# CTI-CRS: A CTI-Driven Cyber Range Scenario - An OT/ICS Case Study

**Eron BAJRAKTARI**

Master thesis submitted under the supervision of
Prof. Thibault DEBATTY

the co-supervision of
Elli MAKDIS ANTOUN

in order to be awarded the Degree of
Master in Cybersecurity
Cryptanalysis and Forensics

Academic year
2025 - 2026

I hereby confirm that this thesis was written independently by myself without the use of any sources beyond those cited, and all passages and ideas taken from other sources are cited accordingly.



15/08/2026

**Title: CTI-CRS: A CTI-driven Cyber Range Scenario - An OT/ICS case**

Author: Eron BAJRAKTARI
Master in Cybersecurity – Cryptanalysis and Forensics
Academic year: 2025 - 2026

## Abstract

Critical infrastructure is increasingly being targeted by sophisticated cyber attacks, and the convergence of IT and OT environments has widened the attack surface of Operational Technology (OT) and Industrial Control Systems (ICS). Training and research for defending these systems increasingly rely on cyber ranges, yet existing platforms focus largely on IT scenarios and rarely ground their exercises in real adversary behavior derived from cyber threat intelligence (CTI). This thesis addresses that gap by developing a methodology for translating CTI into realistic and feasible OT/ICS attack scenarios for cyber ranges. The approach reconstructs an attack from available intelligence using the SANS ICS Kill Chain, scopes the relevant tactics, techniques, and procedures (TTPs), and derives a corresponding range architecture and an intended, manually executable attack scenario, while balancing realism against the practical constraints of reproducing proprietary OT systems. The methodology is demonstrated by applying it to the TRITON campaign: the attack is reconstructed, a corresponding IT/OT range is designed and deployed, and the intended attack path is executed to reach the objective of manipulating a Safety Instrumented System (SIS), for which a custom simulator was developed. The results show that available CTI can be translated into an executable and faithful reproduction of a documented campaign.

**Keywords:** Cybersecurity, CTI, Cyber Range, OT/ICS, Red Teaming

# Preface

During the preparation of this thesis, artificial intelligence (AI) tools were used in a supporting capacity for two distinct purposes: assisting with the development of the accompanying code, and assisting with the writing of the thesis document.

For the code developed as part of this work, Claude Opus 4.8 (Anthropic) was used to assist with code structuring, reformatting, and the translation of legacy code into a maintained, up-to-date version.

For the thesis text, Claude Sonnet 5 and Claude Opus 4.8 (Anthropic) were used to assist with grammar correction and the rephrasing of sentences for clarity and academic tone.

In both cases, all ideas, analyses, technical decisions, and conclusions presented in this thesis are the author's own; AI tools were not used to generate original content, arguments, or findings, and all AI-assisted output was critically reviewed, verified, and edited by the author prior to inclusion. The author takes full responsibility for the accuracy and integrity of the final text and implementation.

# Acknowledgements

This thesis marks the final step in the pursuit of my degree in *Master of Cybersecurity with a focus in Cryptanalysis and Forensics*. It marks the completion of a two-year journey into the field of cybersecurity, following a drastic shift away from my original field of Mathematics, and lays the foundation for what I hope will be a long career ahead.

I would like to express my thanks to Professor Debatty for supporting me in the choice of this topic and for being the supervisor of my thesis and internship.

Many thanks to Elli Makdis Antoun being my co-supervisor, continuously supporting my work, and for always being available to steer me in the right direction and assist me when needed.

My heartfelt appreciation goes out to my family, for their unwavering support in this academic journey. Since I was young, they have always pushed me towards academic excellence and I am eternally grateful for it.

