) – identifying the vulnerabilities in each device is of evident importance for increasing the whole system's security. Moreover, a risk assessment according to the degree of vulnerability as mentioned before using, for example, a CVSS score allows for defining a strategy to prioritize further security measures, e.g., upgrading, reconfiguring, or isolating the network connection.
  - Implementation effort (Low) – as with the previous solution, it is impossible to use agent installation in a production environment. Therefore, identifying vulnerabilities through the network is the best option.
  - Operation effort (Medium) – for the information on the vulnerabilities of the devices to be valid, there must be a frequent updating of the database of known vulnerabilities and a frequent reevaluation of the existing risk.
  - Financial cost (Medium) – the need to cover specific systems reduces the number of existing solutions, thus increasing the cost.

- Network segmentation

  - Security Improvement (High) – the segmentation of the network into smaller parts, thus limiting the scope of a possible threat or attack, is highly important to increase the resilience of any system in general and the production area in particular.
  - Implementation effort (High) – implementation is of great difficulty because it involves reconfiguring the network configurations of connected devices. A production area in operation requires a stop for reconfiguration and testing. The integration of this solution with SDN technology requires a very significant increased effort.

  - Operation effort (High) – the operation effort of this security solution can be considered high if integrated with SDN technology because it requires constant monitoring to ensure that there is no impact on the production area's operation. Plus, if there is the need to integrate new devices or change their network configuration, there is an additional effort to adjust the network configure firewall rules, SDN settings, or new network segment creation.
  - Financial cost (High) – besides the high effort of implementation and operation translating into a high cost, integration with SDN technology also implies the acquisition of compatible network hardware and its corresponding software management solution.

- Malware Endpoint Protection

  - Security Improvement (Medium) – as this security solution requires implementing the software or agents on the endpoint devices, installing it on all of them would be necessary for maximum effectiveness. Nevertheless, as there are device types where this is impossible, as already explained, its applicability is only achievable on a limited device type, such as on Microsoft Windows operating systems-based devices with the required processing power.
  - Implementation effort (High) – implementation requires installing software on as many devices as possible, which would require validation to ensure that the operational characteristics of the devices are not affected. Such validation is a lengthy and complicated process.
  - Operation cost (Medium) – during operation, it must be ensured that existing devices agents or software is continuously upgraded and the consequent installation for the new devices is executed.
  - Finance cost (High) – purchasing and maintaining software licenses and validating that the devices' operational characteristics have not changed would have a high financial cost.

- Intrusion Detection (static rules)

  - Security Improvement (Medium) – identifying intrusions in real-time is a substantial improvement in monitoring a production area. It benefits from taking actions to minimize threats or prevent their actual occurrence. As this solution is based on static rules, it has the advantage of having fewer false positives. However, it is ineffective for attacks unknown to the rules, such as zero-day attacks.
  - Implementation effort (Low) – because this type of solution is implemented in the network domain, it offers a more straightforward implementation and covers all connected devices.
  - Operation effort (Low) – because new rules are being created frequently, it requires constant attention to update them. However, their operation becomes more manageable because they are configured in a single central system – IDS server.

  - Financial cost (Medium) – acquiring an IDS solution that can interpret and analyze proprietary protocols specific to production or industrial area reduces the options available in the market, thus increasing its cost.

- Intrusion Detection (anomaly-based)

  - Security Improvement (Medium) – like the previous solution, real-time intrusion identification substantially improves the monitoring of a production area. It benefits from taking actions to minimize threats or prevent their actual occurrence. However, since it is based on network traffic identification that distinguishes one from the baseline created first, it offers the benefit of identifying zero-day attacks. On the other hand, this type of solution has the disadvantage of a high number of false positives.
  - Implementation effort (Low) – because these solutions are implemented in the network domain, they offer easy implementation and cover all connected devices.
  - Operation effort (Medium) – the operation requires more effort when compared to the static rules solution due to the higher number of false positives, requiring analysis for each possible security event.
  - Finance cost (Medium) – acquiring an IDS solution that can interpret and analyze proprietary and specific protocols for production or industrial area reduces the options available in the market, thus increasing its cost.

Analyzing the four dimensions mentioned and shown in the next chapter, when we weigh these security measures according to the identified threats, it is clear that more than one measure can be applied to mitigate the same threat.

However, it is already possible to highlight that the ”Security Improvement” dimension is the only one that can vary depending on the threat to be mitigated. The remaining dimensions related to the effort or cost of implementation are independent and not variable of the threat because, in the context of a production area, the measure has to be applied to a whole – e.g., a communications network – even if it would only impact a part of the devices.
Table 2.3 shows a summary of the previous analysis, which allows an easier way to identify the most suitable security solutions to apply in a production or industrial area.

2.3 Summary

At the end of this chapter, using the knowledge gained from the articles, reports, and remaining literature reviewed, it has been possible to identify the major topics that impact the security of production areas. Based on this, the most promising security measures were identified, and an impact evaluation of their implementation and the effort and cost of deployment were performed.

Security measure	**Security Improv.**	**Implem. Effort**	**Operation Effort**	**Finance Cost**
Asset identification	Medium	Low	Low	Medium
Device vulnerability	High	Low	Medium	Medium
Network segmentation	High	High	High	(Very) High
Malware endpoint protection	Medium	High	Medium	High
Intrusion Detection (static rules)	Medium	Low	Low	Medium
Intrusion Detection (anomaly-based)	Medium	Low	Medium	Medium

Table 2.3: Security measures improvement, effort and finance cost evaluation.

Chapter 3

Threat model

In this chapter, an initial reflection will be made on the challenge of achieving the ultimate goal of assigning the "security" property to a system. Followed by a review of what a threat model is, how it can be done, and what methodologies are used in this work. Contextualization of a production area will be made by presenting a real example. Finally, a threat model will be built based on a production area representing what is common to the various areas observed in this work. This model will have the primary threats identified and an attack tree that shows how they could be used to create a disruption. At the end of this chapter, an evaluation will be made of the security measures mentioned in the previous chapter that can best mitigate the identified threats.

3.1 Threat model process

What a threat model is intended to achieve, what it is, what the benefits are, and what steps must be taken to accomplish it.

3.1.1 Security as a negative goal

In a broad sense, security is a goal to be achieved in the face of an adversary. As such, a secure system is a system that continues to make a particular service available regardless of what an adversary might do.

This goal is challenging to achieve because, as Saltzer et al.[Saltzer and Kaashoek, 2009] state that security is a negative goal. Moreover, a negative goal is challenging to prove because it requires demonstrating that all possible threats are anticipated. Furthermore, having the time dimension in mind, the measures necessary must keep a system secure, correctly implemented, and maintained over time. So a designer designing a new system or improving an existing one must take a broad view of security and consider any method by which the security plan can be penetrated or circumvented.

To demonstrate this difficulty, consider a positive objective: "Peter must be able to read the contents of the file final_notes.doc." There is an easy way to tell if a system can achieve this goal; ask Peter to try to read the contents of the file.

However, considering a negative objective, “John cannot read the contents of the file final_notes.doc”. In this case, asking John to check if he cannot read the file’s contents is insufficient. It is good to check it, but it is not enough. Instead, it is necessary to consider and think of all possible ways in which John can access the file’s contents. Which then begs the question,” In what different ways can John access the file’s contents?”

Moreover, the number of hypotheses that answer this question is much more significant and potentially unreasonably large. So a final conceptual question arises,” When should we stop thinking about the possible scenarios where the goal is not achieved, i.e., the system is not secure?” Answering this question is what a threat model sets out to do. A systematic process ensures that the most assertive answers to respond to the negative objective of making a system more secure by executing it can be achieved. In an information security context, the result of a threat model enables informed and appropriate decisions to be made to protect a system throughout its operational life in an organization.

3.1.2 What it is

A threat model is a structured process or approach that identifies and prioritizes potential security or privacy threats to a given system, such as structural vulnerabilities or lack of safeguards. Furthermore, it determines the value of potential mitigating actions in reducing or neutralizing these threats. Through systematic analysis, a threat model aims to provide designers with the security controls that need to be included in the system under analysis to make it more secure. A set of risks can be identified and quantified based on information about the system’s nature, the profile of an adversary, the most likely attack vector, and which assets are most desired by an adversary. A threat model answers questions such as:

- What is the function of the system?
- What part of a system is most vulnerable to attack by an adversary?
- What are the main threats?
- What needs to be done to protect the system from the identified threats?

Most people perform a similar analysis in their daily lives without realizing it. Moreover, more precisely, a threat model has been carried out in a military defense context since ancient times. There is no exact time in an information security context, but the first initiatives to standardize a methodology emerged in the late 1990s.

3.1.3 Benefits

It is unnecessary to be a security expert to draft a threat model.As stated in the Threat Model Manifesto Braiterman et al. [2021] it is to be made by everyone and anyone concerned with the privacy, security,

and safety of a specific system. Programmers, system designers, and software designers or architects should aim to include a threat model in their analysis. And when adopted, it should be reviewed now and then because it is in the nature of systems to have some degree of mutability. This manifesto provides guidelines for an approach steered by the following values and principles.
Values:

- A culture of finding and fixing design issues over checkbox compliance.
- People and collaboration over processes, methodologies, and tools.
- A journey of understanding over a security or privacy snapshot.
- Doing threat modeling over talking about it.
- Continuous refinement over a single delivery.

Principles:

- The best use of threat modeling is to improve the security and privacy of a system through early and frequent analysis.
- Threat modeling should be aligned with an organization's development practices and follow design changes in iterations that scope manageable parts of the system.
- The results of threat modeling are meaningful when they are valuable to stakeholders.
- Dialogue is the key to establishing the common understandings that lead to value, while documents record those understandings and enable measurement.

The following guidelines should always be followed so that the result is as close as possible to a correct system representation:

- Systematic Approach – Achieve thoroughness and reproducibility by applying security and privacy knowledge in a structured manner.
- Informed Creativity – Allow for creativity by including both craft and science.
- Varied Viewpoints – Assemble a diverse team with appropriate subject matter experts and cross-functional collaboration.
- Useful Toolkit Support approach with tools that increase productivity, enhance workflows, enable repeatability and provide measurability.
- Theory into Practice Use successfully field tested techniques aligned to local needs, and the latest thinking informs on the benefits and limits of those techniques.

Moreover, the following forms lead to possibly misleading representations, distorting a correct representation of the system:

- Hero Threat Modeler – Threat modeling does not depend on one's innate ability or unique mindset; everyone can and should do it.
- Admiration for the Problem – Go beyond just analyzing the problem; reach for practical and relevant solutions.
- Tendency to Overfocus – Do not lose sight of the big picture, as parts of a model may be interdependent. Avoid exaggerating attention on adversaries, assets, or techniques.
- Perfect Representation – Creating multiple threat modeling representations is better because there is no single ideal view, and additional representations may illuminate different problems.

3.1.4 How is it done

Making a threat model applies to both a simple application and a more complex system. The principles and methods used are equally valid in the design or construction phase or in changing an existing system. It does not matter if we apply it to a web, mobile, more traditional application, or even a server cluster or network infrastructure.

An approach based on sequential steps makes it possible to achieve smaller, easier to achieve goals and make the whole model execution more effective, rather than a single, more considerable step. As Adam Shostack mentions in the introduction of his book "Threat modeling Designing for Security" Shostack [2014] it is essential to answer the following questions to achieve these more reasonable objectives:

1. What is to be built?
2. What can go wrong with it once it is built?
3. What should be done about those things that can go wrong?
4. Did a decent job of analysis have been done?

These questions lead to a fourstep framework, illustrated in figure 3.1; modeling the system, enumerating the threats, addressing the threats, and validation.

Different methods can answer the above questions at each of these stages. This modularity gives the threat model flexibility that allows it to fit into the analysis of different types of systems. Like Lego pieces, it allows choosing different methods that fit into this structure. A threat identification Threat model like

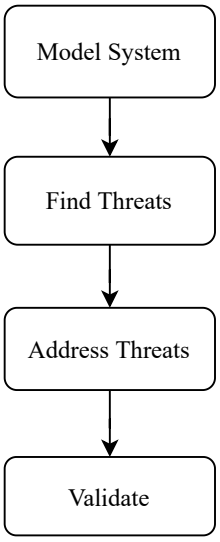


Figure 3.1: Four-Step Framework

STRIDE [1] allows it to fit with various diagrams used to illustrate and aid stakeholder communication and a DREAD model [2] for risk assessment.

Before looking in more detail at how the different stages of the threat model are carried out, it is essential to become familiar with several terms. In order to create a more accurate shared understanding of what they mean. Moreover, using the definitions listed by the OWASP [3] foundation, we have:

- *Policies* are a set of assumptions that define what it means for a system to be secure. It can be seen as the most basic general objective of a system's behavior concerning its secure state. These assertions are fundamental to delimit the scope of threats and associated risks relationship between likelihood and impact to be analyzed and assessed in the threat model.

- A *threat actor or adversary* is an individual or group capable of carrying out a particular threat. Identifying those that can exploit a vulnerability or have a motivation or opportunity against a business or organization is crucial. While some threats require a more excellent technological capability and cost that is only within reach of a prominent criminal or governmental organization, others may be relatively easier to execute with emerging technologies such as the cloud.

- The *Impact* is a measure – qualitative or quantitative – of the potential for harm caused by a partic-

[1] STRIDE is a model for identifying computer security threats developed by two Microsoft's researchers providing a mnemonic for six security threats categories.

[2] Risk-assessing computer security threats methodology that provides a mnemonic for risk rating using five security threats categories.

[3] The Open Web Application Security Project (OWASP) is an online community that produces free articles, methodologies, documentation, tools in the field of web application security at `https://owasp.org/`.

ular threat. Impact and harm can take many forms, and a threat can result in financial, reputational, or physical damage to an asset. They can be direct or indirect harm, and both need to be considered in the threat model. A concrete example is the disclosure of a company's intellectual property or user information about a product under development, which would damage the company's public image and consequently could result in a drop in sales, as the company would no longer be seen as trustworthy in the eyes of customers.

- The *Likelihood* is a measure of the possibility that a threat could be carried out. A wide variety of factors can increase or decrease the likelihood of an attack, i.e., how easy it would be to implement the threat or benefit the threat actor. Alternatively, put another way, what is the cost benefit ratio for the attacker? For example, suppose a threat requires a threat actor with high technical skills to spend tens of thousands of euros in computing power or years to execute it, only to obtain low value information that would likely be publicly available. In that case, the likelihood is very low. However, if, on the other hand, an efficiently executed threat can be carried out quickly and the attacker gets access to sensitive information from which he can make a profit, then the likelihood is much higher.

- *Risk*, as formally defined in ISO 31000 for Standardization [2018c], "is the effect of uncertainty on the objectives". Alternatively, put another way, it is the relationship between impact and likelihood. The assessment of risks and their classification is fundamental so that it is possible to prioritize their treatment in the third step of the fourstep structure.

- As defined in ISO 27001 for Standardization [2014] *Security controls* are safeguards or countermeasures to prevent, detect, neutralize or minimize risks to the security of physical property, information, computer systems, or other assets. In information security, such controls protect information or systems' confidentiality, integrity, and availability.

- *Preventions* are controls that can completely deter a particular threat when implemented. For example, if a threat is identified that exploits a vulnerability of specific functionality in an application or system, e.g., a specific type of remote access and if that functionality is disabled, then it can be said that prevention has been implemented. That is, the likelihood of the threat has been reduced to zero.

- Similarly, *mitigations* are controls that, when implemented, reduce the likelihood or impact of a threat, i.e., reduce the risk, without meaning its complete prevention. An example of mitigation is if the hashes of users' passwords are stored in a simple form in an application, then two users with the same password will have the same hash. If an adversary has access to the stored hashes, they may prefer to attack them because if they can crack the password, they will be able to gain access with the credentials of users sharing the same hash. One mitigation would be adding salt to these

hashes to make them unique, thus increasing the cost or effort the attacker would have to perform. Furthermore, increasing this effort means reducing the likelihood and mitigating the attack.

- A *Data flow diagram* represents how information flows through a system or application and its subcomponents, processes, or subsystems. Showing where information enters or leaves each process or subsystem, including where information is stored, either temporarily or permanently.

- Finally, we have a *trust boundary*, which in a threat model context, is a location in the data flow diagram where information changes its trust level. When information flows between two processes, it is very likely to cross a trust boundary. As Adam Shostack mentions in Part I Shostack [2014] a trust boundary is conceptually similar to an attack surface, for example, in a ship or submarine, the hull is obviously an attack surface, but it is simultaneously a trust boundary. The reason is that the vessel's interior can be defined as trusted, while the exterior is not to the same degree. Thus, an attack surface is a trust boundary that presents a direction through which a threat can be made. The definition of these boundaries in a data flow diagram is helpful for the implementation of controls to have a more significant effect by pointing to where in the system their application is most effective.

Once the critical concepts offered by this terminology have been defined and understood, it is possible to describe the process of developing a threat model. As we saw earlier, the first step in the fourstep framework is to create the system model, which will be accomplished in this work. And as a first thing to do, is to start by identifying the business objectives that the application or system has to achieve and any other security or compliance requirements that are necessary due to mandatory regulations. These business objectives and requirements are also mentioned in clause 4 of ISO27001for Standardization [2014], establishing the "context of the organization". A good example is the pharmaceutical manufacturing industry, where several regulations define safety measures or confidence levels that systems or applications must meet. Having these objectives in mind prior to threat identification will help assess the impact of any threat encountered.

3.1.4.1 Model System

In order to answer the first question, "What is to be built?" it is vital to use data flow diagrams to describe or model the system. So understanding the design of a system or application is fundamental to making the threat model. Even if we are familiar with its design, we may come to identify additional data flows or confidence limits during the risk analysis and assessment process. Because then, by understanding how the system is designed, it will be possible to assess the likelihood and potential impact of any identified threats.

When evaluating an existing system with previous documentation of its design, it is more beneficial to review it. The documentation may be outdated, requiring more recent information to be gathered. Alternatively, there may be no documentation whatsoever requiring it to be created.

In an optimal case, the threat model is preferably performed early in the project and system design to incorporate the model's output in the same design phase. This concept is known as "security by design" instead of "security bolted on" where controls are applied after the design phase. As we will see later in this paper, sometimes, this is the only alternative, such as when evaluating a change to an existing system or when the implementation of controls is limited.

There are many ways to generate a description and design of a system. However, the 4+1 architectural view model described by Philippe Krutchen in his paper" Architectural Blueprints The 4+1 View Model of Software Architecture" Kruchten [1995] provides a method for performing a comprehensive analysis using five different ways of looking at the system, using input from the various stakeholders. As shown in figure 3.2, we then have the **logical view** in which the designers describe the functional requirements for the end user. Then we have the **implementation view** where the programmers describe the components and the subsystems. Next, we have the **process view** created by who does the integration, focusing on the performance or scalability requirements; finally, we have the **physical view** where there is a description of how the integration of the software with the hardware is performed. The additional view is called the **use-cases view**, in which all the stakeholders describe how the previous views work together in an integrated way through the use cases.

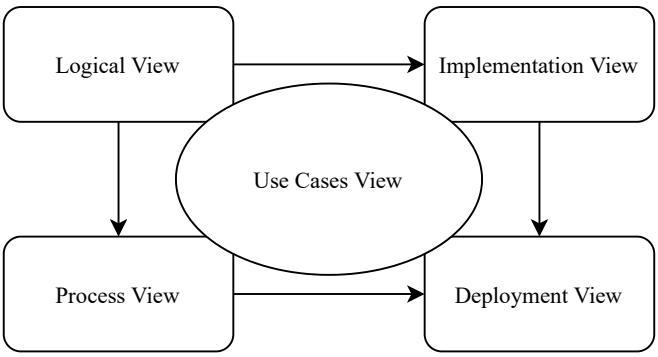


Figure 3.2: "4+1" View Model

It is then possible to start modeling how the system works with its documentation. It is essential to understand how it interacts. The first step is creating flow diagrams and trust boundaries at a high and more generic level. To model the system, the following steps must be carried out:

1. Identify the trusted boundaries of the system, application, module, or environment that is to start with
2. Add actors – internal and external
3. Define internal trusted boundaries. These can be the different security zones that have been designed

4. Relook at the actors identified in step 2 for consistency

A threat model is suitable for analyzing various types of systems, so it is vital to choose which approach best fits the case study. There are situations where it makes more sense to focus the model on assets. Alternatively, in other cases, it might make sense to focus on possible attacks on the software. However, when choosing one of these focuses is preferable to select an approach that combines all three, which tends to be confusing. As Adam Shostack mentions in chapter 2, "Strategies for Threat modeling" Shostack [2014] "(...)These three approaches can be thought of as analogous to Lincoln Log sets, Erector sets, and Lego sets. Each has a variety of pieces, and each enables you to build things, but they may not combine in ways as arbitrary as you'd like. (...)".

Security specialists with experience in structuring their thinking around assets may prefer focusing on assets to establish a dialogue with less technical people. It may be easier to rationalize, "What things have the most value?" In these cases, it is necessary to define assets in three families 3.3; Things adversaries want, Things needed to be protected, or assets that can be used as stepping stones to achieve the previous two families. Examples of things adversaries want could be user keys or passwords, personal information, credit card numbers, or confidential information. Things that need to be protected might be a little less intuitive. However, an example would be the need to maintain the integrity of a temperature log of a refrigerated area where perishable substances are stored so that they can be validated and audited by an external regulatory body.

Finally, we have the assets that can be used as stepping stones, such as devices with simultaneous access to the Internet and a company's internal network. While these may not be assets that adversaries want or that an organization does need to protect, they are in a favorable position to attack more desirable or critical assets.

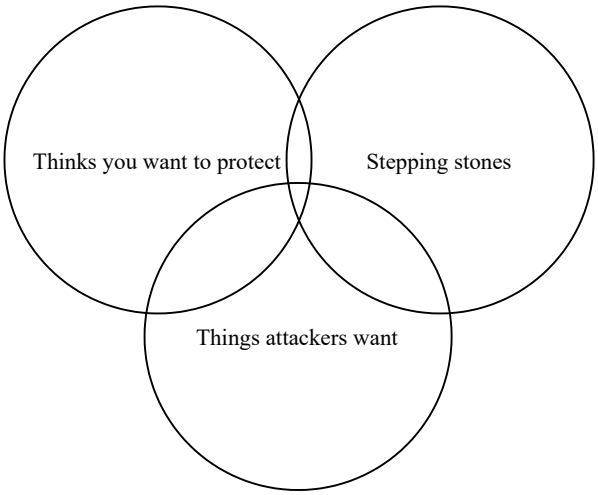


Figure 3.3: The overlapping definitions of assets

Focusing on attacks also seems to be a natural way to perform threat modeling, but perhaps it only makes sense in more specific cases. Whereas with the asset focus, getting information from less technical people is helpful. Using this as a starting point for a brainstorming approach for a given list of adversary types is possible. This attack focused approach can also be useful for analyzing and highlighting the human behavior factor. Such as how an adversary might bribe an employee.

Focusing on the software will be the most helpful method, especially in software development. A data flow diagram (DFD) can be created using the existing documentation to illustrate how information flows throughout an application or system. As Larry Constantine mentions in his book" Structured Design" [Yourdon and Constantine, 1979], a DFD "(...) consists of numbered elements (data stores and processes) connected by data flows, interacting with external entities (those outside the developer's or the organization's control)". As shown in figure A.1, the information in transit is represented by the bidirectional arrows, while two parallel lines represent the information at rest. While data at rest is often considered less vulnerable than information in transit, adversaries often prefer the former. The risk profile for information in transit or at rest depends on the security measures applied. Protecting sensitive information, in either case, is imperative in modern organizations, and adversaries have found increasingly creative ways to access or steal it.

There are several tools used to design a DFD:

- OWASP Threat Dragon is a cross platform project that allows designing a DFD, registering the possible threats, and deciding on their mitigation using a STRIDE methodology.
- Poirot is a tool that allows fault isolation and diagnosis through a fault modeling and simulation process.
- Microsoft Threat Modeling Tool helps find threats in the design phase of software projects.
- SeaSponge is easily accessible online and is web based.
- IriusRisk is a platform for automated threat modeling.

The DFD should define the internal and external limits of trust, the role of users, and the privileges that the applications grant to internal or external identities. For example, users with higher privileges should be highlighted, as is the case of "admin roles". Finally, define the interfaces where potential adversaries can interact or load information in the system.

3.1.4.2 Find Threats

The central purpose of a threat model is, as the name implies, to identify the threats themselves. So the question must be answered, "What can go wrong with it once it is built?" which represents the bulk of the effort spent developing a threat model.

It is necessary to identify the threat actors using the principles of capability, motive, and opportunity and then associate them with system components they can directly interact with. Because the number of threat actors can quickly become large and difficult to work with, it must reduce by taking into account the following ideas:

- Treating them as equivalent classes
- Considering the attacker's motivation when evaluating likelihood
- Consider Insider Threats

Threats can be identified in different ways, and depending on the systems under analysis, each method has advantages and disadvantages. However, in order to facilitate the identification of risks, the following sources and methods can be used:

1. Risks with OWASP Top 10.
2. Testing Procedure with OWASP ASVS.
3. Risks with SANS Top 25.
4. Microsoft STRIDE.
5. Attack trees

It is worth looking in more detail at the STRIDE method, where the initials stand for **S**poofing, **T**ampering, **R**epudiation, **I**nformation **D**isclosure, **D**enial of Service, and **E**levation of Privilege. This approach created by Loren Kohnfelder and Praerit Garg of Microsoft Kohnfelder and Garg [1999] was designed to help software developers identify the types of threats their software was subject to. Instead of using the properties intended for a system; confidentiality, authentication, integrity, nonrepudiation, availability, and authorization, the corresponding STRIDE properties are used, focusing more on the attacks' victims. The following Shostack [2014] shows the relationship of the STRIDE properties:

Threat	**Property Violated**	**Threat Definition**	**Typical Victims**	**Examples**
Spoofing	Authentication	Pretending to be something or someone other than yourself.	Processes, external entities, people	Falsely claiming to be Acme.com, winsock.dll, Barack Obama, a police officer, or the Nigerian Anti-Fraud Group.

Threat	Property Violated	Threat Definition	Typical Victims	Examples
Tampering	Integrity	Modifying something on disk, on a network, or in memory.	Data stores, data flows, processes	Changing a spreadsheet, the binary of an important program, or the contents of a database on disk? Modifying, adding, or removing packets over a network, either local or far across the Internet? Changing either the data a program is using or the running program itself.
Repudiation	Non-Repudiation	Claiming that you didn't do something, or were not responsible. Repudiation can be honest or false, and the key question for system designers, what evidence do you have?	Process	Process or system: "I did not hit the big red button" or "I did not order that Ferrari." Note that repudiation is somewhat the odd threat out here? It transcends the technical nature of the other threats to the business layer.
Information Disclosure	Confidentiality	Providing information to not those authorized to see it.	Data stores, data flows, processes	The most obvious example is allowing access to files, email, or databases. However, information disclosure can also involve filenames ("Termination for John Doe.docx"), packets on a network, or the contents of program memory.
Denial of Service	Availability	Absorbing resources needed to provide service.	Data stores, data flows, processes	A program that can be tricked into using up all its memory, a file that fills up the disk, or so many network connections that real traffic cannot get through.

Threat	Property Violated	Threat Definition	Typical Victims	Examples
Elevation of Privilege	Authorization	Allowing someone to do something they're not authorized to do.	Processes	Allowing a regular user to execute code as admin? Allowing a remote person without any privileges to run code.

Table 3.1: STRIDE properties.

Note that by using STRIDE to identify threats, only the things that can go wrong are listed. The exact mechanisms of how it can go wrong are something that can be developed further using attack trees or attack vectors. Also, during this phase, the following actions need to be conducted:

- Draw attack vectors and attacks tree.
- Identify Use Cases/Abuse Cases.
- Re-define attack vectors to consider multi-step attacks.

Once the use cases and abuse cases have been identified, it is essential to list all the possible abuse cases that should be developed for each use case. Furthermore, reassess the attack vectors because often, after identifying the initial vectors, these can lead to new vectors.
After identifying and typifying each threat, it is time to define the risk, i.e., the probability and impact, assuming a technically capable adversary with time and motivation, and knowing a zero-day attack. Because if not, then most likely, an underestimate of the risk associated with each threat will occur.
The DREAD methodology uses a simple mathematical formula, matching five categories to their parameters:

- **D**amage – how bad would an attack be?
- **R**eproducibility – how easy is it to reproduce the attack?
- **E**xploitability – how much work is it to launch the attack?
- **A**ffected users – how many people will be impacted?
- **D**iscoverability – how easy is it to discover the threat?

The DREAD formula is :
$RiskValue = (\text{Damage} + \text{Affected users}) \times (\text{Reproducibility} + \text{Exploitability} + \text{Discoverability})$
Another sound methodology for risk assessment is PASTA, the initials letters for Process for Attack

Simulation and Threat Analysis. Consisting of seven steps, this framework evaluates the security posture. Each step takes as input the output of the previous step. Based on business objectives, this method allows impact assessment at an early analysis stage rather than addressing them later in the risk assessment. The seven phases are:

1. Define the business objectives
2. Define the technical scope of assets and components
3. Application factoring and identify application controls
4. Threat analysis based on threat intelligence
5. Vulnerability detection
6. Analyse and model attacks
7. Impact analysis and development of countermeasures

Finally, quantify the risks in a risk matrix and order them, starting with the most severe. Equivalences below can use them to quantify qualitatively from a risk value:

- Risk Value: 01 to 12 →Risk Level: Notice
- Risk Value: 13 to 18 →Risk Level: Low
- Risk Value: 19 to 36 →Risk Level: Medium
- Risk Value: 37 to 54 →Risk Level: High

3.1.4.3 Address Threats

It is now time to answer the third question of the fourstep framework mentioned above," What should you do about those things that can go wrong?". Identifying the owners of the risks and then agreeing on how to treat them allows us to define the best controls to apply to each. Following a traditional method of risk treatment, four strategies can be used:

- **Reduce**: building controls in the form of code upgrades, confirming a specific design for the application, or building a specific configuration during the deployment phase to reduce application risk. Alternatively, implement a monitor system in order to detect a threat early.
- **Transfer**: For a specific component in the application, the risk can be transferred to an outsourced third party to develop that component. Furthermore, ensure that the third party is doing the proper testing for the component, or during the deployment phase, outsourcing a third party to do the deployment and transferring that risk to that third party.

- **Avoid**: an example of avoiding the risk is disabling a specific function in the application that is the source or disabling a system feature for that risk.
- **Accept**: if the risk is within acceptable criteria set earlier, in that case, the designer risk owner can accept that risk.

3.1.4.4 Validate

Finally, after all the hard work is done, there is one last step of the fourstep framework, answering the question," Did you do a decent job of analysis?". Whether it is a manual and human process using application testers or tools that automate this validation process, it is vital to test the measures applied, ensuring that the threats no longer have the same level of risk. Many organizations use penetration testing (Pentest) to supplement the threat model.

3.2 Production area context

It is essential to define what a production area means and how it is organized, according to how it was observed in the ZEISS company that served as the basis for this final work. However, this description applies to other production areas in other companies.
As figure 3.4 illustrates, the most general objective of a production area is transforming raw materials into a finished product. Of course, customers can eventually use this product in other production areas, thus continuing the transformation process. For the scope of this final work, the transformation processes are supported by mechanized technologies and fed with information that makes them more efficient and increases production capacity.

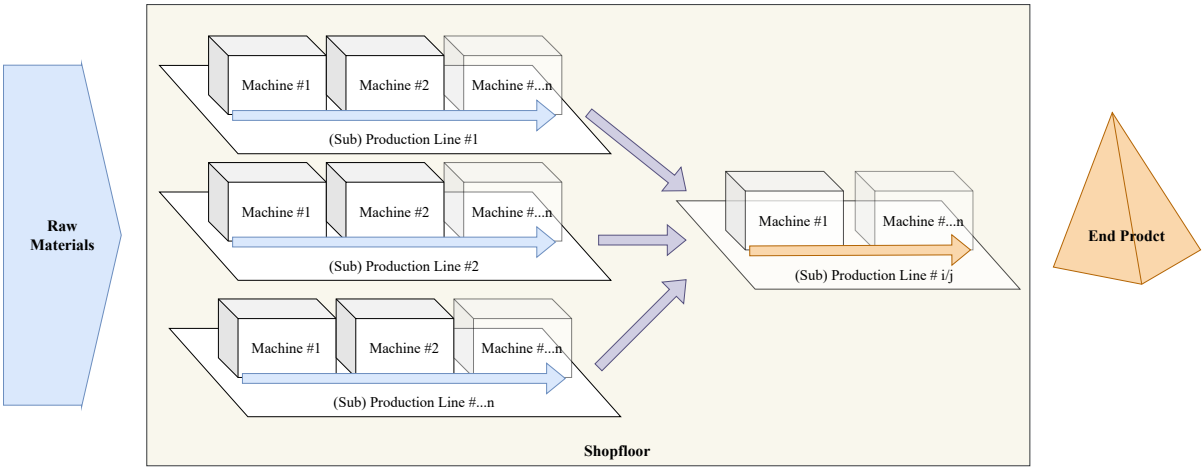


Figure 3.4: Abstract model of an production area

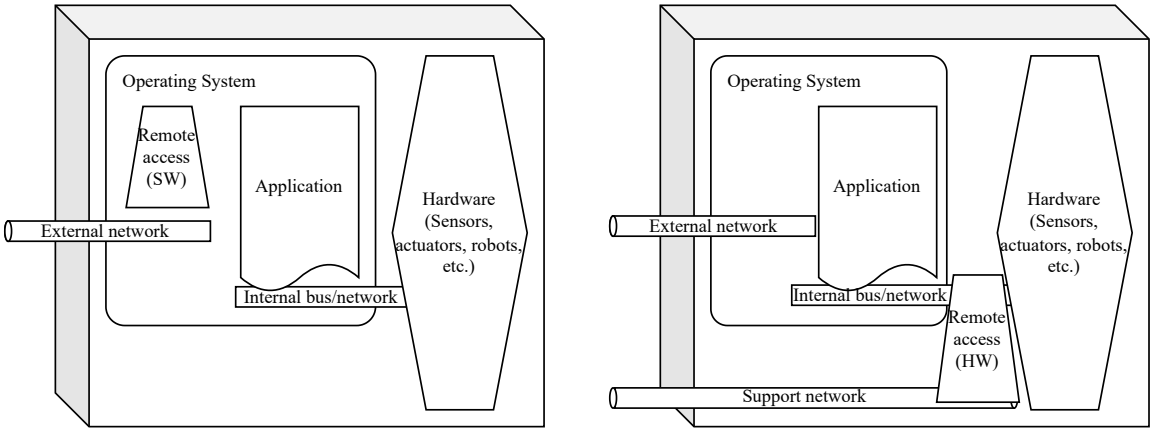
These production areas are composed of machines, each with a specific function, organized in production lines in which the transformation process is executed sequentially. Thus the output of one machine is the input of the next. A production area may be composed of several similar parallel production lines to increase its capacity. Alternatively, it may be composed of production lines that feed other lines to allow flexibility, like extending the product diversity and increasing the resilience of the production area as a whole.

From an information security point of view, a production machine should be seen as the smallest unit in information processing. The reason is that the deployment of any associated security risk mitigation measures cannot be performed on its sub-components.These sub-components are defined and implemented by the machine manufacturer, and any subsequent changes risk affecting the machine's proper functioning. However, to better understand the behavior and requirements of a machine, it is helpful to understand in a very abstract and general way how it is composed and how it connects to the network infrastructure, which serves to receive and send the information necessary for its operation.

In this document, two significant variants of these machines have been observed. Figures 3.5a and 3.5b show that both variants are composed of hardware such as robots or mechanical arms, pneumatic or electromechanical systems, or conveyor belts. Plus, a Microsoft Windows operating system (Linux was not observed on any production machines) running a specific application to control the hardware providing a graphical interface to the machine operator. However, the two variants differ in how remote support and maintenance can be performed. As seen in figure 3.5a, the network interface used by the machine to exchange information is simultaneously used for the remote support function, using an application such as VNC (on older machines) or Teamviewer running on the operating system. In the other case shown in figure 3.5b, a dedicated interface is connected to a device with this specific function and connects directly to the bus or internal network of the machine. This way allows another level of granularity in accessing the internal hardware. This dedicated machine access interface is often terminated in a device that allows out-of-band access, i.e., independent and separate access, for example, to a broadband modem. This type of access presents an immediate security risk for the machine owner, as it becomes much more difficult to implement measures to control and manage access to the machine and simultaneously to the network of the rest of the production area. A high degree of trust is implicit in this network and the devices, and as such, there are often fewer security measures implemented.

3.2.1 Two observed examples of a production area

Two practical examples are presented to better represent and render a production area's description.
The first is a production line part of an eyeglass lens production area. As illustrated in figure C.1, the process begins with "Blocking," where a metal part is fixed to the outside of the semi-finished lens serving as a mechanical support for other machines. Then the semi-finished lens goes through a milling process called "Surfacing" of the inner surface to obtain the optical properties that the ophthalmologist



(a) Main Configuration (b) Alternative Configuration

Figure 3.5: Production machine observed configurations

or optometrists prescribed. Then the lens is polished to remove all the marks made in the "Surfacing". Finally, some marks that guarantee the product's origin are engraved by lasers, such as the small and unnoticeable brand logo or a QR code containing the production date and warranty information. The production process then continues to the quality check and marking phases that allow for subsequent assembly in the frame. Figure C.2a, shows how a conveyor belt connects the milling and polishing machine. The lenses are transported in the trays shown in figure C.2b identified with a work number read by a bar code reader, which serves as an identifier for obtaining all the information necessary for its production. In this way, human intervention becomes unnecessary.

The second example is a production line for intraocular lenses implanted inside the human eye to treat cataracts or myopia, as shown in figure C.3. In this example, we have a production line at a later stage, where after the lens is produced, the fins are glued on to allow for correct placement by the surgeon into the patient's eye. A picture with the centimeter ruler is illustrated in figure C.4. Next, there is an essential inspection phase to ensure maximum quality. Afterward, the lens is placed in an injector with a liquid solution and is used directly in the operating room. Finally, sterilization and packaging finish this production phase. These production steps are performed in a clean room and under strict measures required by entities such as the U.S. Food and Drug Administration, European Medicines Agency, or National Medical Products Administration in China.

3.2.2 How is connected

A network infrastructure allows the connection of existing devices in a production area to obtain the necessary information to make them more efficient through automation.

Since the '90s, the Purdue model Williams [1994] has offered a reference for how a network with MES (Manufacturing Execution System) devices should be functionally structured. In this 5-level model,

systems are grouped by the function they perform:

- Level 0 – The physical process
- Level 1 – Intelligent devices
- Level 2 – Control systems
- Level 3 – Manufacturing operations systems
- Level 4 – Business logistics systems

According to this model, Levels 0, 1, 2, and 3 correspond to all the different devices on the shop floor. Furthermore, from a network security point of view, these devices are assigned an intrinsic trust status. So there is no network security measure advocated to limit their connections. In opposition to level 4, it is assigned a state of untrust, so this Purdue reference model recommends placing a firewall device to control and filter network traffic between levels 3 and 4.
The way the production areas are connected shows parallels with the Purdue model. All devices like production machines, PLCs, printers, and Windows-based desktops used in this context – e.g., for validation and management of the various production phases – and all other device types are therefore connected, creating a communication network. That is logically separated by using a firewall to the remaining company's network.

As it is possible to see in figure C.5, the first difference from what was observed in the production areas used for the realization of this work was that the MES is installed in the corporate network. The main reason for using the same MES is to manage several production areas with increasing frequency.

Another fact observed, and depicted in figure C.6, is the different ways of connecting devices to the Internet. With the recent introduction of more complex and IT-enabled devices, such as new production machines or IIoT, remote assistance and support are needed more often. Therefore the problem arises of connecting these devices to an untrusted network like the Internet.

Two broad ways have been observed to address this. The first is where the connection is established through the firewall and the rest of the corporate network. In this case, security measures are implemented to offer more significant access control guarantees. A second way is often related to machine manufacturers' difficulty in following the first way in the installation phase. For example, a broadband modem connects the machine to the Internet directly. This remote access solution raises concerns because it makes access control very difficult for those in charge of the production area and can arguably be called creating a "backdoor" with little control by the managers over the production area in practice.

It is also essential to draw attention to the fact that introducing devices such as the IIoT (Industrial Internet of Things) in these same networks requires rethinking how the Internet connection is made. This IIoT requirement means exposing the remaining devices with fewer security measures to the risks of unauthorized access by adversaries. Although it is not the scope of this dissertation, the in-depth study of the consequences of this same network sharing of IIoT devices with older devices – and consequently

less secure – is relevant to highlight this problem because, in the following threat analysis, this has to be taken into account.

3.3 Threat modeling in the production area

Threat modeling is the process in which potential threats, such as structural vulnerabilities or lack of security measures, are analyzed so that security controls are identified and proposed. The application design and programming teams often apply this process in software development. However, as mentioned in this chapter, it can also be applied, as in this case study, to the system formed by a set of production machines and their remote access processes, which can be found in the vast majority of production areas.

3.3.1 Modeling the system

The following sections will start by identifying the relevant assets that, although in an abstract way, typify the devices and processes observed in the production areas that were the target of this work. Next, the functional and non-functional properties of the data flows will be characterized, and their relationship between the assets and the different identified threat adversaries – mentioned previously as actors. A DFD will help describe the system under study, identifying the relevant threats and their relation to the assets, data flows, and threat adversaries.

The STRIDE methodology will be used to help reason and answer the underlying question, "What can go wrong with these systems I am analyzing?"
For the risk assessment of each threat, the DREAD classification will be used. For this context, it offers a pragmatic way to make it a more quantifiable classification to allow for prioritization in addressing the risk. For the ten most relevant threats, a possible attack tree will be drawn to graphically show how an attacker can use these to accomplish the goal of disrupting the production area.

Finally, a more detailed evaluation will be done on the security measures mentioned in 2.3 and the main threats identified in this chapter. To better describe the risk assessment process, it will be used three of the ten most critical threats as examples and demonstrate the criteria and reasoning used.
Table 3.2 shows the common types of assets in the researched production areas. Their functional description and element type helps to characterize them. Furthermore, the ID will be used as a reference later.

ID	Title	Function Description	Element type
A01	Machine PLC	Machine Programming Logic Controller (PLC) to control the machine subcomponents. It connects the machine to the organization's network.	Process
A02	Machine IPC	Industrial PC (IPC) to control the machine subcomponents. It connects the machine to the organization's network.	Process

ID	Title	Function Description	Element type
A03	Remote Support Service	Middleware Cloud-based remote support service. It provides a middle connect point to both the machine router and the external user client software. Provides authentication, authorization, and encryption and may also provide other security policies like time and date window allowed connection and logging.	External Entity
A04	Remote Support Router	Hardware router part of the machine and allows the vendor remote support and maintenance service. It enables access to the machine PLC through the cloud middleware service. The direction of the connection by initiating from inside the organization's network allows knowing precisely the traffic's source and destination. Being the source, the IP address of the router, and the connection destination, the middleware cloud-based service.	Process
A05	Manufacturing Execution System	The Manufacturing Execution System (MES) is used in manufacturing and production areas to track and document the transformation of raw materials to finished goods. The MES exchanges the information required to operate the production machinery by sending the job's parameters to produce or transform the physical part. Furthermore, receiving the information required for having an overall production status.	Process
A06	Machine Hardware	One or more PLCs or IPCs control internal machine hardware. This hardware is specific and related to the machine function and comprises sensors and actuators as robot arms with the respective tools. This asset is out of scope from the threat modeling.	Process
A07	Remote Support User #1	Asset to allow the vendor to perform remote support. It connects to the cloud middleware remote support service. It is one of the connection endpoints, and the other is the connection between the machine remote support router and the cloud middleware.	External Entity
A08	Remote Support User #2	Asset to allow the vendor to perform remote support. The remote user connects directly to the machine, most frequently with dedicated access using wireless broadband that is part of the machine.	External Entity

ID	Title	Function Description	Element type

Table 3.2: Asset list.

Similarly to assets, the following data flows mentioned in table 3.3 were identified in the analyzed production area and are frequent in other similar areas. Their functional description, i.e., their purpose, and non-functional description, i.e., how it is performed and what their properties are, describe how the assets communicate with each other. Furthermore, the ID starting with "C" for communication or data flow will later be used as a reference, and the "Element type" is "Data Flow" for all the table rows.

ID	Functional description	Non-Functional description
C01	Internal machine network connection between the PLC and the controlled internal sub-components.	This connection is out of the scope of the threat modeling, as it is specific to each machine.
C02	Machine to Machine (M2M) connection to exchange job status and other machine information – like Job-status completeness information (in-progress, waiting), machine status (running, stopped), measure or tolerances values.	Protocol: Profinet (TCP/UDP or over Ethernet frames for real-time) As the M2M should be almost real-time traffic, Profinet is often used. Moreover, due to the communication low latency functional requirements, authentication and encryption are frequently not enforced or seen as options.
C03	MES to machine connection to send the job parameters, like physical part shape and end dimensions. It also returns the end job status for production tracking.	Protocol: OPC-UA (TCP/UDP) To accomplish the most device compatibility OPC-UA is often seen as the most appropriate solution. OPC-UA offers security modes; 'Sign' or 'SignAndEncrypt' to ensure that authentication at the application level is enforced, plus ensuring integrity and confidentiality. The SecurityMode 'None' does not provide any protection. SecurityMode 'SignAndEncrypt' must be used if the integrity and confidentiality of data have to be protected.

ID	Functional description	Non-Functional description
C04	MES to machine connection to send the job parameters, like physical part shape and end dimensions. It also returns the end job status for production tracking.	Protocol: SMB (TCP/UDP) As a prevalent protocol to exchange information in the IT environment, it is also often used in the production areas. Although the SMBv2 and above use encryption to ensure end-to-end data protection from eavesdropping, SMBv1 does not offer encryption. An attacker who steals a password and logs into an endpoint can capture SMB 1 traffic, view it in plaintext, and even modify the stream to send false commands. Also, historically, SMB vulnerabilities have been used as an entry point into the end devices.
C05	MES to machine connection to send the job parameters, like physical part shape and end dimensions. It also returns the end job status for production tracking.	Protocol: FTP (TCP) As older production machines still use an old protocol, the FTP protocol offers authentication, but all data being transferred is in cleartext without encryption or integrity assurance.
C06	Internal machine connection between the PLC and the vendor router device used for the remote support.	This connection is out of the scope of the threat modeling, as it is specific to each machine.
C07	Remote support connection between the cloud middleware service and the remote support router in the machine.	Protocol: OpenVPN (TCP) or IPSEC Although the machine vendor provides this remote support solution, OpenVPN or IPSEC is used most. By default, OpenVPN can use HTTPS as a fallback mode, making it easier to adopt in an enterprise environment, and IPSEC is a well-known and security-proved protocol. To bypass the machine customer network hurdles, like traffic restriction on the organization firewall(s), internal wireless broadband (3G, LTE) capable routers are often used at the physical layer.
C08	Remote support connection between the cloud middleware service and the remote support user.	Protocol: OpenVPN (TCP) or IPSEC Although the machine vendor provides this remote support solution, OpenVPN or IPSEC is used most. By default, OpenVPN can use HTTPS as a fallback mode, making it easier to adopt in the client-side enterprise environment, and IPSEC is a well-known and security-proved protocol.

ID	Functional description	Non-Functional description
C09	Remote support connection between the IPC running inside the machine and the remote support user directly.	Protocol: VNC (TCP) Although the machine vendor provides this, VNC is often used. To bypass the machine customer network hurdles, like traffic restriction on the organization firewall(s), internal wireless broadband (3G, LTE) capable modems are often used at the physical layer.

Table 3.3: Data flows list.

Identifying the type of adversary that can be a threat is fundamental because it directly influences two of the three necessary factors that every attacker must have, ability, i.e., knowledge, and opportunity. The third factor, motivation, is personal and difficult to characterize and infer. Table 3.4 shows the three most common types of an adversary and each of their type of capabilities.

ID	Description
TA01	External adversary impersonating a remote support user
TA02	Internal adversary with access to the production network
TA03	Internal adversary with physical access to the production endpoint devices

Table 3.4: Threat adversaries.

3.3.2 Data Flow Diagrams

The DFDs 3.6 and 3.7 representation depicts how the assets and data flows are related. Furthermore, how in a logical way, where they are situated. These represent the same model but are split into two diagrams to improve their readability. Previously defined IDs are used to make it easier to interpret, the "A" for assets and "C" for communications or data flows. The trust boundaries are represented using colored rectangles and dashed lines.

This model illustrated in figure 3.6 consists of a production line composed of 3 examples of machines with different characteristics, representative of a real case.

All machines have internal hardware specific to their function. This hardware is controlled by a PLC for the case of an older machine or a machine with a more straightforward function where a processing unit with less performance is sufficient. Or by an IPC with much greater processing and storage capacity. This IPC is an *industrial computer*, more robust and able to operate in environments with more extreme physical conditions – i.e., temperature, humidity, continuous operation, or electromagnetic interference.

To these machines, remote access processes and technologies are associated with the machine manufacturers' operation support or maintenance. Exemplified in this model by direct access to a machine and

another using dedicated hardware, which in the most recent implementations uses a middleware service hosted in the cloud where it is possible to apply access and authentication policies. An MES has the function of controlling the existing processes in the production area.

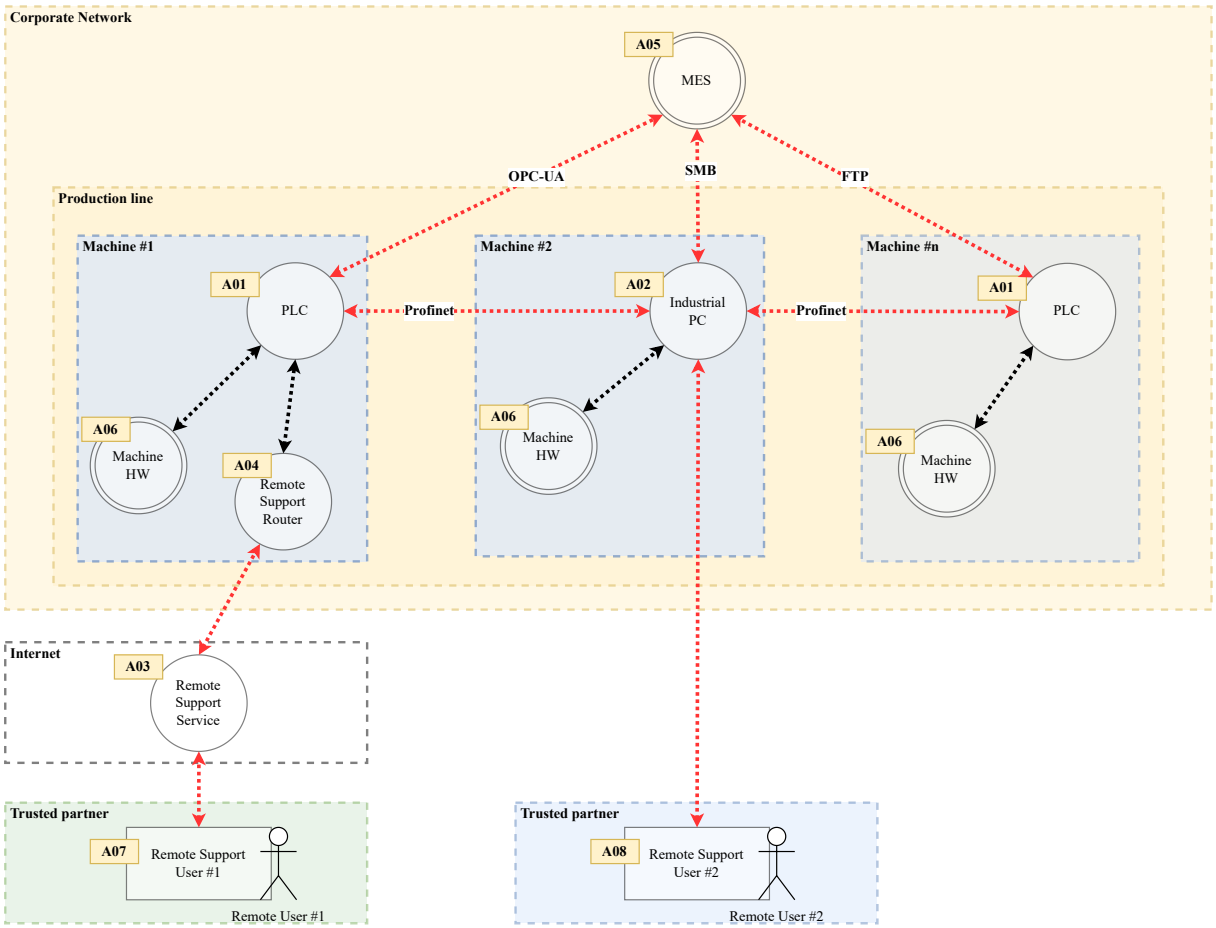


Figure 3.6: Threat modeling – Assets

Analyzing the same diagram model, but focusing on data flows, represented in the figure 3.7 by red arrows, is composed of direct connections between machines to exchange information necessary for their coordinated operation. Moreover, the communication of each machine to the MES to receive the corresponding manufacturing processes parameters and send their status. The black arrows symbolize the internal data flows of machines that are not subject to analysis in this threat modeling because they are specific to each machine and not documented by the vendor. Meaning it is unfeasible to apply security measures to them.

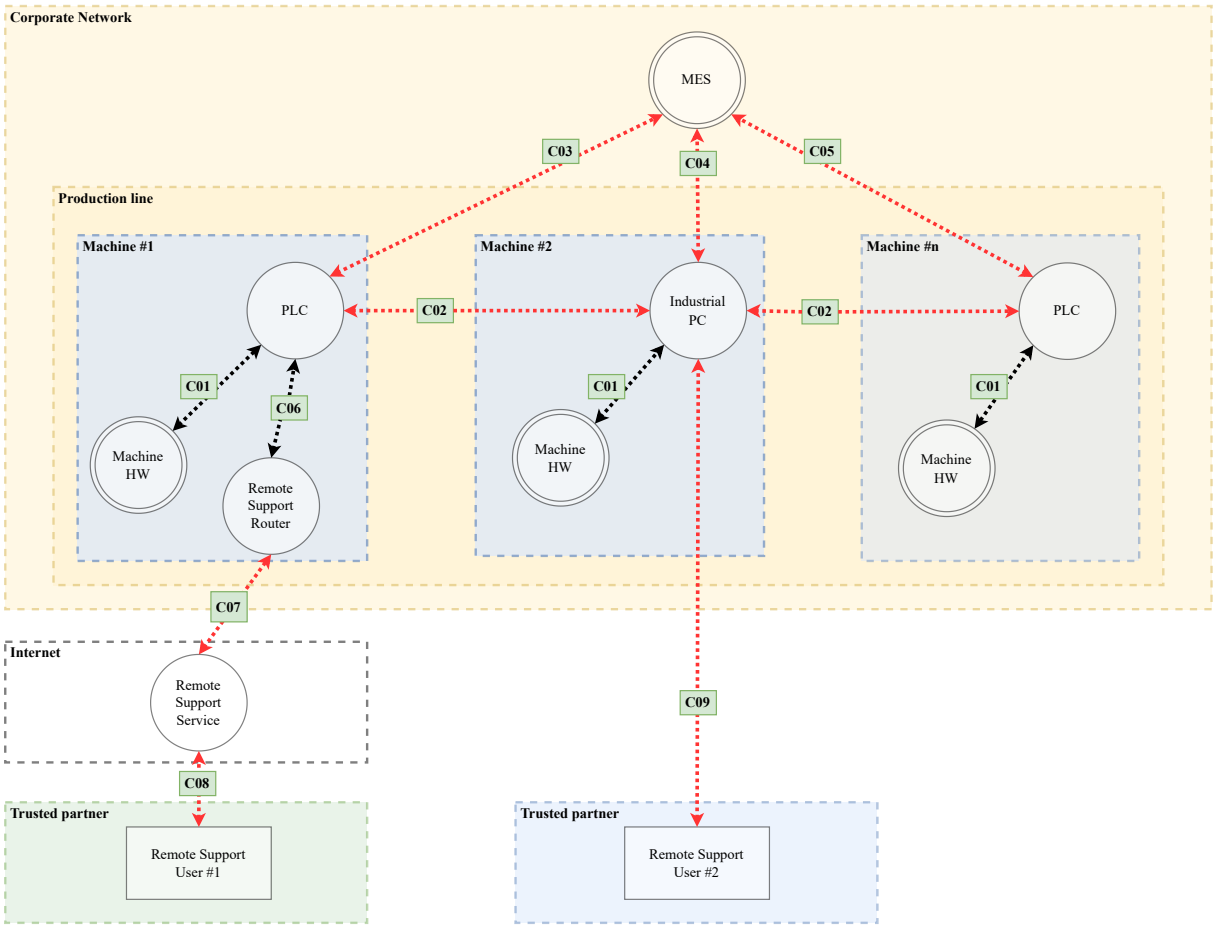


Figure 3.7: Threat modeling – Data Flows

In order to better identify threats, it is necessary to characterize the data flows better. Describing the architecture and information exchanged over network links between each asset helps better assess the impact of threats.

Below, the assets participating in the connection are at the top and on the left side. The complete table E.1 is shown in the appendix.

ID	A02
A01	**Architecture:** Peer-to-Peer **Data Flow:** C02 **Input/Output:** Job completeness status (in-progress, waiting), machine status (running, stopped), measures or tolerances values.

ID	A04
A01* A03†	**Architecture:** Peer-to-Peer **Data Flow:** C06*, C07† **Input/Output:** PLC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status).

ID	A05
A01* A02†	**Architecture:** Client-Server – A01 (client)*, Client-Server – A02 (client)†, A05 (server) **Data Flow:** C03*, C05*, C04† **Task #1 – Input:** Requesting the job parameters, sending Job ID number. **Task #1 – Output:** Job parameter necessary to produce. **Task #2 – Input:** Sending the job complete status, may contain more information like end tolerances, count of jobs produced, etc. **Task #2 – Output:** Message acknowledge.

ID	A06
A01* A02†	**Architecture:** Client-Server – A01 (server)*, Client-Server – A02 (server)†, A06 (client) **Data Flow:** C01 **Input:** Sensor data. **Output:** Data to actuators.

ID	A07
A03	**Architecture:** Peer-to-Peer **Data Flow:** C08 **Input/Output:** PLC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status).

ID	A08
A02	**Architecture:** Client-Server – A02 (server), A08 (client) **Data Flow:** C09 **Input/Output:** IPC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status). And, if needed, software update.

3.3.3 Threats

A *threat* can be defined as an action that generates an event with a negative impact on a system. This impact can affect the performance or the system's availability, thus preventing it from achieving its objective.

In a threat modeling process, various subject matter experts make the identification of threats. These can be security experts, programmers, device manufacturers, system integrators, or production area managers and operators.

For this work, it was not possible to assemble a team. Therefore the most pragmatic way to identify valid threats was to resort to those already identified and used in applications that help realize threat models. The list of 93 initially used threats can be seen in appendix D. After weighing each one in the case study context, 43 were chosen. Table 3.5 shows that the threats are associated with at least one asset or data flow. The threat adversary that can carry out the threat was also identified and associated with each threat. A threat identifier is used for later reference.

ID	Description	Asset ID	Data Flow ID	Threat Adversary ID
T01	Attackers gain access to the system or unauthorised data exploiting a known vulnerability (e.g. missing patch)	A01, A02, A03, A04, A05, A07, A08		TA02
T02	Attackers gain access to the system exploiting insufficient or misconfigured security features	A01, A02, A03, A04, A05, A07, A08		TA02

ID	Description	Asset ID	Data Flow ID	Threat Adversary ID
T03	Attackers try to retrieve banner information through the open ports to discover potential vulnerabilities	A01, A02, A05		TA02
T04	An adversary can bruteforce authentication credentials	A01, A02, A05, A07, A08		TA02
T05	An adversary guesses the default password and user of an authenticator (e.g. admin:admin)	A01, A02, A04		TA02
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks		C02, C04, C05	TA03
T07	Attackers could gain access to sensitive data through a man-in-the-middle attack		C03, C04, C05	TA02
T08	An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)	A02, A07, A08		TA02
T09	An adversary can execute remote code	A01, A02, A07, A08		TA02
T10	An adversary can perform a DoS attack by making consecutive authorization requests	A01, A02, A07, A08		TA02
T11	Unauthorized or unidentified human access to the asset	A01, A02, A04, A07, A08		TA01, TA03
T12	“An adversary guesses, obtains, or “”rides”” a trusted identifier (e.g. session ID, resource ID, cookie, etc.) to perform authorized actions under the guise of an authenticated user or service”	A03, A04, A05		TA01
T13	An adversary is able to connect to the asset with a compromised account and is able to make changes to all content available on the asset. (no roles)	A01, A02, A04		TA02

ID	Description	Asset ID	Data Flow ID	Threat Adversary ID
T14	Logging of the asset will fail when allocated storage capacity is satisfied. Attackers can take advantage of this by creating space-eater malware	A01, A02		TA02
T15	An adversary can complete lateral movement within the asset when no protection is set up	A02		TA02
T16	Unauthorized component access might occur as a result of not being able to restrict unnecessary functions, ports, protocols and/or services (i.e.:missing hardware or software hardening).	A01, A02		TA02
T17	Attackers gain unauthorized access to the admin account due to the lack of configuration of the account	A01, A02, A04		TA02
T18	An adversary may introduce malware into the asset by phishing (email,facebook,...)	A07, A08		TA01
T19	Attackers gain access to the system by exploiting weak security configurations	A01, A02, A04		TA02
T20	An adversary damages the device through physical access methods	A01, A02, A04		TA03
T21	An adversary exploits a weakness in authentication to create an access token to associate a process/thread	A01, A02, A05		TA02
T22	Attackers make undetected and unaudited changes to system configurations	A01, A02, A04		TA02
T23	An adversary could initiate a vast amount of sessions (DoS)	A02, A07, A08		TA02
T24	An adversary can intercept network traffic		C02, C03, C04, C05	TA02

ID	Description	Asset ID	Data Flow ID	Threat Adversary ID
T25	An adversary performs a DoS attack on a non-essential part of the asset which causes the core service to break	A01, A02		TA02
T26	Attackers gain unauthorised access to data or services by accessing a client side secret	A07, A08		TA01
T27	An attacker obtains an authoritative or reputable signer's private signature key by theft		C03, C08, C09	TA01
T28	Attackers gain unauthorized connection to the resources	A01, A02, A04		TA02
T29	Attackers try to take advantage of a wide attack surface	A05, A07, A08		TA01
T30	Unidentified software that executes arbitrary code	A02		TA01
T31	An adversary can use error messages that contain too much information, such as stack traces, to discover vulnerabilities in the running service	A02, A03, A05		TA02
T32	An attacker examines a target system to find sensitive data that has been embedded within it	A05		TA02
T33	An adversary accesses the asset via an untrusted network	A01, A02, A04		TA02
T34	An adversary is able to tamper with software running on a system. On the asset no integrity check is in place	A01, A02		TA02
T35	When a weak encryption algorithm is used an adversary is capable of breaking the algorithm with standard available tools	A01, A02		TA02
T36	Assets might lose the capability to maintain essential functions when operating in a degraded mode as the result of a DoS event or resource exhaustion	A01, A02, A03, A04, A05		TA02

ID	Description	Asset ID	Data Flow ID	Threat Adversary ID
T37	Unidentified backdoor plugged in the asset	A02		TA03
T38	Unauthorized component access might occur as a result of not being able to change default access credentials (example: hardcoded in the software)	A01, A02		TA02
T39	An adversary may craft messages that appear to come from a different principle or use stolen / spoofed authentication credentials		C03, C04, C05	TA02
T40	An adversary is able to delete audit records after a system breach which can make investigations very difficult	A01, A02, A04		TA02
T41	An adversary can fuzz a sensor connected to a PLC which will cause the controller to generate wrong output values which can cause big system failures (DoS)	A06		TA03
T42	An adversary bypasses the secure-boot, executes untrusted or adversarial boot code of the device	A02		TA02
T43	Sensitive data is compromised though attacks against SSL/TLS		C07, C08, C09	TA01

Table 3.5: Threats applicable to the case study

Each threat was classified using one of the STRIDE methodology category properties, as shown in table 3.1 and cross-referenced with the element types identified earlier – "Process" and "External Entity" for assets and "Data Flow" for network communications between devices. The 3.6 [Shostack, 2014] table provides a reference to classify each threat with one of the STRIDE properties.

ID	Description	S	T	R	I	D	E
T01	Attackers gain access to the system or unauthorised data exploiting a known vulnerability (e.g. missing patch)						•
T02	Attackers gain access to the system exploiting insufficient or misconfigured security features	•					

ID	Description	S	T	R	I	D	E
T03	Attackers try to retrieve banner information through the open ports to discover potential vulnerabilities				•		
T04	An adversary can bruteforce authentication credentials	•					
T05	An adversary guesses the default password and user of an authenticator (e.g. admin:admin)	•					
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks		•				
T07	Attackers could gain access to sensitive data through a man-in-the-middle attack				•		
T08	An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)						•
T09	An adversary can execute remote code						•
T10	An adversary can perform a DoS attack by making consecutive authorization requests					•	
T11	Unauthorized or unidentified human access to the asset		•				
T12	An adversary guesses, obtains, or “rides” a trusted identifier (e.g. session ID, resource ID, cookie, etc.) to perform authorized actions under the guise of an authenticated user or service	•					
T13	An adversary is able to connect to the asset with a compromised account and is able to make changes to all content available on the asset. (no roles)						•
T14	Logging of the asset will fail when allocated storage capacity is satisfied. Attackers can take advantage of this by creating space-eater malware					•	
T15	An adversary can complete lateral movement within the asset when no protection is set up						•
T16	Unauthorized component access might occur as a result of not being able to restrict unnecessary functions, ports, protocols and/or services (i.e.:missing hardware or software hardening).						•

ID	Description	S	T	R	I	D	E
T17	Attackers gain unauthorized access to the admin account due to the lack of configuration of the account	•					
T18	An adversary may introduce malware into the asset by phishing (email,facebook,...)						•
T19	Attackers gain access to the system by exploiting weak security configurations						•
T20	An adversary damages the device through physical access methods					•	
T21	An adversary exploits a weakness in authentication to create an access token to associate a process/thread	•					
T22	Attackers make undetected and unaudited changes to system configurations			•			
T23	An adversary could initiate a vast amount of sessions (DoS)					•	
T24	An adversary can intercept network traffic				•		
T25	An adversary performs a DoS attack on a non-essential part of the asset which causes the core service to break					•	
T26	Attackers gain unauthorised access to data or services by accessing a client side secret	•					
T27	An attacker obtains an authoritative or reputable signer's private signature key by theft		•				
T28	Attackers gain unauthorized connection to the resources	•					
T29	Attackers try to take advantage of a wide attack surface						•
T30	Unidentified software that executes arbitrary code						•
T31	An adversary can use error messages that contain too much information, such as stack traces, to discover vulnerabilities in the running service				•		
T32	An attacker examines a target system to find sensitive data that has been embedded within it				•		
T33	An adversary accesses the asset via an untrusted network	•					

ID	Description	S	T	R	I	D	E
T34	An adversary is able to tamper with software running on a system. On the asset no integrity check is in place		•				
T35	When a weak encryption algorithm is used an adversary is capable of breaking the algorithm with standard available tools				•		
T36	Assets might lose the capability to maintain essential functions when operating in a degraded mode as the result of a DoS event or resource exhaustion					•	
T37	Unidentified backdoor plugged in the asset						•
T38	Unauthorized component access might occur as a result of not being able to change default access credentials (example: hardcoded in the software)						•
T39	An adversary may craft messages that appear to come from a different principle or use stolen/spoofed authentication credentials		•				
T40	An adversary is able to delete audit records after a system breach which can make investigations very difficult		•				
T41	An adversary can fuzz a sensor connected to a PLC which will cause the controller to generate wrong output values which can cause big system failures (DoS)					•	
T42	An adversary bypasses the secure-boot, executes untrusted or adversarial boot code of the device						•
T43	Sensitive data is compromised though attacks against SSL/TLS				•		

Table 3.7: Threat STRIDE classification

Verifying a balance among the STRIDE categories helps gauge a comprehensive and valid set of threats. The graph 3.8 below shows a similar number of threat types. The exceptions are “non-repudiation” and “authorization”. However, an analysis in a production area context helps find a justification. A single “non-repudiation” threat type is an “assurance that someone cannot deny the validity of something” that is only relevant to ensuring the worth in system logs. Overall, repudiation issues are not critical for the shopfloor systems. On the other hand, “authorization” with a much higher value than the other categories makes perfect sense in an environment like a production area where devices do not have configured user rights or privileges to resources.

	S	T	R	I	D	E
External Entity	•		•			
Process	•	•	•	•	•	•
Data Flow		•		•	•	
Data Store		•	•	•	•	

Table 3.6: STRIDE-per-Element

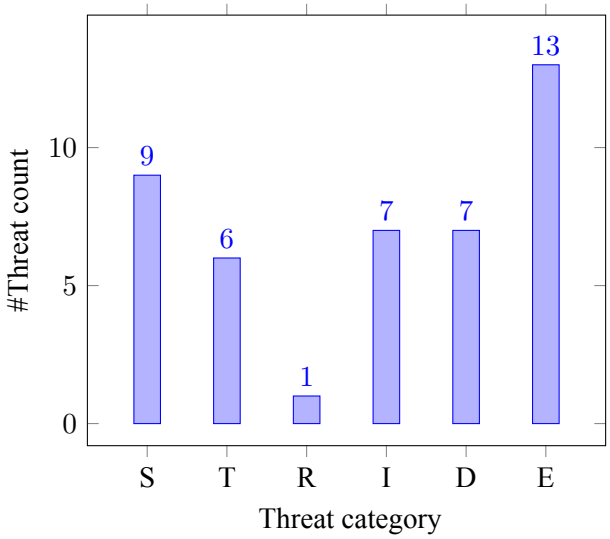


Figure 3.8: Threat count by STRIDE categories

3.3.4 Threat Evaluation

Once the threats have been identified and typified, it is necessary to assess them. This assessment using the DREAD methodology allows for identifying and ranking the threats with the highest potential risk. As mentioned in section 3.1.4.2, each property is assigned a score from 1 to 3 based on the criteria defined in appendix F. And the total score is the result of the mathematical expression:
$RiskValue = (\text{Damage} + \text{Affected users}) \times (\text{Reproducibility} + \text{Exploitability} + \text{Discoverability})$

It was decided to perform the threat assessment based on the total impact of all assets or data flows. An individual and more detailed analysis was unattainable as it would be more time-consuming than the one defined for this work. Therefore the total impact can be seen as a worst-case scenario. All threats can be found in appendix G.

The remaining work will focus on the ten highest scores threats. Therefore, the ones with the most significant potential impact. It is considered that there is a balance between what is needed to support

these final work arguments and the analysis feasible for this work duration.
Table 3.8 shows the ten most relevant threats.

ID	Description	Risk Value
T01	Attackers gain access to the system or unauthorised data exploiting a known vulnerability (e.g. missing patch)	48
T02	Attackers gain access to the system exploiting insufficient or misconfigured security features	42
T03	Attackers try to retrieve banner information through the open ports to discover potential vulnerabilities	36
T04	An adversary can bruteforce authentication credentials	35
T05	An adversary guesses the default password and user of an authenticator (e.g. admin:admin)	32
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks	30
T07	Attackers could gain access to sensitive data through a man-in-the-middle attack	30
T08	An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)	30
T09	An adversary can execute remote code	28
T10	An adversary can perform a DoS attack by making consecutive authorization requests	28

Table 3.8: Ten most relevant threats

3.3.4.1 Threat DREAD evaluation example

Although a risk assessment is always an exercise with some subjectivity, it is essential to better show the process performed and the criteria used for this work. To this end, three threats will be used as an example of what was performed with the remaining ten threats. The reasons for choosing these threats followed these criteria:

- A threat with the highest risk level
- A threat that focuses on assets
- A threat that focuses on data flows and is associated with a different threat adversary than the previous threat

Therefore, following these criteria, the threats mentioned in the table 3.9 were chosen. Furthermore, it will be the target of the exemplification that follows:

ID	Description
T01	Attackers gain access to the system or unauthorized data by exploiting a known vulnerability (e.g., missing patch)
T04	An adversary can bruteforce authentication credentials
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks

Table 3.9: Chosen three threats for DREAD score example

Example I:

ID	Description	Asset ID	Threat Adversary ID
T01	Attackers gain access to the system or unauthorized data exploiting a known vulnerability (e.g., missing patch)	A01, A02, A03, A04, A05, A07, A08	TA02

Table 3.10: First DREAD example

The assets ID, as mentioned above, correspond to the table 3.11

ID	Asset description
A01	Machine PLC
A02	Machine IPC
A03	Remote Support Service
A04	Remote Support Router

ID	Asset description
A05	Manufacturing Execution System
A07	Remote Support User #1
A08	Remote Support User #2

Table 3.11: Example I assets

The Threat Adversary ID as mentioned above correspond to the table 3.12

ID	Threat Adversary ID
TA02	Internal adversary with access to the production network

Table 3.12: Example I – threat adversaries

Based on the risk assessment criteria in appendix F, the table 3.13 has the DREAD scores assigned.

DREAD Property	Detailed reasoning
Damage (3)	Impact on all identified assets (except one), so the damage must be maximum.
Reproducibility (3)	Assuming an already compromised access to the network (TA02) and most exploits do not require authentication, the score is high.
Exploitability (2)	The majority of the exploits require some scripting or specific tool usage. These tools for the ICS devices are also particular for this device type.
Affected clients (3)	The number of clients affected is most of the systems analyzed (especially considering the impact on the MES system).
Discoverability (3)	Information about known exploits is public knowledge, for example, from the US agency CISA. There are reports of new vulnerabilities specific to ICS available in Cybersecurity and Agency-ICS[a]. As such, the score must be the highest.

[a]Cybersecurity and Agency ics-cert advisories https://www.cisa.gov/uscert/ics/advisories

Table 3.13: Example I – DREAD score reasoning

Example II:

ID	Description	Asset ID	Threat Adversary ID
T04	An adversary can bruteforce authentication credentials.	A01, A02, A05, A07, A08	TA02

Table 3.14: Second DREAD example

The assets ID as mentioned above correspond to the table 3.15

ID	Asset description
A01	Machine PLC
A02	Machine IPC
A05	Manufacturing Execution System

ID	Asset description
A07	Remote Support User #1
A08	Remote Support User #2

Table 3.15: Example II assets

The Threat Adversary ID as mentioned above correspond to the table 3.16

ID	Threat Adversary ID
TA02	Internal adversary with access to the production network

Table 3.16: Example II – threat adversaries

Based on the risk assessment criteria in appendix F, the table 3.17 has the DREAD scores assigned.

DREAD Property	Detailed reasoning
Damage (3)	The performance and availability impact on the most critical assets (production machines, MES, and remote users) is high. Therefore the damage has to be maximum.
Reproducibility (2)	Credentials brute force can be made using publicly available specific tools, but to be effective requires gathering preliminary information such as default admin username or firmware manufacturer/version due to the specific system in the production environments.
Exploitability (2)	Without the password hash, the efficiency of the brute force is reduced. Authentication with a less privileged user may be required to use a known exploit.
Affected clients (2)	The number of clients affected is in the most critical systems (especially considering the impact on the MES system), but not all are affected.
Discoverability (3)	The knowledge to perform the brute force is public domain and does not require deep knowledge, so the score must be the highest.

Table 3.17: Example II – DREAD score reasoning

Example III:

ID	Description	Data Flow ID	Threat Adversary ID
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks.	C02, C04, C05	TA03

Table 3.18: Third DREAD example

The assets ID as mentioned above correspond to the table 3.19

ID	Data Flow description
C02	Machine to Machine (M2M) connection to exchange job status and other machine information.
C05	MES to machine connection to send the job parameters (legacy protocol like FTP).

ID	Data Flow description
C04	MES to machine connection to send the job parameter.s

Table 3.19: Example III assets

The Threat Adversary ID as mentioned above correspond to the table 3.20

ID	Threat Adversary ID
TA03	Internal adversary with physical access to the production endpoint devices

Table 3.20: Example III – threat adversaries

Based on the risk assessment criteria in appendix F, the table 3.21 has the DREAD scores assigned.

DREAD Property	Detailed reasoning
Damage (3)	An adversary with physical access to the production area (TA03) and the ability to deploy rogue devices can significantly impact the entire system — for example, access to sensitive information (intellectual property of production processes or patient data).
Reproducibility (2)	Several steps are required to set up such devices. However, this information is public domain and requires no authentication.
Exploitability (1)	Although this is not impossible, it requires physical access to the production area and knowledge of its setup.
Affected clients (3)	The number of clients affected by these data flows is the most critical as they contain the information about the task to be performed by the machines (especially considering the impact on the MES system).
Discoverability (2)	Planting the device is not a trivial action, even with physical access to the production area, but it is not impossible to perform. The tutorial device construction is available at TunnelsUP.com[a].

[a] Raspberry Pi: Phoning Home Using a Reverse Remote SSH Tunnel - *"(...)The idea was to be able to plug it in somewhere and it be small enough that it's not noticed in someone's network. Then if I could access it remotely I am in their network and can do things."* https://www.tunnelsup.com/raspberry-pi-phoning-home-using-a-reverse-remote-ssh-tunnel/

Table 3.21: Example III – DREAD score reasoning

3.3.5 Attack Tree

An attack tree allows us to graphically and intuitively see how an adversary could chain threats to achieve the goal of disrupting the production area.

The diagram H.1 diagram shows the attack tree with the ten most relevant threats identified as tree leaves.
However, the threat *"T08 – An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)"* and *"T09 – An adversary can execute remote code"* are intermediate goals. It is also worth pointing out that this last threat potentiates and is common to other intermediate objectives – i.e., *"Malware or Backdoor"*, *"Denial of Service"*, and *"Configuration Tampering"*.

Although the scope of this work does not directly consider the physical security of the production area, in the attack tree, it was decided to mention "Physical destruction" and "Physical Access" because an adversary of type TA03 can, in principle, perform them.

3.3.6 Security Measures

After identifying and scoring the threats, it is necessary to verify which measures can mitigate them, previously identified at section 2.2 in table 2.3. However, more than one mitigation measure may apply to the same threat.

In the context of this work, the security measures listed above will be applied to the computer networks of the production areas – except for "Malware endpoint protection". Thus, reducing the cost, implementation effort, or operation is unattainable because it is only possible to apply the measures to the entire network and not to a part, even if their impact is not on all connected devices.

Using the "Security Improvement" dimension of the table 3.22, it is possible to choose which measure has the most significant mitigation impact and, therefore, the preferred one. An additional measure will necessarily have a lower impact, or the cost and effort of implementation will be higher than the preferred or primary one.

	Main Mitigation measure		Additional Mitigation measure	
ID	**Measure**	**Impact**	**Measure**	**Impact**
T01	Device vulnerability	High	Intrusion Detection(static rules)	Medium
T02	Device vulnerability	High	Network segmentation	Medium
T03	Intrusion Detection(anomaly-based)	Medium	Network segmentation	Medium
T04	Intrusion Detection(static rules)	Medium	Network segmentation	Medium
T05	Intrusion Detection(static rules)	Medium	Network segmentation	Medium
T06	Asset identification	Low	—	
T07	Intrusion Detection(static rules)	Medium	Network segmentation	Medium
T08	Malware endpoint protection	Medium	—	
T09	Intrusion Detection(anomaly based)	Medium	Malware endpoint protection	Medium
T10	Intrusion Detection(static rules)	Medium	Network segmentation	Medium

Table 3.22: Top 10 threats with the applicable mitigation measures

3.3.6.1 Security measures evaluation example

To better explain the reasoning behind assessing the impact of security measures, the following will explain the reasons, using the same threats used in the previous section.

Example I:

ID	Description
T01	Attackers gain access to the system or unauthorized data exploiting a known vulnerability (e.g., missing patch)

Table 3.23: First threat for the security measure evaluation example

Mitigation measures evaluation for the threat above.

	Main Mitigation measure		Additional Mitigation measure	
ID	**Measure**	**Impact**	**Measure**	**Impact**
T01	Device vulnerability	High	Intrusion Detection(static rules)	Medium

Table 3.24: Example I of the applicable mitigation measures

Main Mitigation measure:

- Device vulnerability (**High**) – This directly addresses the threat because knowing the vulnerabilities of the devices and the applicable patches allows vulnerabilities to be eliminated upfront and max-

imum impact to be achieved. Zero-day vulnerabilities are much rarer, so little affects the impact assessment.

Additional Mitigation measure:

- Intrusion Detection (**Medium**) – Detection of ongoing threats using rules is also a valid mitigation measure. It improves the response time to an incident but does not prevent it.

Example II:

ID	Description
T04	An adversary can bruteforce authentication credentials.

Table 3.25: Second threat for the security measure evaluation example

Mitigation measures evaluation for the threat above.

	Main Mitigation measure		Additional Mitigation measure	
ID	**Measure**	**Impact**	**Measure**	**Impact**
T02	Device vulnerability	High	Network segmentation	Medium

Table 3.26: Example II of the applicable mitigation measures

Main Mitigation measure:

- Intrusion Detection (**Medium**) – This measure directly addresses the threat; brute force can easily be detected using an IDS.

Additional Mitigation measure:

- Network segmentation (**Medium**) – This measure will reduce the scope of devices an adversary could perform brute force, not prevent it. However, the cost and efforts of implementation are very high.

Example III:

ID	Description
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks.

Table 3.27: Third threat for the security measure evaluation example

Mitigation measures evaluation for the threat above.

	Main Mitigation measure		Additional Mitigation measure	
ID	**Measure**	**Impact**	**Measure**	**Impact**
T06	Asset identification	Low	—	

Table 3.28: Example III of the applicable mitigation measures

Main Mitigation measure:

- Asset Identification (**Low**) – None of the measures effectively control physical access to the production area. However, the exact knowledge of the deployed assets helps identify rogue or foreign devices.

3.4 Summary

At the end of this chapter, it is now possible to conclude that the realization of the threat model, using the STRIDE methodologies to identify threats and the DREAD methodology for its risk assessment, was effective in producing a vast number of potential threats. Using the ten most with a greater risk was shown how it is possible to chain them together in an attack tree to disrupt the production area. As a result, we have the correspondence of the threats analyzed with the mitigation measures identified in chapter 2. Moreover, an assessment of the expected impact of the measure in mitigating the threat has also been done.

At this point, it is already possible to answer the first question of this work, *"With the many restrictions and particularities existing in the production areas and applying a threat model with the most appropriate methodologies, what are the most adequate controls or measures to implement?"*. As shown, the threat model produced a set of valid threats matched with the measures mentioned in the related work. It is then possible to conclude that the most adequate measures are those mentioned in table 3.22.

Chapter 4

Security Monitoring

4.1 Security Monitoring

Monitoring security events involves collecting and analyzing information to detect suspicious behavior or unauthorized system changes on the network, defining which behavior should trigger alerts, and taking action on alerts as needed.

Many well-known open-source tools can contribute to incident mitigation—for example, the well-known SNORT for intrusion detection or OpenVAS for vulnerability identification. However, when we have a specific context, such as production areas, where there can be no interference in its operation, it dramatically limits the choice of the most appropriate tool. Nevertheless, the most limiting choice criteria are the need for these tools to recognize ICS devices and their specific and often proprietary network protocols. For example, the lack of specific preprocessors in SNORT for some of the most frequent existing protocols in production areas, such as BACNet [1], Ethernet/IP [2], or Profinet RT [3]. The verification of its sources confirms this fact. These preprocessors allow handling data stretched over multiple packets, creating more practical and assertive rules for detecting threats.

For these reasons, commercial security event monitoring tools devoted to OT environments combine several features, such as identifying ICS assets and their characteristics, such as firmware versions, showing vulnerabilities in the devices, and finally, intrusion detection. This tool coverage makes them particularly interesting in a medium or large production area enterprise context. Therefore, this chapter

[1] BACnet is a communication protocol for Building Automation and Control (BAC) networks that leverage the ASHRAE, ANSI, and ISO 16484-5 standard protocol. http://www.bacnet.org/

[2] EtherNet/IP – IP meaning Industrial Protocol, not to be confused with Internet Protocol – is an industrial network protocol adapting Common Industrial Protocol (CIP) to standard Ethernet. EtherNet/IP is a common industrial protocol in the US and is widely used in various industries, including factory, hybrid, and process.

[3] PROFINET RT handles time-critical data exchange. An arriving PROFINET RT Ethernet frame has the PROFINET EtherType: 0×8892. Upon arrival at the destination node, the frame is directed straight from Ethernet (Layer 2) to the PROFINET application (Layer 7). The frame skips the TCP/IP layers and avoids the variable time it takes to be processed. Thus, communication speed and determinism improve significantly.

will analyze the tool's effectiveness – the last threat modeling Four-Step "Validate" step and answerthe question "Did a decent job of analysis have been done?" – with the functionalities already mentioned in an actual production area.

It is also interesting to show how this tool fits into a framework like the NIST Cyber Security Framework and its core functions mentioned in chapter 2, topic 2.2. Table 4.1 shows the relationship between each of the tools' features with the NIST core functions.
Therefore, it can be argued that the incident likelihood is reduced by applying security measures to the first three core functions.

Security Measure	**NIST CSF Core function**
Asset identification	Identify
Device vulnerability	Protect
Intrusion Detection	Detect
—	Response
—	Recover

Table 4.1: Security Measures relation with the NIST CSF core functions

4.1.1 Commercial Solution Market

Although there has been a clustering of these tools in recent years with the market entry of leading technology companies such as Cisco and Microsoft – buying smaller companies and adding their products to their portfolio – there are still more than a dozen similar tools. See appendix I table with the solutions identified.

It is impossible to test them within the scope of this work. However, it is possible to resort to some public reports by advisory companies like Gartner or Forrester. Through their website, Gartner [4] presents a comparison based on the scores given by its community of peers.

Nevertheless, the 2021 last quarter's report from Forrester is more interesting, where they compare 12 incident monitoring solutions in greater detail. Unfortunately, how they performed the comparison or collected the information is not explained. However, the representation illustrated in figure I.1 in appendix I shows all the vendors used in the report and which are the solution's type leaders.

4.2 Proof of value

The proof of value was conducted in the production area belonging to the ZEISS company located in the German city of Oberkochen. The two solutions, Cisco Cyber Vision and Nozomi Guardian were chosen as an internal project of the company and as part of this dissertation. This decision is because these

[4] Gartner peer review - products in operational technology (OT) security market. https://www.gartner.com/reviews/market/operational-technology-security#

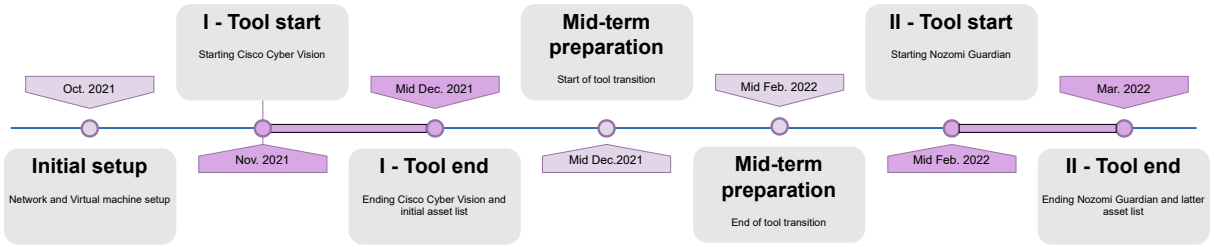


Figure 4.1: Proof of Value timeline

solutions fit the company's technological strategy and the cooperation with the current partner supporting and operating the network infrastructure. The choice of this production area took into account its size, with about 200 devices and device diversity.

Because it was not possible to test the solutions simultaneously, it was decided to run them sequentially. First, a preparation phase was necessary. Software and hardware updates were made to the network equipment and their reconfiguration to accommodate the components – agents and the virtual machine for the management console. Next, the Cisco Cyber Vision solution was installed and tested for approximately one and half months, followed by an intermediate phase of 2 months to remove the latter solution and prepare and install the Nozomi Guardian solution, which was also run for approximately the same period as the previous one. The timeline is shown in figure 4.1.

4.2.1 Architecture

These tools have identical architectures. All the information collected by these tools is by "listening" – also known as a passive mode – to the network traffic created by the endpoint devices. It is necessary to install agents on the network switches. In this test, the Cisco Catalyst 9000 series switches were used due to their ability to run the Docker container agents. These agents perform a deep packet inspection of the traffic and send the relevant metadata to the management console. It is later consolidated, stored, and analyzed in more detail depending on the configuration and user's use. Figure 4.2 illustrates the architecture of the solutions used.

4.2.2 Setup

One of the initial decisions that needed to be made was, "Which switches to install the agents on?" This decision is because the company's network architecture follows the best practice of "Hierarchical Network Design" [Cisco, 2014] – also sometimes referred to as the "Hierarchical internetwork model". A LAN network is organized into three layers following this best practice, as illustrated in figure J.1. The first, the "Access layer", has the network switches that provide access – and physically connect – to the endpoint devices. Many network ports characterize these switches to connect as many devices as possible. They also serve several other functionalities, such as layer two switching, port security, QoS

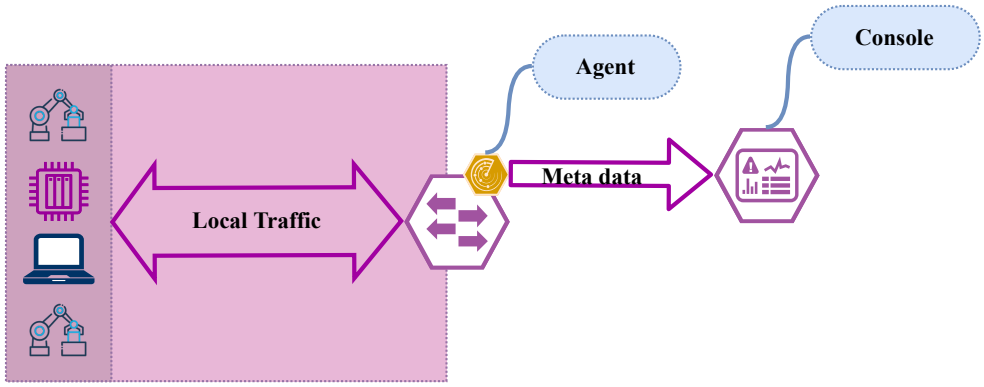


Figure 4.2: Security monitoring tools architecture

classification, and packet marking. The second layer, called the distribution layer, aggregates the traffic from the access layer before transmitting it to the final layer, called the core layer. This middle layer provides features such as LAN link aggregation, security through policies implemented by Access Control Lists, redundancy, and load balancing. Finally, the core layer comprises high-performance switches with redundancy and resiliency capabilities. This layer is responsible for the traffic exchange between, for example, several campuses or sites.

The best place to perform the agents' installation choice is to balance product licensing, the granularity of the data flows transported, and network device coverage by the switch. The preferred choice is always between maximizing the observation of data flows by installing agents on the access switches or achieving a similar – if theoretically smaller – reach with fewer agents by installing them on the distribution switches – maximizing the device coverage. Moreover, reduce the amount of data generated by the set of agents and the processing resources of the switches needed to run them. An example to better describe this choice is; taking two devices connected to the same switch and only exchanging information with each other. Only one agent installed on an access switch can catch the data stream. However, if we take the example of two devices on two different networks and switches, all observable traffic between them- i.e., layer 3 – passes through this distribution layer.
For this proof of value, it was decided to install the agents on the access switches, maximizing the number of data flow to better gauge the capabilities of the tools.

Furthermore, after installing the software, it was also necessary to perform initial settings, such as the name and IP addresses of other networks with which the devices most frequently communicate. This additional information improves the resulting reports, enriching the network context, for example, when viewing network traffic patterns.

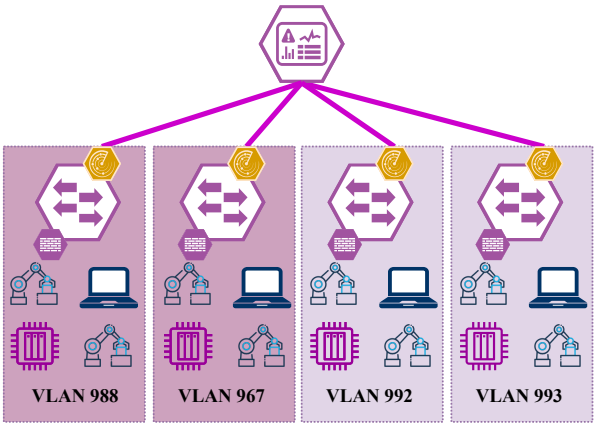


Figure 4.3: Proof of value network architecture

4.2.2.1 Network deployment

As shown in figure 4.3, the network scope was 4 VLANs that contain the existing devices in the production area. Exactly how many and what type of endpoints were unknown at the beginning. It was also expected to identify other devices related to facility management. These device types are because systems related to environmental conditions – e.g., clean rooms or temperature and humidity-controlled areas – are essential to the production process's quality.

Because it is information related to a sensitive area to the company, the third octet of the IP network address has been anonymized and replaced by a letter. However, this fact does not affect the validity of the data shown in this document.
The VLANs and their respective IP ranges are mentioned in the 4.2 table. The nonusage of DHCP, and therefore the manual assignment, was essential because it allows the IP address attribute to be used as an identifier for comparison.

VLAN	IP range	VLAN	IP range
988	10.15.A.0/24	992	10.15.B.0/24
967	10.15.C.0/25	993	10.14.D.0/23

Table 4.2: VLAN in scope

4.2.3 Asset identification

The first security measure that has to be checked is asset identification.
Managing assets or devices in a production area is often a manual process and is maintained irregularly over time. As it is easy to understand, maintaining such a register is not the top operational priority task performed by the production area's responsible staff. Moreover, an asset record should be as up-to-date

as possible and correspond to the device's life cycle, from its addition on the shop floor to its removal.

Asset management is essential for information security. Only by knowing what to protect can the proper additional measures be deployed or have the ability to identify rogue devices that should not be on the network along with the rest of the devices.

A tool that helps to automate the identification of new devices, and reduce the effort of human intervention in keeping an updated asset record, should be seen as the first security measure to be implemented. Fortunately, there was a partial and manual kept asset record – for VLAN 988 and 967 – of the connected devices in the production area under analysis. The two remaining VLANs do not have a log because they are managed by a different team and are related to building management – i.e., indirectly related to the production area. Having a record is sufficient to validate the security monitoring tool feature.

The complete initial asset record is shown in appendix K table K.1 and consists of 168 devices. As shown in table 4.3, after running the Cisco Cyber Vision tool, it can be seen that 113 devices have been identified – using the same IP address. Compared to the initial asset register, this value corresponds to 67%.

Because of the time interval between the two tools – 1.5 months plus two months for implementing the Nozomi Guardian tool, the asset record was requested again, as shown in table K.2 containing 148 devices. After 1.5 months of running Nozomi Guardian, 88 devices were identified, corresponding to 59%. It is also important to note that both tools identified eight new devices, not in the asset inventory.

Security tool	Assets in asset list	Assets found	Assets found (%)	New assets found
Cisco Cyber Vision	168	113	67	8
Nozomi Guardian	148	88	59	8

Table 4.3: Asset identification results

The value of the percentage of devices found by both tools is lower than expected – 67% for Cisco Cyber Vision and 59% for Nozomi Guardian. Because the values were not expected, a more careful analysis was performed by comparing the two inventory versions and exchanging information with the production managers. The six devices shown in table 4.4 were identified as still being in the asset list, although they are decommissioned. Furthermore, five devices shown in table 4.5 were temporarily disconnected or disabled before the Cisco Cyber Vision run and reconnected during the Nozomi Guardian run.

Although it is not possible to confirm why the remaining devices in the inventory are not identified, it can be stated that the manual updating process is subject to human error and not as frequent as one might wish. Moreover, to make this fact more evident, table 4.6 shows the identified new devices by each tool (Cisco for "Cisco Cyber Vision" and Nozomi for "Nozomi Guardian"), not in the asset inventory, at any moment.

In conclusion, it can be said that although not entirely accurate, both tools offer essential help in

keeping track of assets in a production area. They allow automation of the process, thus relieving human intervention. Human intervention can then focus on removing and verifying devices from the inventory.

Name	IP
Y100AC06	10.15.A.7
Y100POR030	10.15.A.84
Y1NCM032	10.15.A.157
Y100AJ29	10.15.A.187
Y100AJ30	10.15.A.188
Y100AJ31	10.15.A.189

Table 4.4: Device later removed from the asset list

Name	IP
Y100AJ57	10.15.A.15
Y100AD48	10.15.A.59
Y100AD71	10.15.A.88
Y100AD64	10.15.A.89
Y100AG52	10.15.A.117

Table 4.5: Assets disabled during Cisco Cyber Vision but enable during Nozomi Guardian

New device IP	Cisco	Nozomi	New device IP	Cisco	Nozomi
10.15.A.100	•		10.15.A.23		•
10.15.A.123		•	10.15.A.34	•	
10.15.A.126	•	•	10.15.A.41		•
10.15.A.129	•		10.15.A.69		•
10.15.A.133	•		10.15.A.77	•	•
10.15.A.144		•	10.15.C.11	•	
10.15.A.145		•	10.15.C.21	•	

Table 4.6: New devices identified by both tools not in the asset lists

4.2.4 Device vulnerability

Identifying publicly known vulnerabilities in end devices is an essential preventive mitigation measure for the system's overall security. In a production area where the ability to intervene in devices is limited, it is even more important to be aware of the vulnerabilities. This knowledge makes it possible to prioritize which devices are the most critical and plan measures that prevent the exploitation of threats (e.g., patching, disabling unnecessary services or software, or changing settings).

The two analyzed tools can infer vulnerabilities through the network traffic generated by the devices. Even though it is possible to argue that this inference is far from being exact and generating false positives, it is nevertheless the possible way to ensure that no interaction is carried out through the network with the end devices. Because, they have a limited processing capacity and may see their operation affected by a network traffic overload.

As shown in the table 4.7, the Cisco Cyber Vision tool identified 2260 vulnerabilities, of which 39

were distinct, in 110 devices. Furthermore, Nozomi Guardian identified 36569 vulnerabilities, of which 4435 were distinct, in 36 devices. The total number of vulnerabilities is the sum of all vulnerabilities identified on all devices. In contrast, the unique vulnerabilities are the different types, i.e., CVE identifiers, independent of the associated devices.

The big difference in the number of vulnerabilities identified between the two solutions is evident. After a closer analysis and discussion with the two manufacturers to understand the cause of this difference, the best conclusion is how the vulnerability inference is performed. In the Cisco Cyber Vision tool, only vulnerabilities with a high probability of actually existing are shown, thus reducing the number of false positives. The low device and vulnerability number are related to an optional product feature called "Active Discovery"[5], which has been disabled so that there is no possibility of interaction with the devices in the production area. While in Nozomi Guardian, all possible vulnerabilities for the identified firmware or operating system version are shown.

To accomplish this work, it was impossible to verify the actual existence of the vulnerabilities because it is not authorized to perform a vulnerability network scan – i.e., using Nessus or Rapid7 InsightVM, which is the tool used in the company, or do a penetration test. It is only possible to perform a TCP open port sweep to minimal ports, which only allows identifying which devices are connected but does not help verify vulnerabilities.
The difference in the number of devices with vulnerabilities is also significant, and the justification lies in the device quality information of each tool, especially in identifying the firmware or operating system.

In order to show greater detail, in table L.1, it is possible to see the number of vulnerabilities shown by device for each tool. To better compare the two solutions, a PLC was chosen. It is an ICS device distinct from a Windows one and frequent on the shopfloor – i.e., of OT technology – to show which vulnerabilities are identified by each tool. The detail is shown in table L.2.

Tool	Devices with vulnerabilities	CVE	Unique CVE
Cisco Cyber Vision	110	2260	39
Nozomi Guardian	36	36569	4435

Table 4.7: Device vulnerabilities overview

4.2.5 Thread or Intrusion Detection

Rapid detection and response to security incidents are critical to mitigate their impact. Furthermore, having a tool that does this is a significant benefit to any organization.

[5] According to the user guide, *"Active Discovery is a feature to enforce data enrichment on the network. As opposed to passive traffic (...) Active Discovery is an optional feature that explores traffic in an active way.(...) Moreover, some information like firmware version can be difficult to obtain because they are not exchanged often between components." Currently Active Discovery supports three broadcast protocols (EtherNet/IP (Rockwell), and Profinet and S7 Discovery (Siemens)."*

However, it is essential to start by making a distinction between "incident" and "event". Their definition can be found in the ISO 27000 standard [for Standardization, 2018a], which defines an information security event as *"identified occurrence of a system, service or network state indicating a possible breach of information security (3.28) policy (3.53) or failure of controls (3.14), or a previously unknown situation that can be security relevant"*. And information security incident as a *"single or a series of unwanted or unexpected information security events (3.30) that have a significant probability of compromising business operations and threatening information security (3.28)"*. It is possible to summarize by saying that an event is a change to the state of a system, and a security incident is an event that impacts the confidentiality, integrity, or availability properties of the same system.

In this work, it was intended to analyze the effectiveness of these tools in detecting security events. However, it is essential not to forget the context in which this analysis was performed. By using an actual production area, we have the benefit of observing the behavior of the tools as close as possible to reality. However, this also means that carrying out a penetration test to perform intrusion simulation is impossible, thus limiting the data available on the events identified by the tools to the results from the regular production area operation. Moreover, these reasons did not allow for performing an analysis with the intended detail and depth that this work deserves.

Since it was impossible to perform simulations, the occurrences of Port Scan events performed by the company's existing tool were used. As a port scan is an initial step often performed by an adversary to do some reconnaissance of the network and identify existing devices, it is possible to use these events as indicators of potential threats. Table 4.8 shows the days of the detected occurrences in the respective complete months in which the tests were performed. The M.1 and M.2 figure show an example of a generated event for each tool.

Tool	**Mo**	**Tu**	**Th**	**Mo**	**Th**	**Sa**	**Mo**	**Tu**	**Th**	**Mo**	**We**	**Th**	**Mo**
Cisco – Nov. 2021	1		4	8	11		15	16	18	22	24	25	29
Nozomi – Mar. 2022		8	10		17	19		22	24			31	

Table 4.8: Day of the month with port scan occurrences

4.3 Summary

Implementing a tool like the one shown here can be a security measure to mitigate the threats mentioned in the previous chapter effectively. However, they should not be considered a solution that completely fulfills all requirements. As has been shown, it cannot offer the guarantee of identifying all devices. Nor can it handle the case of asset removal from the inventory. Moreover, these tools cannot assign the criticality property of an asset. For all these reasons, human intervention is still necessary, thus dispelling the idea that technology is sufficient to make a production area completely secure.

Regarding the vulnerability identification and threat monitoring functionality, creating a simulated

production area would have significantly benefited from obtaining more factual data to confirm the data presented by the tools. With such an environment, there would be the liberty to perform more in-depth tests (e.g., vulnerability scans, penetration tests, or configuration changes) without the impact limitation on the production processes.

The tools themselves also require constant tuning to the specific production area. This change of their configurations is laborious and time-consuming, which was impossible to accomplish during this work. Examples are the anomaly detection feature and the definition of the most relevant events so as not to flood the users or other downstream systems (e.g., SIEM) with inconsequential information.

Because of its high vulnerability, false positives number and generated events, it remains to be proven that a tool that does not depend on agents in the devices and infers all the information by the network traffic can solely prevent all the threats identified in this work.

After this chapter, it is possible to answer the second question of this work, *"Can a system with the threefold functionality of identifying the devices and their vulnerabilities, real-time monitoring of data flow on the network, and alerting to security events be an appropriate security measure to mitigate the threats identified?"*. As shown, these tools, even if not exact, can significantly increase the knowledge of a production environment, starting with identifying the devices that compose it and their vulnerabilities and detecting threats in real time. Therefore, it is then possible to conclude that a system implemented by these tools is an appropriate security measure to mitigate the identified threats.

Chapter 5

Recommendations

5.1 Recommendations

Throughout this study, information and lessons were collected, forming a set of recommendations that should help an actual application. These recommendations result from observing the company reality in which the work was carried out and interactions with the company's stakeholders.

In order to structure these recommendations, the categories of the NIST CSF framework mentioned in appendix B were used. As this work focuses on the first three framework categories, the recommendations are also related to Identify, Protect and Detect categories.

5.1.1 Identify

The NIST CSF "Identify" function is directly related to the work carried out in this study. As mentioned in chapter 3, modeling the system in a threat model requires analyzing it and identifying its sub-components, people who use it, or even its capabilities. In chapter 4 the security solutions analyzed also aimed to contribute to a better understanding of the production environment with the identification of networked devices.

The "Identify" function sets the foundation for the other functions ahead. The table 5.1 highlights the recommendations regarding the "Identify" function recognized during this work.

Subcategory	Recommendation
ID.GV-3 Legal and regulatory requirements regarding cybersecurity, including privacy and civil liberties obligations, are understood and managed	The information security topic of a country's critical infrastructure has been discussed even more recently. In Europe, beginning with several countries' requirements, for example, the publication in August 2021 in Germany of the so-called KritisV2 law – *"Kritische Infrastrukturen"* – consists of a complete set of laws and regulations that have to be adopted. Alternatively, the approval in May 2022 of the European directive NIS2 (Network and Information Security Directive) and the RCE (Resilience of Critical Entities) define a framework to protect critical infrastructures at the European level. NIS2 extends the scope and requirements for organizations, and the RCE prescribes resilience and continuity measures and will replace the current ECI (European Critical Infrastructures) directive. Both NIS2 and RCE will have to be transposed into national law after a final agreement by the European Commission and parliament. It is also recommended not to forget that a crucial source of threats for a production area is related to the privacy of data or patient information, where applicable. There are several regulations that organizations are required to comply with. In Europe, there is the GDPR (General Data Protection Regulation), China has the recent MLPS (Multi-Level Protection Schema), and the US has the HIPPA (Health Insurance Portability and Accountability Act).
ID.GV-4 Governance and risk management processes address cybersecurity risks	Companies should align the implementation and documentation of the measures resulting from the threat model with standards such as ISO27001 and IEC/ISA 62443. Because organizations simultaneously intend to implement these standards and disclose their certification to assure their stakeholders that information security is a priority for them.
ID.RA-2 Cyber threat intelligence is received from information sharing forums and sources	Due to recent socio-political events, more work has been done and published on developing frameworks for industrial and manufacturing areas. Entities such as ENISA in Europe and some national entities such as the German BSI[1] or CISA and NIST in the US have published reports with the most recent threats observed and the measures advised to mitigate them. Because information security is constantly changing, a permanent review of these sources of information is essential. This volatility is also the main factor that should trigger the organizations' risk reevaluation and, therefore, the needed periodic reassessment of the threat models.

[1] *"Bundesamt für Sicherheit in der Informationstechnik"* German federal cybersec. auth. - https://www.bsi.bund.de

Subcategory	Recommendation
ID.RA-3 Threats, both internal and external, are identified and documented	To aid in the initial discussion of threats, it is possible and recommended to use a predefined set of threats serving as a baseline for further definition of specific threats. The threats that are part of this work in appendix D provide a good starting point.
ID.RA-4 Potential business impacts and likelihoods are identified	Although all production areas have several factors in common that, in a more general case, allow us to draw the conclusions mentioned in this work, their implementation is always different and directly related to the actual manufacturing processes. This reason is why creating a comprehensive team to identify threats is recommended in realizing the threat model. Such a team must be composed of a mix of security specialists who understand the specifics of a production area and its technologies and a set of manufacturing process experts with knowledge of the various transformation stages. In a nutshell, it is possible to reason that security specialists have a more nuanced understanding of assessing the likelihood of a threat. At the same time, manufacturing process experts are better tuned in the threat impact assessment of the system. These two sides of the equation complement each other to get a more accurate and less abstract risk evaluation of the associated threats.
ID.RM-1 Risk management processes are established, managed, and agreed to by organizational stakeholders	An attack tree creation, while seemingly a non-essential step, is an excellent way to graphically show the true potential of threats to achieve a larger goal. This graphical way of representing threats is a perfect tool to illustrate and explain the real threat to an organization's decision-makers, that are often less security savvy. After all, it is undoubtedly up to them to enable future resource allocation – time, money, or human resources – to implement security measures.

Subcategory	Recommendation
ID.SC-3 Contracts with suppliers and third-party partners are used to implement appropriate measures designed to meet the objectives of an organization's cybersecurity program and Cyber Supply Chain Risk Management Plan	The supply chain is an essential topic for organizations because it is a source of security threats. As such, the organization must consider the security requirements its suppliers must fulfill, for example, in the production machines or devices acquired. Moreover, these requirements must be shown clearly to the suppliers at the beginning of the procurement process - e.g., through a specific checklist.
ID.SC-4 Suppliers and third-party partners are routinely assessed using audits, test results, or other forms of evaluations to confirm they are meeting their contractual obligations	The maturity of a supplier's ISMS (Information Security management System) directly impacts its customers. The evidence is that during this work, the production area was subject to audits carried out by customers. The existence of a security solution for the same threats identified in this work was topic discussed within it.

Table 5.1: Identify category recommendations

5.1.2 Protect

As the "Protect" function is one of the domains addressed in this study, it was necessary not to forget the impact that security measures have on the organization and how the entire company plays an active role in its implementation. Table 5.2 highlights the recommendation related to the "Protect" function identified throughout this work that results from the proof of concept carried out in the company.

Subcategory	Recommendation
PR.IP-9 Response plans (Incident Response and Business Continuity) and recovery plans (Incident Recovery and Disaster Recovery) are in place and managed	Organizations should understand that the implementation and operation of a security monitoring solution must be accompanied by the definition of a strategy for responding to incidents. For this, it is recommended to involve both operational teams on the security side (i.e., SOC) and production areas to elaborate action plans that dictate what steps need to be taken, also called playbooks. Because SOC teams often do not have the ability or autonomy to interact with endpoint devices, these plans have to involve the shop floor teams and assign them the responsibility for quick action for incident analysis and troubleshooting.

Table 5.2: Protect category recommendations

5.1.3 Detect

The "Detect" function is also directly related to the work done in this study. In particular with the solutions analyzed in chapter 4.1 to detect device vulnerabilities and ongoing threats. Table 5.3 highlights the recommendation related to the "Detect" function identified throughout this work that results from the proof of concept carried out in the company.

Subcategory	Recommendation
DE.AE-2 Detected events are analyzed to understand attack targets and methods	A significant effort must be made to adjust these tools to specific production areas. Furthermore, this effort must be supported by the responsible production area operation team and not only by the organization's security team. Because the first is responsible for the shop-floor operation, it is their job to perform the final analysis, evaluate the security events, and distinguish the false positives from the relevant events. The same is true during the remaining life cycle of the tools, not only after their implementation. In short, this means that the greater visibility these tools bring implies an increase in the effort required from operational teams, which are often already overloaded. It is vital to highlight this fact to the decision-makers of an organization so that they budget the resources necessary for the success of the security measures.
DE.DP-1 Roles and responsibilities for detection are well defined to ensure accountability	

Table 5.3: Detect category recommendations

5.2 Summary

This chapter is not intended to be an exhaustive checklist of recommendations for applying the NIST Cyber Security Framework. However, it highlights aspects within this work scope that should be kept in mind by those planning to apply it to other organizations.

Chapter 6

Conclusion

After the experimentation and analysis of the mitigating actions in the previous chapter recommended by the proposed four-step framework, which has been used as the guideline for this work, it is now possible to conclude and discuss future work. Nevertheless, to better solidify these conclusions, it is essential to look at the identified threats and validate that they can be mitigated with the tools used. The 6.1 table shows how the threats have been mitigated.

ID	Description	Observed results
T01	Attackers gain access to the system or unauthorised data exploiting a known vulnerability (e.g. missing patch)	Potential device vulnerabilities are identified.
T03	Attackers try to retrieve banner information through the open ports to discover potential vulnerabilities	Port scans were identified.
T06	An adversary can plant a rogue device to perform man-in-the-middle attacks	It was shown that keeping an updated inventory of assets is possible, thus helping identify rogue devices. Additionally, as shown in the image N.1, events are generated when new devices are identified.

Table 6.1: Threats mitigated based on the tool observation

The threats shown in table 6.2 below could not be tested in an actual and running production area, where the possible actions were limited.

ID	Description
T04	An adversary can bruteforce authentication credentials
T05	An adversary guesses the default password and user of an authenticator (e.g. admin:admin)
T09	An adversary can execute remote code
T10	An adversary can perform a DoS attack by making consecutive authorization requests

Table 6.2: Threats that need further investigation

Like the previous, the threats shown in table 6.3 require more experimentation and would benefit from further actions such as penetration testing. However, strong evidence was observed that the tools could mitigate them.

ID	Description	Observed results
T02	Attackers gain access to the system exploiting insufficient or misconfigured security features	Events like those in the figure N.2 are generated, possibly aiding in the detection of settings misusage.
T07	Attackers could gain access to sensitive data through a man-in-the-middle attack	Events like those in the figure N.3 are generated, possibly aiding in the detection of MITM.

Table 6.3: Threats with evidence that can be used to mitigate them

Finally, the "T08 An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)" threat is outside the tools' scope. However, it could hypothetically be possible to identify network traffic that the malware generates.

Based on this work, it is now possible to conclude that despite the specifics and constraints of a production area, performing a threat model using the STRIDE methodologies for threat classification and the DREAD methodology for its risk assessment produces a valid set of threats. Furthermore, from there, the measures or controls that are best suited can be emanated. It is also possible to state that by using solutions similar to the ones analyzed, with the ability to identify the devices and their vulnerabilities and real-time monitoring of data flow on the network, and alerting to security events, it is possible to mitigate threats. Resulting in a security maturity increase of a production area, therefore obtaining an increase in its resilience against constantly emerging threats.

Clearly, answering the first question of the objectives of this work, *"With the many restrictions and particularities existing in the production areas and applying a threat model with the most appropriate methodologies, what are the most adequate controls or measures to implement?"*, the threat model produced a set of valid threats, which when matched with the measures mentioned in the related work, allows us to identify the most appropriate mitigating measures – table 3.22.

To answer the second question of this work, *"Can a system with the threefold functionality of identifying the devices and their vulnerabilities, real-time monitoring of data flow on the network, and alerting to security events be an appropriate security measure to mitigate the threats identified?"*, these tools, can significantly increase the knowledge of a production environment, starting with identifying the devices that compose it and their vulnerabilities and detecting threats in real-time. Based on their results observation and study, it is then possible to conclude that a system implemented by these tools is an appropriate security measure to mitigate the threats identified.

6.1 Future Work

Despite the conclusions of this work, further studies and investigations can be pursued.

As already mentioned, it was evident that it was impossible to do all the experimentation with total freedom in an actual production area. As such, one of the future works would be creating a simulated production area environment to perform hacking activities on the devices to validate the events generated. Although this work was done side-by-side, it explored ways to perform this task using a network simulator (e.g., GNS3[1]) and virtual machines to install the tested solutions. However, due to time limitations was not possible to identify an efficient way to simulate the various specific devices of a production area. For example, production machines or how to connect software that simulates PLCs to this simulated environment described. Nor was it possible to identify how to install the agents – dockers containers – on the simulated switches – in GNS3. For these reasons, it seems almost preferable, if it were feasible – to use real devices that are not in use.

Another future research path would be the interconnection with other systems like SIEM. Where for example, the creation of a *playbook*[2] set aligned simultaneously by the SOC[3] and production area managers. This research would allow leveraging a more significant benefit of threat detection. Eventual automation by a SOAR[4] system would close the loop on incident management. Similarly, developing interfaces for loading asset inventory using the available APIs (Application Programming Interface) would allow an essential part of asset management to be accomplished.

The reporting topic would also be another idea for future analysis. For example, exporting collected information to create dashboards.

Regarding the threat model, it would be interesting to investigate or compare the effectiveness of other methodologies and increase the scope of production areas to model a larger and more complex system. As well as expanding the team participating in the discussions to identify and assess the threats

[1] GNS3 (Graphical Network Simulator-3) is a network software emulator first released in 2008. It allows the combination of virtual and real devices to simulate complex networks. https://www.gns3.com/

[2] An all-encompassing, organization-wide manual that dictates precisely what actions to take when an security incident occurs.

[3] An SOC platform aims to provide security incident detection and response services.

[4] A SOAR system (Security Orchestration, Automation, and Response) automates the investigation via workflow automation of a playbook.

to gauge how different points of view would result in entirely different threats than those utilized for this work.

Tools analyses that assist in identifying threats using a DFD diagram are also presented as an area for further work. For an organization, using these tools would allow for more scalable processes for the widespread adoption of the application and revision of threat models, with the consequent possible simplified reuse of past threats and information.

References

Asghar, M. R., Hu, Q., and Zeadally, S. (2019). Cybersecurity in industrial control systems: Issues, technologies, and challenges. *Computer Networks*, 165. 9, 10

Barkalov, A., Titarenko, L., and Mazurkiewicz, M. (2019). *Programmable Logic Controllers*, volume 195. Newnes, Oxford, sixth edition edition. 1

Bhamare, D., Zolanvari, M., Erbad, A., Jain, R., Khan, K., and Meskin, N. (2020). Cybersecurity for industrial control systems: A survey. *Computers and Security*, 89. 8, 10

Braiterman, Z., Shostack, A., Marcil, J., de Vries, S., Michlin, I., Wuyts, K., Hurlbut, R., Schoenfield, B. S., Scott, F., Coles, M., Romeo, C., Miller, A., Tarandach, I., Douglen, A., and French, M. (2021). Threat modeling manifesto. 18

Cisco (2014). *Cisco Networking Academy Connecting Networks Companion Guide: Hierarchical Network Design*. Cisco Press. 63

Danielis, P., Beckmann, M., and Skodzik, J. (2020). An iso-compliant test procedure for technical risk analyses of iot systems based on stride. In *An ISO-Compliant Test Procedure for Technical Risk Analyses of IoT Systems Based on STRIDE*. 6

for Standardization, I. O. (2014). Iso 27001 information security management systems – requirements. Technical report, International Organization for Standardization. 22, 23

for Standardization, I. O. (2018a). Iso 27000 information security management systems — overview and vocabulary. Technical report, International Organization for Standardization. 69

for Standardization, I. O. (2018b). Iso 27005 information security management systems – information security risk management. Technical report, International Organization for Standardization. 5

for Standardization, I. O. (2018c). Iso 31000 risk management - principles and guidelines. Technical report, International Organization for Standardization. 22

Hollerer, S., Kastner, W., and Sauter, T. (2021). Towards a threat modeling approach addressing security and safety in ot environments. *IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS*, 2021-June. 7

Khan, R., McLaughlin, K., Laverty, D., and Sezer, S. (2017). Stride-based threat modeling for cyber-physical systems. In *STRIDE-based threat modeling for cyber-physical systems*, volume 2018-January, pages 1–6. Institute of Electrical and Electronics Engineers Inc. 7

Kohnfelder, L. and Garg, P. (1999). The threats to our products. *Microsoft Security Development Blog*, 8:1–1. 27

Kruchten, P. (1995). Architectural blueprints—the ”4+1”; view model of software architecture. *IEEE Software*, 12:1–15. 24

Kure, H. I., Islam, S., and Razzaque, M. A. (2018). An integrated cyber security risk management approach for a cyber-physical system. *Applied Sciences (Switzerland)*, 8. 8, 10

Lella, I., Theocharidou, M., Tsekmezoglou, E., Malatras, A., García, S., Valeros, V., and for Cybersecurity., E. U. A. (2021). Enisa threat landscape for supply chain attacks. *ENISA Publication*. 8, 10

Makhdoom, I., Abolhasan, M., Lipman, J., Liu, R. P., and Ni, W. (2019). Anatomy of threats to the internet of things. *IEEE Communications Surveys and Tutorials*, 21:1636–1675. 6, 10

Malatras, A., Skouloudi, C., and Koukounas, A. (2019). Enisa industry 4.0 cybersecurity: Challenges & recommendations. *ENISA Publication*. 9, 10

Mashkina, I. and Garipov, I. (2018). Threats modeling and quantitative risk analysis in industrial control systems. *2018 International Russian Automation Conference, RusAutoCon 2018*. Mashkina I and Garipov I, make a brief focusing on the attack vectors that an ICS system can be targeted. The most interesting part is the way they model these threats through a cognitive map (as the authors call it) where it is possible to visualise the path an attacker would have to follow in order to gain access to the various components of the system. Clearly showing that there is a big difference in the risk of an attack being carried out by an adversary with access to the LAN network or with access to the ICS systems (insider threat) of another with remote access. However it is not clear how the cascading effect can affect the various components. 5, 10

Messe, N., Chiprianov, V., Belloir, N., El-Hachem, J., Fleurquin, R., and Sadou, S. (2020). Asset-oriented threat modeling. In *Asset-oriented threat modeling*. 7

Moustafa, N., Adi, E., Turnbull, B., and Hu, J. (2018). A new threat intelligence scheme for safeguarding industry 4.0 systems. *IEEE Access*, 6. 7, 10

Nweke, L. O. and Wolthusen, S. D. (2020). A review of asset-centric threat modelling approaches. *International Journal of Advanced Computer Science and Applications*, pages 1–6. 6

Saltzer, J. H. and Kaashoek, M. F. (2009). *Principles of Computer System Design*. Elsevier, Burlington. 17

Shostack, A. (2014). *Threat Modeling Designing for Security*, volume 53. Wiley, Boulevard Indianapolis,. III, 5, 20, 23, 25, 27, 47

Tatam, M., Shanmugam, B., Azam, S., and Kannoorpatti, K. (2021). A review of threat modelling approaches for apt-style attacks. *Heliyon*, 7. 6, 10

Tuma, K., Sandberg, C., Thorsson, U., Widman, M., Herpel, T., and Scandariato, R. (2021). Finding security threats that matter: Two industrial case studies. *Journal of Systems and Software*, 179. 6

Williams, T. J. (1994). The purdue enterprise reference architecture and methodology (pera). *Computers in Industry*, 24. 33

Xiong, W. and Lagerström, R. (2019). Threat modeling – a systematic literature review. In *Threat modeling – A systematic literature review*, volume 84, pages 53–69. Elsevier Ltd. 7

Yeboah-Ofori, A. and Islam, S. (2019). Cyber security threat modeling for supply chain organizational environments. *Future Internet*, 11. 8, 10

Yourdon, E. and Constantine, L. L. (1979). *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*, volume 1. Prentice-Hall, Inc., USA, 1st edition. 26

Zografopoulos, I., Ospina, J., Liu, X., and Konstantinou, C. (2021). Cyber-physical energy systems security: Threat modeling, risk assessment, resources, metrics, and case studies. *IEEE Access*, 9:29775–29818. 6, 10

Appendix A

Modern DFD model example

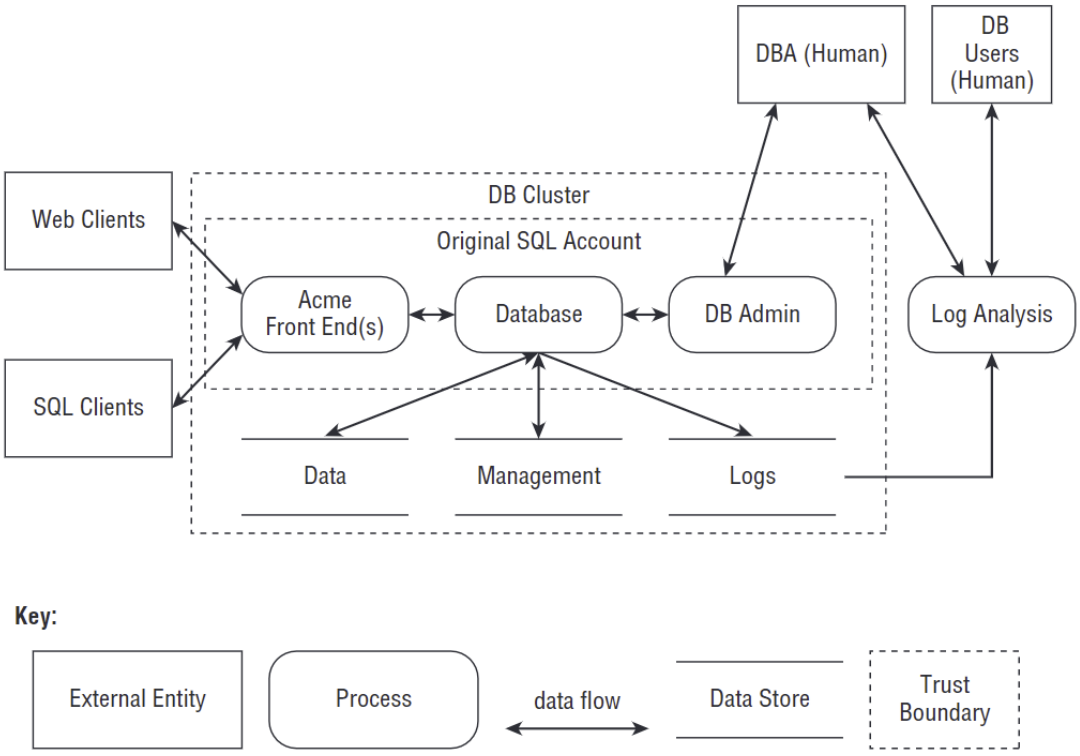


Figure A.1: A modern DFD model

Appendix B

NIST CSF Core Functions

Function Unique Identifier	Function	Category Unique Identifier	Category
ID	Identify	ID.AM	Asset Management
		ID.BE	Business Environment
		ID.GV	Governance
		ID.RA	Risk Assessment
		ID.RM	Risk Management Strategy
		ID.SC	Supply Chain Risk Management
PR	Protect	PR.AC	Identity Management and Access Control
		PR.AT	Awareness and Training
		PR.DS	Data Security
		PR.IP	Information Protection Processes and Procedures
		PR.MA	Maintenance
		PR.PT	Protective Technology
DE	Detect	DE.AE	Anomalies and Events
		DE.CM	Security Continuous Monitoring
		DE.DP	Detection Processes
RS	Respond	RS.RP	Response Planning
		RS.CO	Communications
		RS.AN	Analysis
		RS.MI	Mitigation
		RS.IM	Improvements
RC	Recover	RC.RP	Recovery Planning
		RC.IM	Improvements
		RC.CO	Communications

Figure B.1: NIST Cyber Security Framework Core Functions

Appendix C

Production areas observed and examples

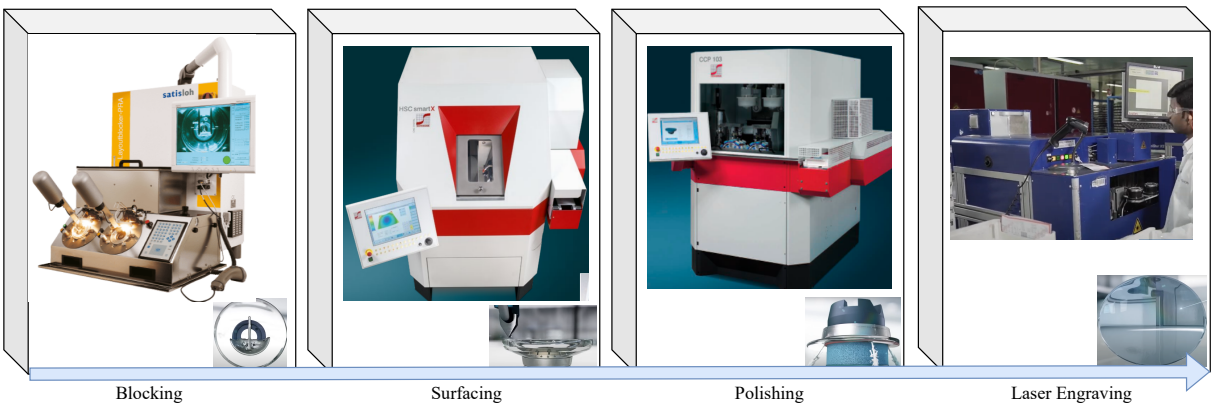
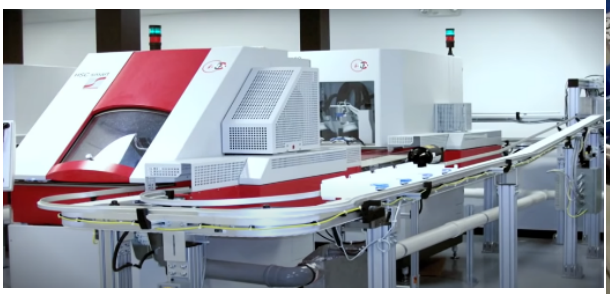


Figure C.1: Eye glasses lens production process example



(a) Machines inline

(b) Production trays

Figure C.2: Eye glasses lens production

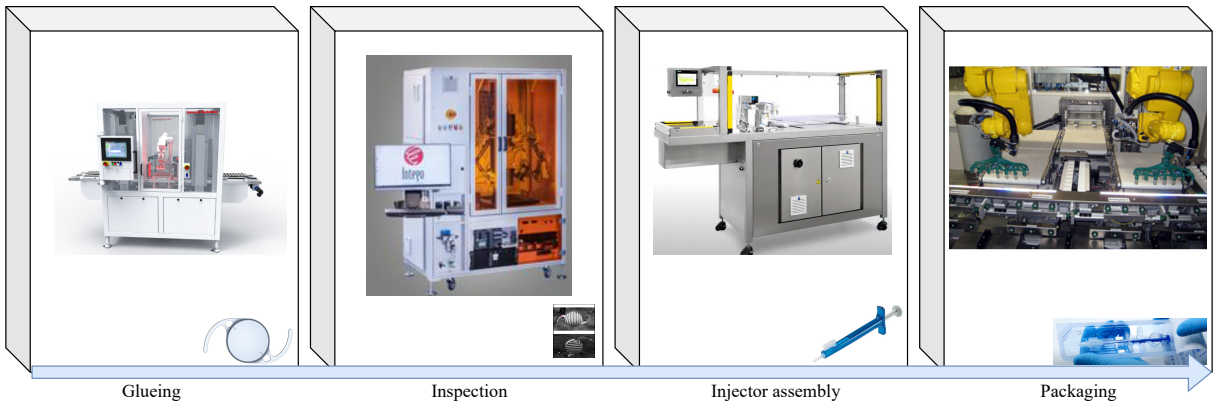


Figure C.3: IOL lens production machines example

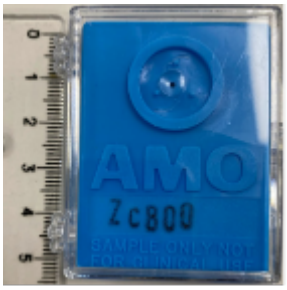


Figure C.4: IOL lens

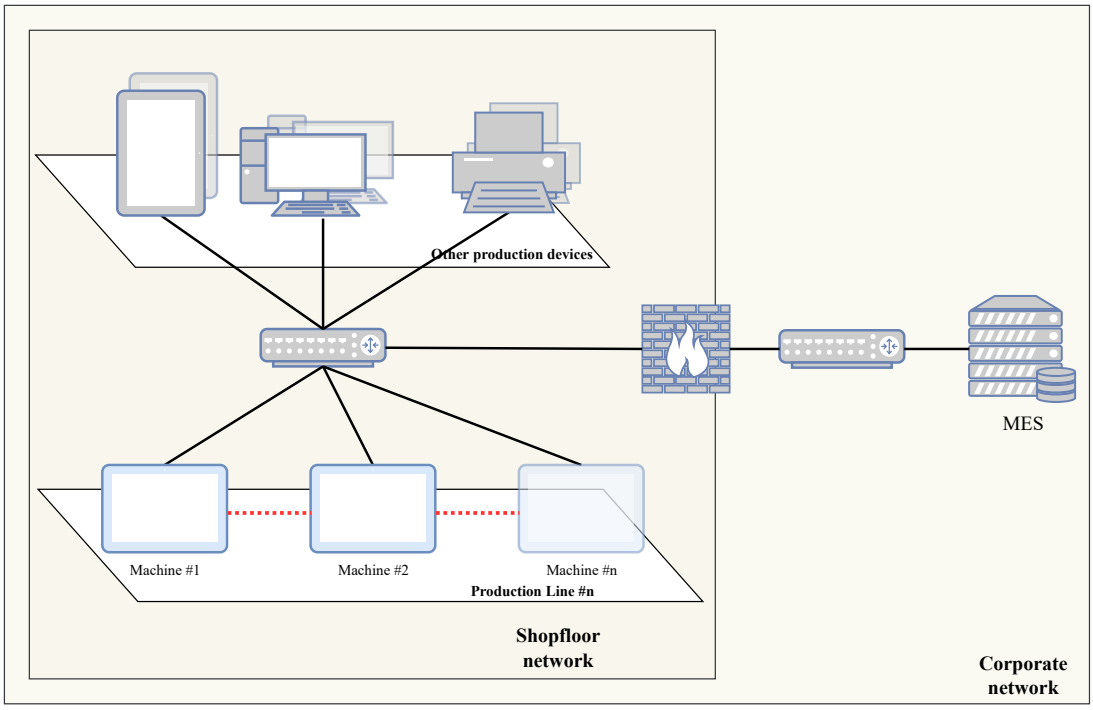


Figure C.5: Production area network connections

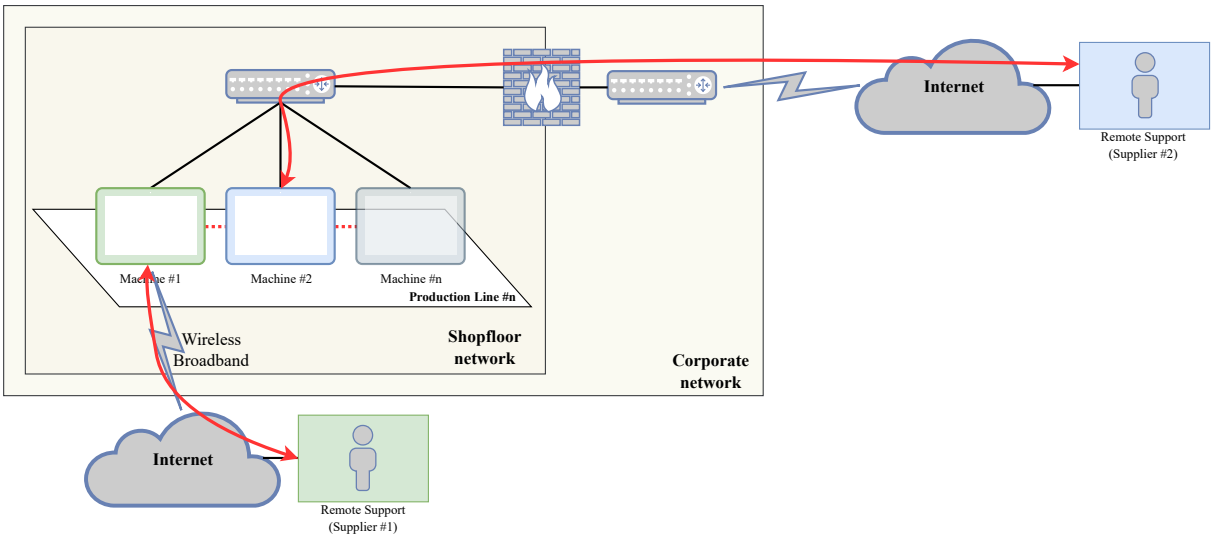


Figure C.6: Production area remote access

Appendix D

Threat full list

- Attackers gain access to the system or unauthorised data exploiting a known vulnerability (e.g. missing patch)
- An adversary receives authenticator feedback which helps him in user enumeration. E.g. invalid username. Invalid password
- Attackers try to retrieve banner information through the open ports to discover potential vulnerabilities
- Adversaries who compromised the asset can dispute there was no message stating access was only for authorized personnel
- An adversary can bruteforce authentication credentials
- A rogue employee has unfettered access to a key and might use it for a malicious purpose or pass it onto someone else to the same end
- An adversary guesses the default password and user of an authenticator (e.g. admin:admin)
- An adversary can plant a rogue device to perform man-in-the-middle attacks
- Attackers could gain access to sensitive data through a man-in-the-middle attack
- An adversary that has previously obtained unauthorized access to certain device resources, uses that access to obtain information such as location and network information
- An adversary could have planted a malware or backdoor on the asset without the user ever knowing (no anti-virus)
- An adversary is able to retrieve unencrypted passwords after a DBMS/DB -server breach

- An adversary can execute remote code
- An adversary is able to guess the password based on identifying an authenticator
- An attacker generates a message or datablock that causes the recipient to believe that the message or datablock was generated and cryptographically signed by an authoritative or reputable source, misleading a victim or victim operating system into performing malicious actions
- An adversary exploits a weakness resulting from using a hashing algorithm with weak collision resistance to generate a certificate signing request (CSR) that contains collision blocks in the ”to be signed” part
- An adversary can perform a DoS attack by making consecutive authorization requests
- A system user admin is not notified when there is an error logging the asset. Adversaries can use this by making the logging of the asset prone to errors by introducing a bug in the process. Now adversaries can compromise the system without it being logged
- An adversary compromises a system without knowing when he compromised it
- An adversary damages the device through physical access methods
- An adversary is able to intercept an unencrypted file in transit
- An adversary can access files through a device which was released from active service and which was not purged
- An adversary exploits a weakness in authentication to create an access token to associate a process/thread
- An adversary is able to hop between internal networks since they are not segregated
- Attackers make undetected and unaudited changes to system configurations
- An adversary could initiate a vast amount of sessions (DoS)
- An adversary can intercept network traffic
- An adversary may compromise the asset and edit the logfiles with non root privileges
- An adversary compromises a system without it being timely notified and detected by for example a IDS
- An adversary performs a DoS attack on a non-essential part of the asset which causes the core service to break

- Assets might not regain their state (user- and system-level information) due to storage failure
- An adversary can cause a power outage of a security system which will give an adversary to compromise the asset
- Attackers gain unauthorised access to data or services by accessing a client side secret
- An attacker obtains an authoritative or reputable signer's private signature key by theft
- Rogue assets might enter the network as a result of not being able to verify component identities
- Attackers gain unauthorized connection to the resources
- Attackers gain unauthorized access to the control of the environment
- Attackers try to take advantage of a wide attack surface
- Sensitive data is compromised through unauthorized access to data storage
- Exploitation of insufficient logging and monitoring
- Attackers perform unauthorized or gain network access
- The data is exposed through the transmission channel
- An adversary introduces malware onto the asset via a physical diagnostic or test interface
- An adversary can execute malicious code by compromising the host server, performing DNS spoofing, or modifying the code in transit.
- Unidentified software that executes arbitrary code
- An adversary modifies the device memory using a flash programmer
- An adversary can use error messages that contain too much information, such as stack traces, to discover vulnerabilities in the running service
- An adversary bypasses the secure-boot process and executes their own untrusted, malicious boot code
- An attacker examines a target system to find sensitive data that has been embedded within it
- Users lose trust in the system due to a perceived lack of security
- Attackers gain access to in-memory passwords/credentials

- Attackers manipulate session IDs and resource IDs to take advantage of the fact that some software accepts user input without verifying its authenticity
- Application contains security vulnerabilities not identified during the development process
- An adversary accesses the asset via an untrusted network
- Attackers gain unauthorised access to the application by the use of deprecated client-side technologies
- An adversary is able to tamper with software running on a system. On the asset no integrity check is in place
- When a weak encryption algorithm is used an adversary is capable of breaking the algorithm with standard available tools
- Assets might lose the capability to maintain essential functions when operating in a degraded mode as the result of a DoS event or resource exhaustion
- An attacker obtains an authoritative signer's private signature key by exploiting a cryptographic weaknesses
- Attackers perform an exhaustive (brute force) search on the key space to determine the key that decrypts the cipher text to obtain the plaintext
- Data leakage or disclosure to unauthorized parties
- Attackers who compromise the application or application server could directly access and modify the data store
- Unidentified backdoor plugged in the asset
- Attackers gain undetected access to the system by changing Virtual Machine configurations
- Unauthorized component access might occur as a result of not being able to change default access credentials (example: hardcoded in the software)
- An adversary may craft messages that appear to come from a different principle or use stolen / spoofed authentication credentials
- An adversary is able to delete audit records after a system breach which can make investigations very difficult
- Users' passwords are compromised if the storage medium is compromised

- An adversary can fuzz a sensor connected to a PLC which will cause the controller to generate wrong output values which can cause big system failures (DoS)
- Attackers attempt to exploit unpatched flaws to gain unauthorized access or knowledge of the system
- An adversary bypasses the secure-boot, executes untrusted or adversarial boot code of the device
- An attacker monitors information transmitted between logical or physical nodes of a network
- Sensitive data is compromised though attacks against SSL/TLS
- Attackers gain unauthorized access to data and/or systems through Injection attacks
- An adversary is able to exploit the update process of the device to escalate its privileges

Appendix E

Data flow description

See next page

ID	A02	A04	A05	A06	A07	A08
A01	**Architecture:** Peer-to-Peer **Data Flow:** C02 **Input/Output:** Job completeness status (in-progress, waiting), machine status (running, stopped), measures or tolerances values.	**Architecture:** Peer-to-Peer **Data Flow:** C06 **Input/Output:** PLC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status).	**Architecture:** Client-Server - A01 (client), A05 (server) **Data Flow:** C03, C05 **Task #1 Input:** Requesting the job parameters, sending Job ID number. **Task #1 Output:** job parameter necessary to produce. **Task #2 Input:** Sending the job complete status, may contain more information like end tolerances, count of jobs produced, etc. **Task #2 Output:** Message acknowledge	**Architecture:** Client-Server - A01 (server), A06 (client) **Data Flow:** C01 **Input:** Sensor data **Output:** Data to actuators		
A02			**Architecture:** Client-Server - A02 (client), A05 (server) **Data Flow:** C04 **Task #1 Input:** Requesting the job parameters, sending Job ID number **Task #1 Output:** job parameter necessary to produce **Task #2 Input:** Sending the job complete status, may contain more information like end tolerances, count of jobs produced, etc. **Task #2 Output:** Message acknowledge	**Architecture:** Client-Server - A01 (server), A06 (client) **Data Flow:** C01 **Input:** Sensor data **Output:** Data to actuators		**Architecture:** Client-Server - A02 (server), A08 (client) **Data Flow:** C09 **Input/Output:** IPC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status). And the eventual software update.
A03		**Architecture:** Peer-to-Peer **Data Flow:** C07 **Input/Output:** PLC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status).			**Architecture:** Peer-to-Peer **Data Flow:** C08 **Input/Output:** PLC access to perform configuration changes and get the machine operational data (error codes, counters, sub-component status).	

Figure E.1: Data Flow description full table

Appendix F

DREAD score definition

Damage Potential	
If the vulnerability is exploited, how much damage will be caused? (1-low , 3-bad)	
1	Individual system or sub component are compromised. Overall system availability or performance not affected.
2	Overall system performance affected but not the availability.
3	Both system performance and availability is affected.

Table F.1: DREAD - Damage Potential definition

Reproducibility	
How reliably can the vulnerability be exploited (attacker efficiency)? (1-very hard, 3-easy)	
1	Very hard or impossible, even for administrators. The vulnerability is unstable and statistically unlikely to be reliably exploited.
2	One or two steps required, tooling/scripting readily available.
3	Unauthenticated users can trivially and reliably exploit it.

Table F.2: DREAD - Reproducibility definition

Exploitability	
How difficult is the vulnerability to exploit (target simplicity)? (1-very hard, 3-easy)	
1	Even with direct knowledge of the vulnerability we do not see a viable path for exploitation.
2	Advanced techniques required, custom tooling. Only exploitable by authenticated users.
3	Exploit is available/understood, trivial usage.

Table F.3: DREAD - Exploitability definition

Affected Clients	
How many users will be affected? (1-limited, 3-very large)	
1	Individual user or system or sub component.
2	More than one system or sub component, but not all.
3	All or almost all systems are impacted.

Table F.4: DREAD - Affected Clients definition

Discoverability	
How easy is it to discover the threat, to learn of the vulnerability (information available about threat)? (1-hard, 3-very easy)	
1	Very hard to impossible to discover even given privilege access to running systems or system setup knowledge.
2	Can be figure it out by guessing or by monitoring network traffic (e.g. from a TA02 adversary).
3	Details of faults like this are already in the public domain or can be easily discovered.

Table F.5: DREAD - Discoverability definition

Appendix G

Threat DREAD assessment full list

ID	Damage	Reproducibility	Exploitability	Affected users	Discoverability	Risk Value
T01	3	3	2	3	3	48
T02	3	2	2	3	3	42
T03	2	3	3	2	3	36
T04	3	2	2	2	3	35
T05	2	3	2	2	3	32
T06	3	2	1	3	2	30
T07	3	2	2	2	2	30
T08	3	2	2	2	2	30
T09	2	2	2	2	3	28
T10	2	2	3	2	2	28
T11	3	2	2	2	1	25
T12	2	2	2	2	2	24
T13	2	2	2	2	2	24
T14	2	3	3	1	2	24
T15	2	2	2	2	2	24
T16	2	3	2	1	3	24
T17	2	3	2	1	3	24
T18	2	3	3	1	1	21
T19	2	2	3	1	2	21
T20	2	3	3	1	1	21
T21	2	1	2	2	2	20
T22	2	2	2	2	1	20
T23	2	2	2	1	2	18

ID	Damage	Reproducibility	Exploitability	Affected users	Discoverability	Risk Value
T24	1	2	2	2	2	18
T25	1	2	2	2	2	18
T26	2	2	1	1	3	18
T27	1	2	2	2	2	18
T28	1	2	2	2	2	18
T29	2	2	2	1	2	18
T30	1	3	2	1	3	16
T31	1	3	3	1	2	16
T32	1	1	2	3	1	16
T33	1	1	2	2	2	15
T34	2	1	1	1	3	15
T35	1	2	2	2	1	15
T36	2	2	1	1	2	15
T37	1	2	2	1	3	14
T38	1	2	2	1	3	14
T39	1	1	2	2	1	12
T40	1	1	1	1	3	10
T41	1	2	1	1	1	8
T42	1	1	1	1	1	6
T43	1	1	1	1	1	6

Table G.1: Threat DREAD classification

Appendix H

Attack tree

See next page

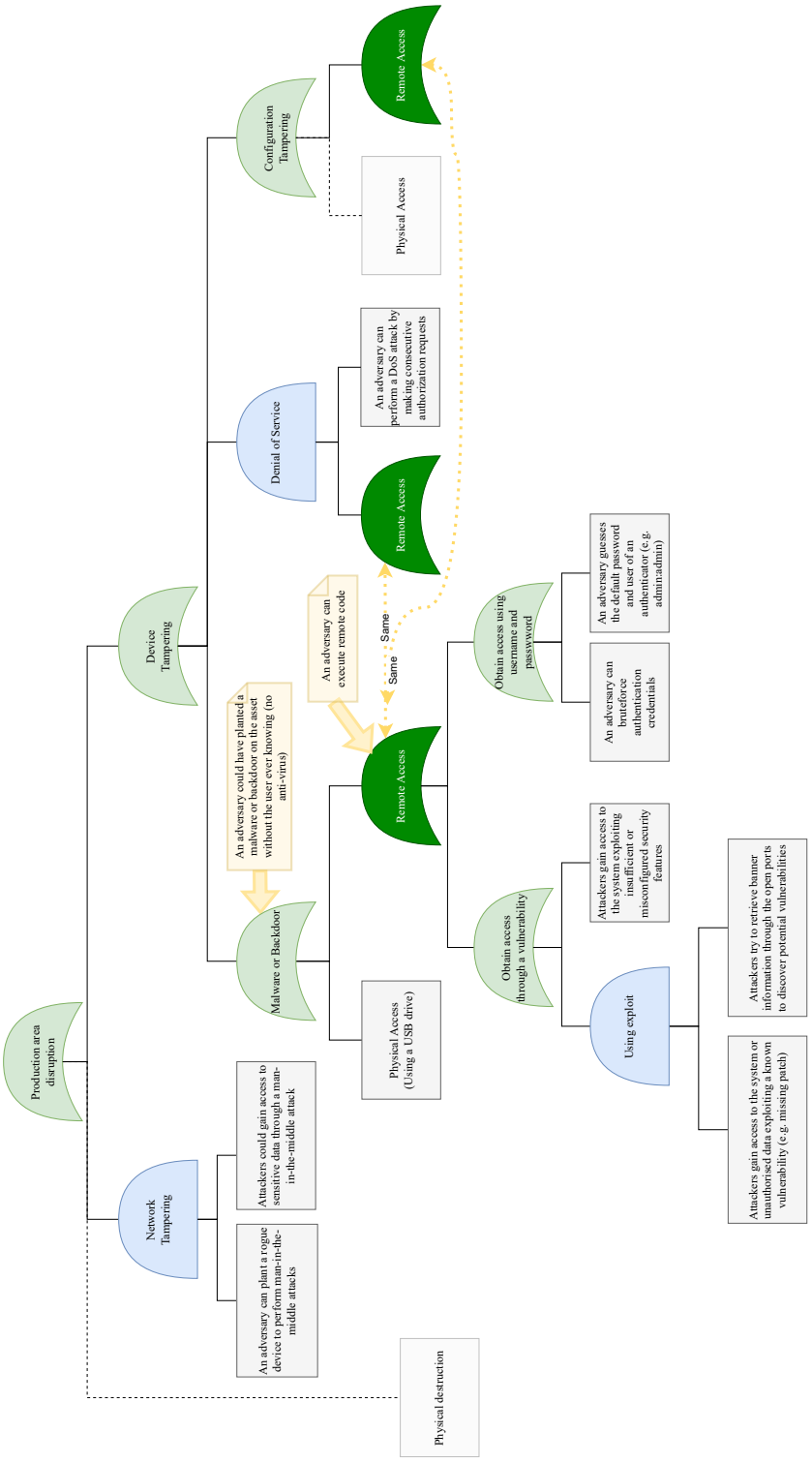


Figure H.1: Attack Tree

Appendix I

Commercial security monitoring tools

See next page

Figure I.1: Forrest vendors quadrant

Vendor	Product	Vendor statement	Asset discovery	Device vulnerability	Intrusion Detection
Microsoft	Defender for IoT	Continuous asset discovery, vulnerability management, and threat detection for your Internet of Things (IoT) and operational technology (OT) devices	•	•	•
Cisco	Cyber Vision	You cannot secure what you don't know. Cisco Cyber Vision gives you full visibility into your industrial control system (ICS), including dynamic asset inventory and real-time monitoring of process data.	•	•	•
Dragos	Dragos Platfrom	Most trusted industrial control systems (ICS) cybersecurity technology–providing comprehensive visibility of your ICS/OT assets and the threats you face, with best-practice guidance to respond before a significant compromise.	•	•	•
Claroty	Claroty Plataform	Our platform arms you with this knowledge by revealing and contextualizing 100% of your network's contents—including its invisible or poorly understood contents. The result is a centralized, easy-to-manage, and always up-to-date inventory of all OT, IoT, and IIoT assets, processes, and connectivity paths in your network, as well as definitive insight into what normal looks like.	•	•	•
Tenable	tenable.ot	It provides industrial and critical infrastructure operations with the visibility, security and control you need to ensure ongoing, safe facility operation while reducing overall risk.	•	•	•
Forescout	Forescout Plataform	Extends the industry leading device visibility, classification and profiling capabilities of the Forescout platform far deeper into OT and ICS environments. It enables the identification and effective remediation of a full range of both cyber and operational threats	•	•	•

Vendor	Product	Vendor statement	Asset discovery	Device vulnerability	Intrusion Detection
Nozomi	Guardian	See, secure and monitor all your ICS, OT, IoT, IT, edge and cloud assets with Guardian virtual and physical sensors.	•	•	•
Armis	Armis Plataform	Discover every asset in your environment with 100% visibility and rich context to track behavior, detect threats, and help you take action to protect your business.	•	•	•
Darktrace	Darktrace for OT	defends against known and unknown attacks at their earliest stages, providing unified protection across Operational Technology, IT, IIoT, and converged IT/OT ecosystems.	•		•
SCADAfence	SCADAfense Plataform	Continuous OT security network monitoring that provides visibility, risk management and threat detection.	•	•	•
Verve	Verve Plataform	OT/ICS cybersecurity platform that integrates a comprehensive suite of protection and defense to reduce cost and simplify operation.	•	•	Only Host IDS
PAS	PAS Cyber Integrity	PAS Cyber Integrity protects all control systems (Level 3 - Level 0) against cyber threats.	•	•	

Table I.1: Commercial tools overview

Appendix J

Hierarchical internetworking model

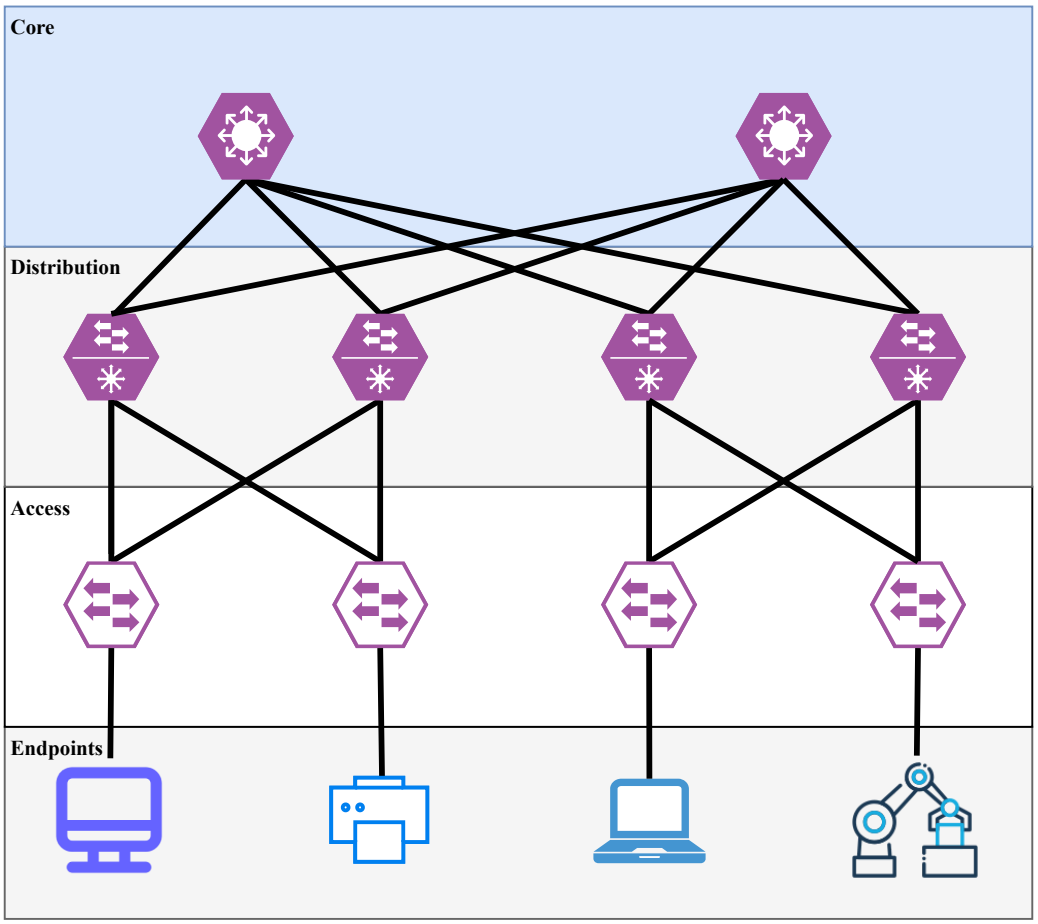


Figure J.1: Hierarchical internetworking model

Appendix K

Asset inventory

See next page

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.10	Y100AF36	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3030136v006
10.15.A.101	Y100AG11	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v004
10.15.A.102	Y100PNCM122	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193518
10.15.A.103	Y100PNCM123	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193558
10.15.A.104	Y100PNCM124	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193545
10.15.A.105	Y100PNCM125	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194041
10.15.A.106	Y100PNCM126	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.107	Y100PNCM127	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194045
10.15.A.108	Y100PNCM128	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194045
10.15.A.11	ZONCM060	Windows XP	Ridder		Masch. Nr.: E-10632
10.15.A.110	Y100AG51	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v002
10.15.A.111	Y100PNCM131	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.112	Y100PNCM132	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.113	Y100PNCM133	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.114	Y100PNCM134	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.115	ZONCM086	Windows XP	Siemens		
10.15.A.117	Y100AG52	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	M110 i5-7500 AIO 23,6”	3008837S003
10.15.A.118	Y100AD88	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v002
10.15.A.119	Y100PSROED01	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.12	Y100AI69	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v009
10.15.A.120	Y100PSROED04	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.121	Y100AD61	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v003
10.15.A.122	Y100PSROED06	Windows 10 LTSB 2016 - 64-Bit	Röders		
10.15.A.123	Y100JC0009	Windows 10 LTSC 2019 - 64-Bit		Virtueller Server	
10.15.A.124	Y100AG65	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v008
10.15.A.125	Y100PNCM135	Linux			
10.15.A.127	Y100PSROED07	Windows 10 LTSC 2019 - 64-Bit	Röders		

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.128	Y100AG42	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v001
10.15.A.13	Y100PVTU03	Microsoft Windows CE .NET 4.0			
10.15.A.130	Y100AG63	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3076011v008
10.15.A.132	Y100AG64	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3076011v010
10.15.A.139	Y100AG44	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v020
10.15.A.14	Y100AH69	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v005
10.15.A.140	Y100AC15	Windows 7 - 64-Bit	EXTRA Computer GmbH	3236-S i7-4790 4HE	2939749V002
10.15.A.141	Y100POR041	Microsoft Windows CE 5.0			
10.15.A.142	Y100POR042	Microsoft Windows CE 5.0			
10.15.A.143	Y100POR043	Microsoft Windows CE 5.0			
10.15.A.144	Y100POR044	Microsoft Windows CE 5.0			
10.15.A.145	Y100POR045	Microsoft Windows CE 5.0			
10.15.A.146	Y100POR046	Microsoft Windows CE 5.0			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.147	Y100AE04	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v008
10.15.A.148	Y100AE05	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v007
10.15.A.15	Y100AJ57	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3633-S i3-9100 Uni4	3149498s001
10.15.A.151	Y1NCM006	OEM embedded	NUM Telemecanique		
10.15.A.152	Y1NCM040	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.155	Y1NCM038	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.156	Y1NCM017	OEM embedded	Grundig		
10.15.A.157	Y1NCM032	OEM embedded			
10.15.A.158	Y1NCM033	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11160000343
10.15.A.159	Y1NCM030	Microsoft Windows 95/98/ME	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 684733

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.16	Y100PNCM16	Microsoft Windows CE .NET 4.0	MAKINO		
10.15.A.160	Y1NCM014	Microsoft Windows 95/98/ME	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 684743
10.15.A.161	Y1NCM019	OEM embedded	Heidenhain		Masch. Nr: 00/27208-1001
10.15.A.162	Y1NCM020	OEM embedded	Heidenhain		Masch. Nr: 01/28027-1004
10.15.A.163	Y1NCM031	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr. 6627663
10.15.A.164	Y1NCM049	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001523
10.15.A.165	Y1NCM023	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200000533
10.15.A.166	ZONCM090	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001243

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.167	Y1NCM025	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200000773
10.15.A.168	Y1NCM027	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200001253
10.15.A.169	Y1NCM029	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11300000063
10.15.A.17	Y100AC70	Windows 7 - 64-Bit	EXTRA Computer GmbH	3236-S i7-4790 4HE	2948275V007
10.15.A.170	Y1NCM041	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 1131000043
10.15.A.171	Y1NCM022	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001503
10.15.A.172	Y1NCM045	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11610000053

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.173	Y1NCM048	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Interne Nr.: 1109
10.15.A.174	Y1NCM021	OEM embedded	Siemens		
10.15.A.175	Y1NCM001	OEM embedded	Siemens		
10.15.A.176	Y1NCM009	Microsoft Windows NT Workstation 4.0	Siemens		
10.15.A.177	Y1NCM003	OEM embedded	Siemens		
10.15.A.178	Y1NCM010	OEM embedded	Siemens		
10.15.A.179	Y1NCM011	OEM embedded	Siemens		
10.15.A.180	ZONCM012	DOS	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.181	Y100POR057	Microsoft Windows CE 5.0			
10.15.A.182	Y100POR058	Microsoft Windows CE 5.0			
10.15.A.183	Y100POR059	Microsoft Windows CE 5.0			
10.15.A.184	Y100POR060	Microsoft Windows CE 5.0			
10.15.A.185	Y100POR061	Microsoft Windows CE 5.0			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.186	Y100AE46	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	3021410v006
10.15.A.187	Y100AJ29	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3117938V012
10.15.A.188	Y100AJ30	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3117938V017
10.15.A.189	Y100AJ31	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3117938v016
10.15.A.19	Y100PHT02	Windows 10 LTSB 2016 - 64-Bit			
10.15.A.190	Y100PROEDZ1	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.191	Y100PROEDZ2	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.192	Y100PROEDZ3	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.2	Y100AD05	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2956220V007
10.15.A.20	Y100AF83	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v008
10.15.A.21	Y100NCM20	Linux	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.29	Y19NCM32	Microsoft Windows 2000 Professional Edition	MAKINO		
10.15.A.30	Y19NCM35	Microsoft Windows CE 5.0			
10.15.A.31	Y100PCPT01	Microsoft Windows CE 5.0			
10.15.A.32	Y100PCPT02	Microsoft Windows CE 5.0			
10.15.A.33	Y100PCPT03	Microsoft Windows CE 5.0			
10.15.A.35	Y100PCPT05	Microsoft Windows CE 5.0			
10.15.A.36	Y100PCPT06	Microsoft Windows CE 5.0			
10.15.A.37	Y100PCPT07	Microsoft Windows CE 5.0			
10.15.A.39	Y19NCM33	Microsoft Windows CE 5.0			
10.15.A.4	Y100AD22	Windows 7 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2956220V013
10.15.A.40	Y100POR020	Microsoft Windows CE 5.0			
10.15.A.43	Y100POR013	Microsoft Windows CE 6.0			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.44	Y100POR014	Microsoft Windows CE .NET 4.0			
10.15.A.45	Y100POR015	Microsoft Windows CE .NET 4.0			
10.15.A.46	Y100POR016	Microsoft Windows CE .NET 4.0			
10.15.A.47	Y100POR017	Microsoft Windows CE .NET 4.0			
10.15.A.48	Y100POR018	Microsoft Windows CE .NET 4.0			
10.15.A.49	Y100POR019	Microsoft Windows CE .NET 4.0			
10.15.A.5	Y100AD40	OEM embedded			0000051508300541
10.15.A.51	Y19NCM36	Microsoft Windows 2000 Professional Edition	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.52	Y100PSROED02	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.53	Y100PS11	OEM embedded			hr50680409b2005gf
10.15.A.54	Y100PS12	OEM embedded			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.55	Y100PSROED03	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.56	ZONCM092	Windows XP	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11700001023
10.15.A.57	Y100PDMU4	Microsoft Windows 2000 Professional Edition	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.59	Y100AD48	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2983458v003
10.15.A.61	Y19NCM25	Microsoft Windows 2000 Professional Edition	MAKINO		
10.15.A.62	Y100POR050	Microsoft Windows CE 5.0			
10.15.A.63	Y100POR021	Microsoft Windows CE 6.0			
10.15.A.64	Y100POR022	Microsoft Windows CE 6.0			
10.15.A.65	Y100POR023	Microsoft Windows CE 6.0			
10.15.A.66	Y100POR024	Microsoft Windows CE 6.0			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.67	Y100POR025	Microsoft Windows CE 6.0			
10.15.A.68	Y100POR026	Microsoft Windows CE 6.0			
10.15.A.69	Y100POR027	Microsoft Windows CE 6.0			
10.15.A.7	Y100AC06	Windows 7 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2915440s002
10.15.A.70	Y100POR028	Microsoft Windows CE 6.0			
10.15.A.73	Y100POR040	Microsoft Windows CE 6.0			
10.15.A.75	FRT02	Microsoft Windows 95/98/ME	FRT		
10.15.A.76	Y100POR029	Windows 7 - 64-Bit	Shuttle		X50V2P1601C01F00509
10.15.A.79	Y100AH57	Windows 7 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v003
10.15.A.8	Y100AD18	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S1 i5-6500 4HE	2972034v001
10.15.A.81	Y100AC31	Windows 7 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2936501S002
10.15.A.82	Y100AC84	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2951299V001
10.15.A.83	Y100AH73	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v004

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.84	Y100POR030	Windows 7 - 32-Bit	Shuttle		
10.15.A.86	NONAMEBDE	Microsoft Windows CE 5.0			
10.15.A.87	Y100POR051	Microsoft Windows CE 5.0			
10.15.A.88	Y100AD71	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v001
10.15.A.89	Y100AD64	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2983458v001
10.15.A.9	Y100AG09	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v010
10.15.A.90	Y100PSROED05	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.91	Y100POR052	Microsoft Windows CE 5.0			
10.15.A.92	Y100POR007	Windows 7 - 32-Bit			
10.15.A.93	Y100POR054	Microsoft Windows CE 5.0			
10.15.A.94	Y100POR053	Microsoft Windows CE 5.0			
10.15.A.96	Y100POR012	Microsoft Windows CE .NET 4.0			
10.15.A.97	Y100POR055	Microsoft Windows CE 5.0			

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.A.98	Y100POR056	Microsoft Windows CE 5.0			
10.15.A.99	Y100AB98	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2915440s009
10.15.C.1	Y100PIBF13	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPMO955810
10.15.C.10	Y100PIBF33	Windows XP	Siemens		
10.15.C.12	Y100PACE01	Windows XP	Alzmetall		
10.15.C.13	Y100AC23	Windows 7 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2915440s007
10.15.C.15	Y100AE57	Windows 10 LTSB 2016 - 64-Bit	Beckhoff Automation GmbH & Co. KG		3378991-001
10.15.C.16	Y100PIBF12	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPLO955925
10.15.C.17	Y100PIBF08	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPM6957257
10.15.C.18	Y100PIBF10	Windows XP	Siemens		
10.15.C.19	Y100PIBFE2	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPM8954341
10.15.C.2	Y100PIBF09	Windows XP	Siemens		
10.15.C.20	Y100AJ14	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3633-S i3-9100 Uni4	3136392v006
10.15.C.3	Y100PIBF03	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPL1954959
10.15.C.35	Y100PIBF35	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPL3958959
10.15.C.4	Y100PIBF31	Windows XP	Siemens		

IP	Name	Operating system	Vendor	Standard Hardware	Serial number
10.15.C.5	Y100PIBF21	Windows XP	Siemens		
10.15.C.6	Y100PIBFE3	Windows XP	Siemens		
10.15.C.7	Y100PIBF07	Windows XP	Siemens		
10.15.C.9	Y100PIBF11	Windows 10 LTSB 2016 - 64-Bit	Siemens		VPM2955633

Table K.1: VLAN 988 and 967 asset inventory - Initial

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.10	Y100AF36	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3030136v006
10.15.A.101	Y100AG11	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v004
10.15.A.102	Y100PNCM122	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193518
10.15.A.103	Y100PNCM123	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193558
10.15.A.104	Y100PNCM124	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 193545
10.15.A.105	Y100PNCM125	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194041
10.15.A.106	Y100PNCM126	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.107	Y100PNCM127	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194045
10.15.A.108	Y100PNCM128	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		Masch. Nr.: 194045
10.15.A.11	ZONCM060	Windows XP	Ridder		Masch. Nr.: E-10632
10.15.A.110	Y100AG51	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v002
10.15.A.111	Y100PNCM131	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.112	Y100PNCM132	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.113	Y100PNCM133	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.114	Y100PNCM134	OEM embedded	GF Machining Solutions GmbH (ehemals ”AgieCharmilles”)		
10.15.A.115	ZONCM086	Windows XP			
10.15.A.117	Y100AG52	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	M110 i5-7500 AIO 23,6”	3008837S003
10.15.A.118	Y100AD88	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v002
10.15.A.119	Y100PSROED01	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.12	Y100AI69	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v009
10.15.A.120	Y100PSROED04	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.121	Y100AD61	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v003

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.122	Y100PSROED06	Windows 10 LTSB 2016 - 64-Bit	Röders		
10.15.A.124	Y100AG65	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v008
10.15.A.125	Y100PNCM135	Linux			
10.15.A.127	Y100PSROED07	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.128	Y100AG42	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v001
10.15.A.13	Y100PVTU03	Microsoft Windows CE .NET 4.0			
10.15.A.130	Y100AG63	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3076011v008
10.15.A.132	Y100AG64	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S2 i5-6500 4HE	3076011v010
10.15.A.139	Y100AG44	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3073529v020

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.14	Y100AH69	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v005
10.15.A.140	Y100AC15	Windows 7 - 64-Bit	EXTRA Computer GmbH	3236-S i7-4790 4HE	2939749V002
10.15.A.146	Y100POR046	Microsoft Windows CE 5.0			
10.15.A.147	Y100AE04	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v008
10.15.A.148	Y100AE05	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v007
10.15.A.15	Y100AJ57	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3633-S i3-9100 Uni4	3149498s001
10.15.A.151	Y1NCM006	OEM embedded	NUM Telemecanique		
10.15.A.152	Y1NCM040	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.155	Y1NCM038	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.156	Y1NCM017	OEM embedded	Grundig		

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.158	Y1NCM033	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11160000343
10.15.A.159	Y1NCM030	Microsoft Windows 95/98/ME	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 684733
10.15.A.16	Y100PNCM16	Microsoft Windows CE .NET 4.0	MAKINO		
10.15.A.160	Y1NCM014	Microsoft Windows 95/98/ME	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 684743
10.15.A.161	Y1NCM019	OEM embedded	Heidenhain		Masch. Nr: 00/27208-1001
10.15.A.162	Y1NCM020	OEM embedded	Heidenhain		Masch. Nr: 01/28027-1004
10.15.A.163	Y1NCM031	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr. 6627663
10.15.A.164	Y1NCM049	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001523

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.165	Y1NCM023	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200000533
10.15.A.166	ZONCM090	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001243
10.15.A.167	Y1NCM025	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200000773
10.15.A.168	Y1NCM027	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11200001253
10.15.A.169	Y1NCM029	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11300000063
10.15.A.17	Y100AC70	Windows 7 - 64-Bit	EXTRA Computer GmbH	3236-S i7-4790 4HE	2948275V007
10.15.A.170	Y1NCM041	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 1131000043

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.171	Y1NCM022	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11640001503
10.15.A.172	Y1NCM045	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11610000053
10.15.A.173	Y1NCM048	OEM embedded	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Interne Nr.: 1109
10.15.A.174	Y1NCM021	OEM embedded			
10.15.A.175	Y1NCM001	OEM embedded			
10.15.A.176	Y1NCM009	Microsoft Windows NT Workstation 4.0			
10.15.A.177	Y1NCM003	OEM embedded			
10.15.A.178	Y1NCM010	OEM embedded			
10.15.A.179	Y1NCM011	OEM embedded			
10.15.A.180	ZONCM012	DOS	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.181	Y100POR057	Microsoft Windows CE 5.0			

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.182	Y100POR058	Microsoft Windows CE 5.0			
10.15.A.183	Y100POR059	Microsoft Windows CE 5.0			
10.15.A.184	Y100POR060	Microsoft Windows CE 5.0			
10.15.A.185	Y100POR061	Microsoft Windows CE 5.0			
10.15.A.186	Y100AE46	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	3021410v006
10.15.A.19	Y100PHT02	Windows 10 LTSB 2016 - 64-Bit			
10.15.A.190	Y100PROEDZ1	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.191	Y100PROEDZ2	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.192	Y100PROEDZ3	Windows 10 LTSC 2019 - 64-Bit			
10.15.A.2	Y100AD05	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2956220V007

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.20	Y100AF83	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v008
10.15.A.21	Y100NCM20	Linux	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.24	Y100AE42		EXTRA Computer GmbH		
10.15.A.25	Y100AK17		EXTRA Computer GmbH		
10.15.A.26	Y100AK13		EXTRA Computer GmbH		
10.15.A.29	Y19NCM32	Microsoft Windows 2000 Professional Edition	MAKINO		
10.15.A.30	Y19NCM35	Microsoft Windows CE 5.0			
10.15.A.33	Y100PCPT03	Microsoft Windows CE 5.0			
10.15.A.4	Y100NCM21	Windows 7 - 64-Bit	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)	3433-S i3-6100T Uni4	2956220V013
10.15.A.43	Y100POR013	Microsoft Windows CE 6.0			
10.15.A.44	Y100POR014	Microsoft Windows CE .NET 4.0			

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.45	Y100POR015	Microsoft Windows CE .NET 4.0			
10.15.A.46	Y100POR016	Microsoft Windows CE .NET 4.0			
10.15.A.48	Y100POR018	Microsoft Windows CE .NET 4.0			
10.15.A.49	Y100POR019	Microsoft Windows CE .NET 4.0			
10.15.A.5	Y100AD40	OEM embedded			0000051508300541
10.15.A.50	Y100AK34		EXTRA Computer GmbH		
10.15.A.51	Y19NCM36	Microsoft Windows 2000 Professional Edition	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.52	Y100PSROED02	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.53	Y100PS11	OEM embedded			hr50680409b2005gf
10.15.A.54	Y100PS12	OEM embedded			
10.15.A.55	Y100PSROED03	Windows 10 LTSC 2019 - 64-Bit	Röders		

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.56	ZONCM092	Windows XP	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		Masch. Nr: 11700001023
10.15.A.57	Y100PDMU4	Microsoft Windows 2000 Professional Edition	DMG MORI AKTIENGESELLSCHAFT (ehemals ”Deckel”, ”MAHO” und ”Gildemeiser”)		
10.15.A.59	Y100AD48	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2983458v003
10.15.A.61	Y19NCM25	Microsoft Windows 2000 Professional Edition	MAKINO		
10.15.A.62	Y100POR050	Microsoft Windows CE 5.0			
10.15.A.63	Y100POR021	Microsoft Windows CE 6.0			
10.15.A.64	Y100POR022	Microsoft Windows CE 6.0			
10.15.A.65	Y100POR023	Microsoft Windows CE 6.0			
10.15.A.66	Y100POR024	Microsoft Windows CE 6.0			
10.15.A.67	Y100POR025	Microsoft Windows CE 6.0			

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.68	Y100POR026	Microsoft Windows CE 6.0			
10.15.A.70	Y100POR028	Microsoft Windows CE 6.0			
10.15.A.73	Y100POR040	Microsoft Windows CE 6.0			
10.15.A.75	FRT02	Microsoft Windows 95/98/ME	FRT		
10.15.A.76	Y100POR029	Windows 7 - 64-Bit	Shuttle		X50V2P1601C01F00509
10.15.A.8	Y100AD18	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3446-S1 i5-6500 4HE	2972034v001
10.15.A.82	Y100AC84	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2951299V001
10.15.A.83	Y100AH73	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3102357v004
10.15.A.87	Y100POR051	Microsoft Windows CE 5.0			
10.15.A.88	Y100AD71	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2989272v001

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.A.89	Y100AD64	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S i3-6100T Uni4	2983458v001
10.15.A.9	Y100AG09	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3433-S2 i3-6100 Uni4	3067530v010
10.15.A.90	Y100PSROED05	Windows 10 LTSC 2019 - 64-Bit	Röders		
10.15.A.91	Y100POR052	Microsoft Windows CE 5.0			
10.15.A.93	Y100POR054	Microsoft Windows CE 5.0			
10.15.A.94	Y100POR053	Microsoft Windows CE 5.0			
10.15.A.96	Y100POR012	Microsoft Windows CE .NET 4.0			
10.15.A.97	Y100POR055	Microsoft Windows CE 5.0			
10.15.A.98	Y100POR056	Microsoft Windows CE 5.0			
10.15.A.99	Y100AB98	Windows 10 LTSB 2016 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2915440s009

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.C.1	Y100PIBF13	Windows 10 LTSB 2016 - 64-Bit			VPMO955810
10.15.C.10	Y100PIBF33	Windows XP			
10.15.C.11	Y100PIBFE1				
10.15.C.12	Y100PACE01	Windows XP	Alzmetall		
10.15.C.13	Y100AC23	Windows 7 - 64-Bit	EXTRA Computer GmbH	3313-S3 QuadCore Uni3	2915440s007
10.15.C.15	Y100AE57	Windows 10 LTSB 2016 - 64-Bit	Beckhoff Automation GmbH & Co. KG		3378991-001
10.15.C.16	Y100PIBF12	Windows 10 LTSB 2016 - 64-Bit			VPLO955925
10.15.C.17	Y100PIBF08	Windows 10 LTSB 2016 - 64-Bit			VPM6957257
10.15.C.18	Y100PIBF10	Windows XP			
10.15.C.19	Y100PIBFE2	Windows 10 LTSB 2016 - 64-Bit			VPM8954341
10.15.C.2	Y100PIBF09	Windows XP			
10.15.C.20	Y100AJ14	Windows 10 LTSC 2019 - 64-Bit	EXTRA Computer GmbH	3633-S i3-9100 Uni4	3136392v006
10.15.C.22	Y100PIBFE3				

IP	Name	Operating System	Vendor	Standard Hardware	Serial Number
10.15.C.3	Y100PIBF03	Windows 10 LTSB 2016 - 64-Bit			VPL1954959
10.15.C.35	Y100PIBF35	Windows 10 LTSB 2016 - 64-Bit			VPL3958959
10.15.C.4	Y100PIBF31	Windows XP			
10.15.C.5	Y100PIBF21	Windows XP			
10.15.C.7	Y100PIBF07	Windows XP			
10.15.C.9	Y100PIBF11	Windows 10 LTSB 2016 - 64-Bit			VPM2955633

Table K.2: VLAN 988 and 967 asset inventory - at the End

IP	Name	Vendor	Firmware version	Tags
10.15.A.100	Wibrain 10.15.A.100	WIBRAIN		
10.15.A.101	Y100AG11	Fujitsu Technology Solutions GmbH	10.0.14393	File Transfer Server;HTTP Client;Remote Admin Server;Microsoft Exchange;Windows;Email Server;
10.15.A.110	Y100AG51	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.114	Wiesemann 10.15.A.114	WIESEMANN & THEIS GMBH		
10.15.A.115	PO809	Siemens AG		Windows;
10.15.A.119	Y100PSROED01	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.12	Y100AI69	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.120	Y100PSROED04	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.122	Y100PSROED06	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.123	Y100JC0009	VMware, Inc.		Windows;
10.15.A.124	Y100AG65	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.125	Siemens 10.15.A.125	Siemens AG, Sector Industry, Drive Technologies, Motion Control Systems		Remote Admin Server;
10.15.A.126	Wiesemann 10.15.A.126	WIESEMANN & THEIS GMBH		File Transfer Server;

IP	Name	Vendor	Firmware version	Tags
10.15.A.127	Y100PSROED07	Fujitsu Technology Solutions GmbH	10.0.17763	Windows;Email Server;File Transfer Server;HTTP Client;Remote Admin Server;Microsoft Exchange;
10.15.A.128	Y100AG42	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.129	Wiesemann 10.15.A.129	WIESEMANN & THEIS GMBH		
10.15.A.130	Y100AG63	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.132	Y100AG64	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.133	Wiesemann 10.15.A.133	WIESEMANN & THEIS GMBH		
10.15.A.14	Y100AH69	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.140	Y100AC15	Fujitsu Technology Solutions GmbH	6.1.7601	Windows;Email Server;File Transfer Server;HTTP Client;HTTPS Client;Remote Admin Server;Web Server;Microsoft Exchange;
10.15.A.141	Advantech 10.15.A.141	ADVANTECH CO., LTD.		
10.15.A.142	Advantech 10.15.A.142	ADVANTECH CO., LTD.		
10.15.A.143	Advantech 10.15.A.143	ADVANTECH CO., LTD.		
10.15.A.144	Advantech 10.15.A.144	ADVANTECH CO., LTD.		
10.15.A.145	Advantech 10.15.A.145	ADVANTECH CO., LTD.		Admin Server;File Transfer Server;Web Server;
10.15.A.146	Advantech 10.15.A.146	ADVANTECH CO., LTD.		

IP	Name	Vendor	Firmware version	Tags
10.15.A.147	Y100AE04	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.148	Y100AE05	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.152	Dr 10.15.A.152	DR. JOHANNES HEIDENHAIN GmbH		File Transfer Server;
10.15.A.156	Wiesemann 10.15.A.156	WIESEMANN & THEIS GMBH		
10.15.A.164	Dr 10.15.A.164	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.165	Dr 10.15.A.165	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.166	Dr 10.15.A.166	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.167	Dr 10.15.A.167	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.168	Dr 10.15.A.168	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.169	Dr 10.15.A.169	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.17	Y100AC70	Fujitsu Technology Solutions GmbH	6.1.7601	Windows;
10.15.A.170	Dr 10.15.A.170	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.171	Dr 10.15.A.171	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.172	Dr 10.15.A.172	DR. JOHANNES HEIDENHAIN GmbH		

IP	Name	Vendor	Firmware version	Tags
10.15.A.173	Dr 10.15.A.173	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.174	Wiesemann 10.15.A.174	WIESEMANN & THEIS GMBH		
10.15.A.176	Wiesemann 10.15.A.176	WIESEMANN & THEIS GMBH		
10.15.A.177	Wiesemann 10.15.A.177	WIESEMANN & THEIS GMBH		File Transfer Server;
10.15.A.178	Wiesemann 10.15.A.178	WIESEMANN & THEIS GMBH		File Transfer Server;
10.15.A.179	Wiesemann 10.15.A.179	WIESEMANN & THEIS GMBH		
10.15.A.181	Advantech 10.15.A.181	ADVANTECH CO., LTD.		
10.15.A.182	Advantech 10.15.A.182	ADVANTECH CO., LTD.		
10.15.A.183	Advantech 10.15.A.183	ADVANTECH CO., LTD.		
10.15.A.184	Advantech 10.15.A.184	ADVANTECH CO., LTD.		
10.15.A.185	Advantech 10.15.A.185	ADVANTECH CO., LTD.		
10.15.A.186	Y100AE46	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.19	Y100PHT02	ads-tec GmbH		Windows;
10.15.A.190	Y100PROEDZ1	Fujitsu Technology Solutions GmbH	10.0.17763	Windows;
10.15.A.191	Y100PROEDZ2	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.192	Y100PROEDZ3	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.2	Y100AD05	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.20	Y100AF83	Fujitsu Technology Solutions GmbH	10.0.14393	Microsoft Exchange;Windows;Email Server;File Transfer Server;HTTP Client;Host Config Client;

IP	Name	Vendor	Firmware version	Tags
10.15.A.21	Dr 10.15.A.21	DR. JOHANNES HEIDENHAIN GmbH		
10.15.A.23	SCCMTESTMASD	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.29	Makino 10.15.A.29	Makino Milling Machine Co., Ltd.		
10.15.A.30	Advantech 10.15.A.30	ADVANTECH CO., LTD.		
10.15.A.31	Advantech 10.15.A.31	ADVANTECH CO., LTD.		
10.15.A.32	Advantech 10.15.A.32	ADVANTECH CO., LTD.		
10.15.A.33	Advantech 10.15.A.33	ADVANTECH CO., LTD.		
10.15.A.34	Advantech 10.15.A.34	ADVANTECH CO., LTD.		
10.15.A.35	Advantech 10.15.A.35	ADVANTECH CO., LTD.		
10.15.A.36	Advantech 10.15.A.36	ADVANTECH CO., LTD.		
10.15.A.37	Advantech 10.15.A.37	ADVANTECH CO., LTD.		
10.15.A.39	Advantech 10.15.A.39	ADVANTECH CO., LTD.		
10.15.A.4	Y100AD22	Fujitsu Technology Solutions GmbH	6.1.7601	Windows;
10.15.A.40	Advantech 10.15.A.40	ADVANTECH CO., LTD.		
10.15.A.44	Advantech 10.15.A.44	ADVANTECH CO., LTD.		
10.15.A.45	Advantech 10.15.A.45	ADVANTECH CO., LTD.		
10.15.A.47	Advantech 10.15.A.47	ADVANTECH CO., LTD.		
10.15.A.52	Y100PSROED02	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.53	Hurco 10.15.A.53	Hurco Automation Ltd.		
10.15.A.54	Hurco 10.15.A.54	Hurco Automation Ltd.		
10.15.A.55	Y100PSROED03	Fujitsu Technology Solutions GmbH		Windows;

IP	Name	Vendor	Firmware version	Tags
10.15.A.56	ZONCM092	DR. JOHANNES HEIDENHAIN GmbH	5.1.2600	Windows;
10.15.A.57	Y100PDMU4	DR. JOHANNES HEIDENHAIN GmbH	5.0.2195	File Transfer Server;Windows;
10.15.A.61	Makino 10.15.A.61	Makino Milling Machine Co., Ltd.		
10.15.A.62	Advantech 10.15.A.62	ADVANTECH CO., LTD.		
10.15.A.66	Advantech 10.15.A.66	ADVANTECH CO., LTD.		
10.15.A.69	Advantech 10.15.A.69	ADVANTECH CO., LTD.		Admin Server;File Transfer Server;Web Server;
10.15.A.70	Advantech 10.15.A.70	ADVANTECH CO., LTD.		
10.15.A.73	Advantech 10.15.A.73	ADVANTECH CO., LTD.		
10.15.A.76	Y100POR029			Windows;
10.15.A.77	Wiesemann 10.15.A.77	WIESEMANN & THEIS GMBH		
10.15.A.79	Y100AH57	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.8	Y100AD18	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.81	Y100AC31		6.1.7601	Windows;
10.15.A.82	Y100AC84	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.A.83	Y100AH73	Fujitsu Technology Solutions GmbH		Windows;
10.15.A.86	Advantech 10.15.A.86	ADVANTECH CO., LTD.		
10.15.A.87	Advantech 10.15.A.87	ADVANTECH CO., LTD.		
10.15.A.9	Y100AG09	Fujitsu Technology Solutions GmbH		Windows;

IP	Name	Vendor	Firmware version	Tags
10.15.A.90	Y100PSROED05	Fujitsu Technology Solutions GmbH		Windows;Remote Admin Server;
10.15.A.91	Advantech 10.15.A.91	ADVANTECH CO., LTD.		
10.15.A.92	Y100POR007	BIOSTAR Microtech Int'l Corp.		Windows;
10.15.A.99	Y100AB98	Fujitsu Technology Solutions GmbH	10.0.14393	Windows;
10.15.C.1	Y100PIBF13	SIEMENS AG		HTTP Client;HTTPS Client;Host Config Client;Web Server;Windows;File Transfer Server;
10.15.C.10	Y100PIBF33	Siemens AG,		Windows;
10.15.C.11	Y100PIBFE1	PORTWELL, INC.		Windows;
10.15.C.12	Y100PACE01	SIEMENS AG		Web Server;Windows;File Transfer Server;Host Config Client;
10.15.C.13	Y100AC23	Fujitsu Technology Solutions GmbH	6.1.7601	HTTP Client;Windows;
10.15.C.15	Y100AE57	Intel Corporate	10.0.14393	Windows;
10.15.C.16	Y100PIBF12	SIEMENS AG	10.0.14393	Windows;File Transfer Server;HTTP Client;HTTPS Client;Host Config Client;Remote Admin Server;Web Server;
10.15.C.17	Y100PIBF08	SIEMENS AG		Windows;File Transfer Server;HTTP Client;HTTPS Client;Host Config Client;Web Server;
10.15.C.18	Y100PIBF10	Siemens AG	5.1.2600	Windows;
10.15.C.19	Y100PIBFE2	SIEMENS AG		Windows;

IP	Name	Vendor	Firmware version	Tags
10.15.C.2	Y100PIBF09	Siemens AG	5.1.2600	Windows;
10.15.C.20	Y100AJ14	Fujitsu Technology Solutions GmbH		Windows;
10.15.C.21	Y100AD07	Siemens AG,	6.1.7601	Windows;
10.15.C.3	Y100PIBF03	SIEMENS AG		Web Server;Windows;File Transfer Server;HTTP Client;HTTPS Client;Host Config Client;Remote Admin Server;
10.15.C.35	Y100PIBF35	SIEMENS AG		HTTP Client;HTTPS Client;Host Config Client;Remote Admin Server;Web Server;Windows;File Transfer Server;
10.15.C.4	Y100PIBF31	SIEMENS AG	5.1.2600	File Transfer Server;Router;Windows;Controller;Host Config Client;Web Server;Citect Alarm Server;Citect Report Server;Citect Trend Server;
10.15.C.5	Y100PIBF21	Siemens AG	5.1.2600	Windows;Database Server;File Transfer Server;Host Config Client;Web Server;
10.15.C.6	Y100PIBFE3	Siemens AG	5.1.2600	Windows;
10.15.C.7	Y100PIBF07	Siemens AG		Windows;
10.15.C.9	Y100PIBF11	SIEMENS AG		File Transfer Server;HTTP Client;HTTPS Client;Host Config Client;Web Server;Windows;

Table K.3: VLAN 988 and 967 Cisco Cyber Vision asset identified

IP	Name	Operating System	Vendor
10.15.A.101	Y100AG11	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.110	Y100AG51	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.115	PO809	Windows XP	Siemens AG A&D ET
10.15.A.117	Y100AG52	Windows 10 / Server 2016 / Server 2019 / Server 2016	MITAC INTERNATIONAL CORP.
10.15.A.119	Y100PSROED01	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.12	Y100AI69	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.120	Y100PSROED04	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.122	Y100PSROED06	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.123	Y100JC0009	Windows 10 / Server 2016 / Server 2019	VMware, Inc.
10.15.A.124	Y100AG65	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.125	10.15.A.125		Siemens AG, Sector Industry, Drive Technologies, Motion Control Systems
10.15.A.126	ZDEOKO04SCOM05		WIESEMANN & THEIS GMBH
10.15.A.127	Y100PSROED07	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.128	Y100AG42	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.130	Y100AG63	Windows 10	Fujitsu Technology Solutions GmbH
10.15.A.132	Y100AG64	Windows 10	Fujitsu Technology Solutions GmbH
10.15.A.14	Y100AH69	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.140	Y100AC15.local	Windows 7 SP1	Fujitsu Technology Solutions GmbH

IP	Name	Operating System	Vendor
10.15.A.144	TPC61		ADVANTECH CO., LTD.
10.15.A.145	10.15.A.145		
10.15.A.146	TPC61		ADVANTECH CO., LTD.
10.15.A.147	Y100AE04	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.148	Y100AE05	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.15	Y100AJ57	Windows 10 / Server 2016 / Server 2019	KONTRON COMPACT COMPUTERS AG
10.15.A.152	10.15.A.152		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.164	10.15.A.164		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.165	10.15.A.165		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.166	10.15.A.166		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.168	10.15.A.168		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.17	Y100AC70	Windows 7 SP1	Fujitsu Technology Solutions GmbH
10.15.A.170	10.15.A.170		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.171	10.15.A.171		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.172	10.15.A.172		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.173	10.15.A.173		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.177	COMSERVER-<wut1>		WIESEMANN & THEIS GMBH
10.15.A.178	COMSERVER-<wut1>		WIESEMANN & THEIS GMBH
10.15.A.179	10.15.A.179		WIESEMANN & THEIS GMBH
10.15.A.181	TPC61		ADVANTECH CO., LTD.
10.15.A.182	TPC61		ADVANTECH CO., LTD.
10.15.A.183	TPC61		ADVANTECH CO., LTD.
10.15.A.184	TPC61		ADVANTECH CO., LTD.
10.15.A.185	TPC61		ADVANTECH CO., LTD.
10.15.A.186	Y100AE46	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH

IP	Name	Operating System	Vendor
10.15.A.19	Y100PHT02	Windows 10 / Server 2016 / Server 2019 / Server 2016	ads-tec GmbH
10.15.A.190	Y100PROEDZ1	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.191	Y100PROEDZ2	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.192	Y100PROEDZ3	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.2	Y100AD05	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.20	Y100AF83	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.21	10.15.A.21		
10.15.A.23	Y100AJ32	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.24	Y100AE42	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.25	Y100AK17	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.4	10.15.A.4		DR. JOHANNES HEIDENHAIN GmbH
10.15.A.41	Y100AK09	Windows 10 / Server 2016 / Server 2019	KONTRON COMPACT COMPUTERS AG
10.15.A.50	Y100AK34	Windows 10 / Server 2016 / Server 2019	KONTRON COMPACT COMPUTERS AG
10.15.A.52	Y100PSROED02	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.53	10.15.A.53		Hurco Automation Ltd.
10.15.A.54	10.15.A.54		Hurco Automation Ltd.
10.15.A.55	Y100PSROED03	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.56	ZONCM092	Windows XP SP3	DR. JOHANNES HEIDENHAIN GmbH

IP	Name	Operating System	Vendor
10.15.A.57	Y100PDMU4	Windows 2000 SP1/SP2/SP3/SP4	DR. JOHANNES HEIDENHAIN GmbH
10.15.A.59	Y100AD48	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.66	10.15.A.66		ADVANTECH CO., LTD.
10.15.A.69	10.15.A.69		ADVANTECH CO., LTD.
10.15.A.70	10.15.A.70		
10.15.A.73	TPC61		ADVANTECH CO., LTD.
10.15.A.76	Y100POR029	Windows 7 SP1 / Server 2008 R2 SP1	
10.15.A.77	10.15.A.77		WIESEMANN & THEIS GMBH
10.15.A.8	Y100AD18	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.81	Y100AC31	Windows 7 SP1 / Server 2008 R2 SP1	Fujitsu Technology Solutions GmbH
10.15.A.82	Y100AC84	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.83	Y100AH73	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.88	Y100AD71	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.89	Y100AD64	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.9	Y100AG09	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.A.90	Y100PSROED05	Windows 10 / Server 2016 / Server 2019 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.A.99	Y100AB98	Windows 10 / Server 2016 / Server 2019 / Server 2016	Fujitsu Technology Solutions GmbH
10.15.C.1	Y100PIBF13	Windows 10 / Server 2016	
10.15.C.10	Y100PIBF33	Windows XP	Siemens AG
10.15.C.11	Y100PIBFE1	Windows 2000	PORTWELL, INC.
10.15.C.12	Y100PACE01	Windows XP	SIEMENS AG

IP	Name	Operating System	Vendor
10.15.C.13	Y100AC23	Windows 7 SP1	Fujitsu Technology Solutions GmbH
10.15.C.15	Y100AE57.local	Windows 10	Intel Corporate
10.15.C.16	Y100PIBF12	Windows 10 / Server 2016	SIEMENS AG
10.15.C.17	Y100PIBF08	Windows 10 / Server 2016	
10.15.C.18	Y100PIBF10	Windows XP SP3	Siemens AG A&D ET
10.15.C.19	Y100PIBFE2		
10.15.C.2	Y100PIBF09	Windows XP	Siemens AG A&D ET
10.15.C.20	Y100AJ14	Windows 10 / Server 2016 / Server 2019	Fujitsu Technology Solutions GmbH
10.15.C.22	Y100PIBFE3		
10.15.C.3	Y100PIBF03	Windows 10 / Server 2016	SIEMENS AG
10.15.C.35	Y100PIBF35	Windows 10 / Server 2016	SIEMENS AG
10.15.C.4	SINUMERIK PCU50.3	Windows XP SP3	SIEMENS AG
10.15.C.5	SINUMERIK PCU50.3	Windows XP SP2	Siemens AG A&D ET
10.15.C.7	Y100PIBF07	Windows XP	Siemens AG A&D ET
10.15.C.9	Y100PIBF11	Windows 10 / Server 2016	SIEMENS AG

Table K.4: VLAN 988 and 967 Nozomi Guardian asset identified

Appendix L

Vulnerabilities finding

	Vulnerabilities count			Vulnerabilities count	
Device	**Cisco**	**Nozomi**	**Device**	**Cisco**	**Nozomi**
10.14.D.24	8	7	10.15.A.8	21	
10.14.D.25	8	7	10.15.A.81	21	
10.14.D.26	8	7	10.15.A.82	21	2464
10.14.D.27	8	7	10.15.A.83	21	
10.14.D.28	8	7	10.15.A.88	21	
10.14.D.30	8	7	10.15.A.9	21	
10.14.D.31	8	7	10.15.A.90	21	
10.14.D.32	8	7	10.15.A.92	21	
10.14.D.33	8	7	10.15.A.99	21	
10.14.D.61	5	4	10.15.B.147	21	
10.14.D.62	8	7	10.15.B.153	21	
10.14.D.111	21		10.15.B.154	21	1956
10.14.D.185	21	2475	10.15.B.155	21	
10.14.D.199	21	2458	10.15.B.160		1956
10.14.D.212	21		10.15.B.163	21	
10.14.D.215	21		10.15.B.164	21	
10.14.D.220		1956	10.15.B.165	21	
10.14.D.221		1956	10.15.B.166	21	
10.14.D.223		1956	10.15.B.167	21	
10.14.D.225	21		10.15.B.180	21	
10.14.D.230	21		10.15.B.181	21	
10.15.A.10	21		10.15.B.183	21	2458
10.15.A.101	21		10.15.B.240	21	

	Vulnerabilities count	
Device	**Cisco**	**Nozomi**
10.15.A.110	21	
10.15.A.115	31	686
10.15.A.119	21	
10.15.A.12	21	
10.15.A.120	21	
10.15.A.122	21	
10.15.A.123	21	
10.15.A.124	21	
10.15.A.125	21	
10.15.A.127	21	
10.15.A.128	21	
10.15.A.130	21	
10.15.A.132	21	
10.15.A.14	21	
10.15.A.140	21	1950
10.15.A.147	21	
10.15.A.148	21	
10.15.A.15	21	
10.15.A.17	21	1950
10.15.A.186	21	
10.15.A.19	21	
10.15.A.190	21	
10.15.A.191	21	
10.15.A.192	21	
10.15.A.2	21	
10.15.A.20	21	
10.15.A.23	21	
10.15.A.4	21	
10.15.A.52	21	
10.15.A.55	21	
10.15.A.56	30	536
10.15.A.57	21	478
10.15.A.59	21	
10.15.A.76	21	
10.15.A.79	21	

	Vulnerabilities count	
Device	**Cisco**	**Nozomi**
10.15.B.241	21	
10.15.B.242	42	
10.15.B.243	21	
10.15.B.244	21	
10.15.B.245	21	
10.15.B.63	21	1956
10.15.B.71	21	
10.15.B.72	21	
10.15.B.73	21	
10.15.B.79	21	
10.15.B.85	21	
10.15.B.88	21	
10.15.B.89	21	
10.15.B.90	21	
10.15.C.1	21	
10.15.C.10	31	686
10.15.C.11	21	478
10.15.C.12	21	686
10.15.C.13	21	1950
10.15.C.15	21	2464
10.15.C.16	21	
10.15.C.17	21	
10.15.C.18	30	536
10.15.C.19	21	
10.15.C.2	30	686
10.15.C.20	21	
10.15.C.21	21	
10.15.C.3	21	
10.15.C.35	21	
10.15.C.4		536
10.15.C.5	30	596
10.15.C.6	30	
10.15.C.7	31	686
10.15.C.9	21	

	Vulnerabilities count			Vulnerabilities count	
Device	**Cisco**	**Nozomi**	**Device**	**Cisco**	**Nozomi**

Table L.1: Number of vulnerabilities found per each tool

ID	Description	Score	Found	
			Cisco	Nozomi
CVE-2015-8214	The implemented access protection level enforcement of the affected communication processors (CP) could possibly allow unauthenticated users to perform administrative operations on the CPs if network access (port 102/TCP) is available and the CPs' configuration was stored on their corresponding CPUs.	9.8	•	•
CVE-2016-8672	Siemens has released an advisory regarding vulnerabilities affecting SIMATIC CP 343-1 Advanced/CP-443-1 Advanced devices and SIMATIC S7-300/S7-400 CPUs. Inverse Path auditors and the Airbus ICT Industrial Security team reported these vulnerabilities directly to Siemens. Siemens has made new firmware versions available for several products and a temporary fix for the remaining affected products to mitigate these vulnerabilities. These vulnerabilities could be exploited remotely.	5.3	•	•
CVE-2016-8673	Siemens has released an advisory regarding vulnerabilities affecting SIMATIC CP 343-1 Advanced/CP-443-1 Advanced devices and SIMATIC S7-300/S7-400 CPUs. Inverse Path auditors and the Airbus ICT Industrial Security team reported these vulnerabilities directly to Siemens. Siemens has made new firmware versions available for several products and a temporary fix for the remaining affected products to mitigate these vulnerabilities. These vulnerabilities could be exploited remotely.	8.8	•	•
CVE-2017-2680	Multiple Denial of Service vulnerabilities could cause the targeted device to enter a denial-of-service condition, which may require human interaction to recover the system.	6.5	•	•
CVE-2017-2681	Specially crafted PROFINET DCP packets sent on a local Ethernet segment (Layer 2) to an affected product could cause a denial of service condition of that product. Human interaction is required to recover the system. PROFIBUS interfaces are not affected. This vulnerability affects only SIMATIC HMI Multi Panels and HMI Mobile Panels, and S7-300/S7-400 devices.	6.5		•

ID	Description	Score	Found	
			Cisco	Nozomi
CVE-2018-4843	A vulnerability has been identified in SIMATIC CP 343-1 Advanced (All versions), SIMATIC CP 343-1 Standard (All versions), SIMATIC CP 443-1 Advanced (All versions), SIMATIC CP 443-1 Standard (All versions), SIMATIC S7-1500 Software Controller incl. F (All versions <V1.7.0), SIMATIC S7-1500 incl. F (All versions <V1.7.0), SIMATIC S7-300 incl. F and T (All versions), SIMATIC S7-400 H V6 (All versions), SIMATIC S7-400 PN/DP V6 Incl. F (All versions <V6.0.7), SIMATIC S7-400 PN/DP V7 Incl. F (All versions), SIMATIC S7-410 (All versions <V8.1), SIMATIC WinAC RTX 2010 incl. F (All versions), SINUMERIK 828D (All versions <V4.7 SP6 HF1), Softnet PROFINET IO for PC-based Windows systems (All versions). Responding to a PROFINET DCP request with a specially crafted PROFINET DCP packet could cause a Denial-of-Service condition of the requesting system. The security vulnerability could be exploited by an attacker located on the same Ethernet segment (OSI Layer 2) as the targeted device. Successful exploitation requires no user interaction or privileges and impacts the availability of core functionality of the affected device. A manual restart is required to recover the system. Siemens provides mitigations to resolve the security issue. PROFIBUS interfaces are not affected.	6.5	•	•
CVE-2019-13946	PROFINET-IO (PNIO) stack versions prior v06.00 do not properly limit internal resource allocation when multiple legitimate Diagnostic package requests are sent to the DCE-RPC interface. This could lead to a denial-of-service condition due to lack of memory for devices that include a vulnerable version of the stack.	7.5	•	•
CVE-2020-25242	Specially crafted packets sent to TCP port 102 could cause a Denial-of-Service condition on the affected devices. A cold restart might be necessary in order to recover.	7.5	•	

ID	Description	Score	Found	
			Cisco	Nozomi
CVE-2021-33737	Sending a specially crafted packet to Port 102/TCP of an affected device could cause a denial-of-service condition. A restart is needed to restore normal operations.	7.5	•	

Table L.2: Vulnerabilities detail for device with IP address 10.14.D.33 - Found by both tools

Appendix M

Port scan events

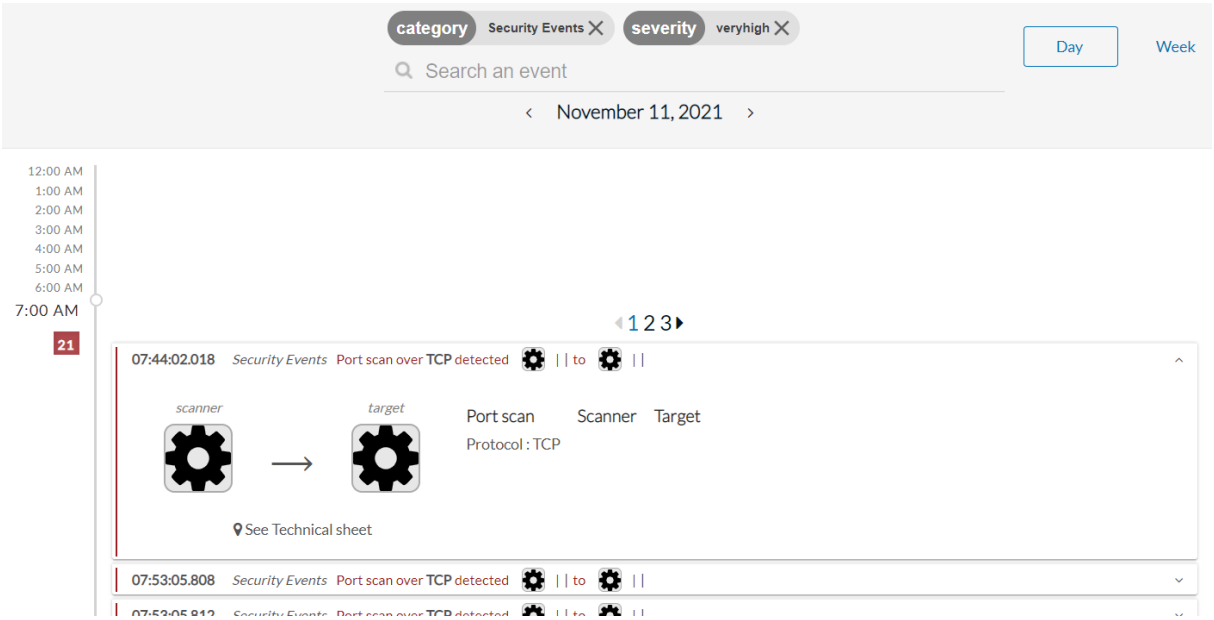


Figure M.1: Cisco Cyber Vision Port Scan event

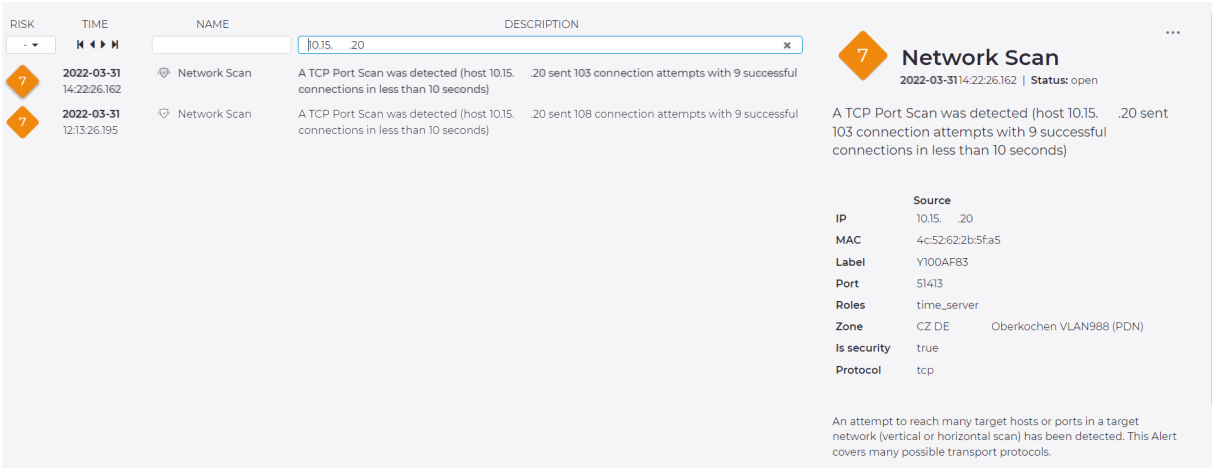


Figure M.2: Nozomi Guardian Port Scan event

Appendix N

Other tools events

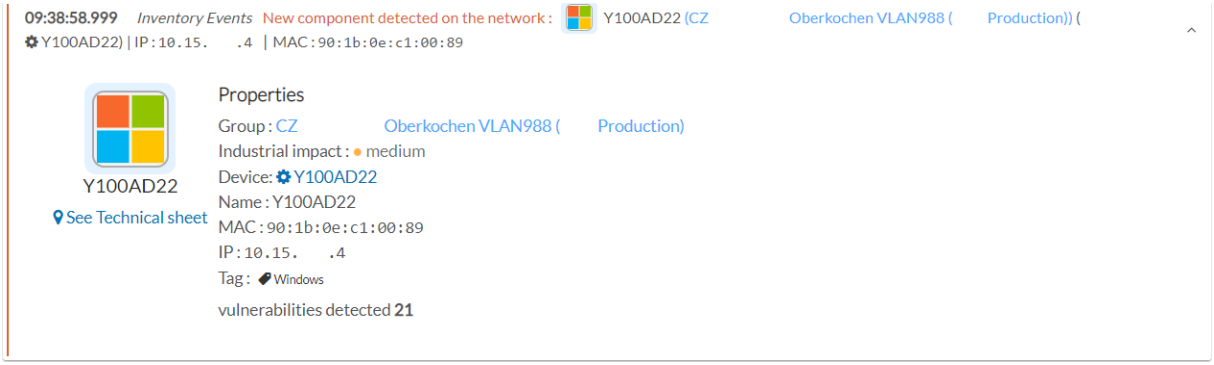


Figure N.1: Cisco Cyber Vision new device detected event

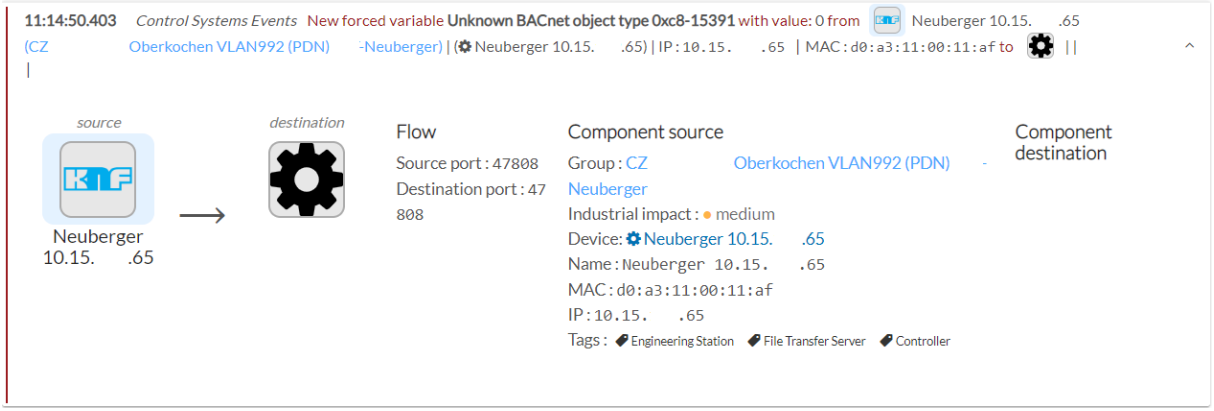


Figure N.2: Cisco Cyber force variable event

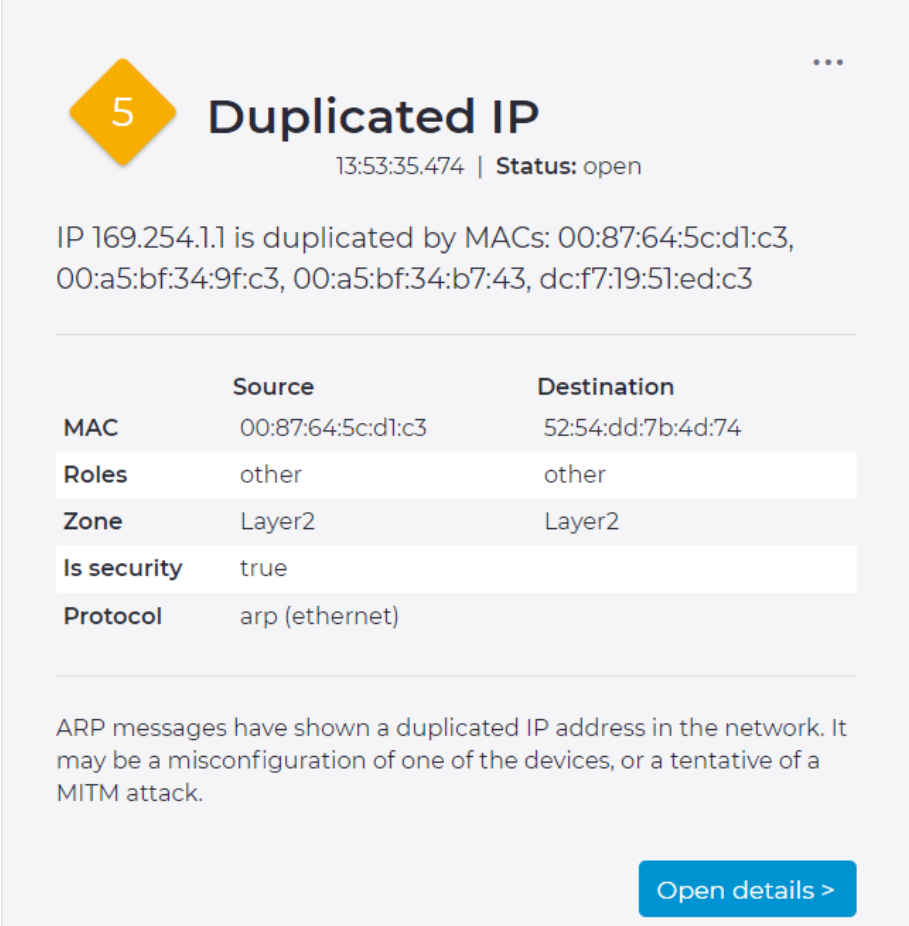


Figure N.3: Nozomi Guardian duplicated IP event