Last but not least, my deepest appreciation goes out to my girlfriend for dealing with my shenanigans and proof-reading this thesis with a commitment I did not expect. She has now become an IT/OT convergence enthusiast :)

# Table of Contents

# List of Figures

# List of Tables

# List of Abbreviations

| | |
|---|---|
| AD | Active Directory |
| AI | Artificial Intelligence |
| APT | Advanced Persistent Threat |
| AV | Anti-Virus |
| C2 | Command and Control |
| CLI | Command-Line Interface |
| CTI | Cyber Threat Intelligence |
| DC | Domain Controller |
| DCS | Distributed Control System |
| ICS | Industrial Control Systems |
| IDMZ | Industrial Demilitarized Zone |
| IoC | Indicator of Compromise |
| IP | Internet Protocol |
| IT | Information Technology |
| OT | Operational Technology |
| RDP | Remote Desktop Protocol |
| RDS | Remote Desktop Services |
| SIEM | Security Information and Event Management |
| TCP | Transport Control Protocol |
| TTP | Tactics, Techniques, and Procedures |
| UDP | User Datagram Protocol |
| VLAN | Virtual Local Area Network |
| VM | Virtual Machine |

# Chapter 1
# Introduction

Modern critical infrastructure is increasingly characterized by the convergence of Information Technology (IT) and OT, which expands the attack surface and introduces significant security challenges. This exposure is particularly consequential because, in the current geopolitical landscape and modern warfare, critical infrastructure has become an attractive target for incapacitating an adversary [3]. This has been demonstrated on several occasions, namely the Stuxnet malware campaign targeting Iran's nuclear program [4], the TRITON attack on Safety Instrumented Systems at a petrochemical facility in Saudi Arabia [5], and the 2015 Ukraine Electric Power Attack [6]. These incidents highlight not only the significance of protecting critical infrastructure, but also the need for skilled personnel capable of doing so. While such attacks can have severe and disruptive consequences, they also generate valuable analytical insights, as intelligence can be derived from forensic artifacts and post-incident investigations.

More broadly, the evolving digital landscape reflects a continuous rise in cyber attacks [7]. This trend has driven both an increasing demand for skilled professionals and a surge in the production and availability of threat intelligence. In response, modern training programs increasingly incorporate cyber security exercises conducted in controlled, intentionally vulnerable environments known as cyber ranges, where personnel can develop both offensive and defensive capabilities.

However, while cyber security training platforms are widely available and extensively used, many primarily focus on IT-based scenarios involving vulnerabilities and misconfigurations. As a result, training for OT environments remains limited, particularly in the context of realistic, scenario-based exercises. Furthermore, existing platforms often lack the integration of up-to-date cyber threat intelligence, resulting in scenarios that do not provide an evidence base and justification of the architectural choices. This highlights the need for approaches that translate cyber threat intelligence into high-fidelity scenario environments, where adversary Tactics, Techniques, and Procedures (TTP) are accurately represented.

Modern ICS environments rely on the integration of IT and OT equipment to provide full inter-connectivity between different departments of an organization. While this increases capability and practicality, it provides a widened attack surface making it an attractive target for threat actors. Thus, there is a need for environments where professionals can train, research, and analyze real attacker behavior in order to provide better defense.

Currently available cyber range labs for training and research typically offer environments where vulnerabilities and misconfigurations can be exploited to reach a defined goal, but they rarely provide an environment to reproduce the actual behavior and attack steps of real adversaries. CTI, now an increasingly important element of security programs in modern organizations, directly documents this adversary behavior, making it a natural foundation for modeling scenarios in which real attack paths can be reproduced [8].

However, OT environments often include proprietary components from various vendors, whose inner workings and documentation are rarely available to the public. This makes them infeasible to be reproduced faithfully in a cyber range scenario. To solve this issue,

a balance must be struck between the realism of the environment and its feasibility.

In this context, this thesis aims to demonstrate how cyber threat intelligence can be translated into cyber range scenarios. Such environments enable users to reproduce these TTPs in a controlled setting, progressing through realistic attack paths to achieve defined objectives. Beyond training, these scenarios can also be leveraged for testing defensive capabilities and supporting research, thereby bridging the gap between intelligence, and practical application across both IT and OT domains. As attacks on critical infrastructure continue to evolve, the need for realistic training and research environments only grows, yet the environments currently available fall short in specific ways. The question this thesis aims to answer is:

**How can cyber threat intelligence be used to design realistic and feasible OT/ICS attack scenarios for cyber-range environments?**

To address this question, this thesis makes the following contributions:

- Methodology for the translation of CTI into a cyber range scenario
- A threat actor profile of TEMP.Veles
- A SANS ICS Kill chain reconstruction of the TRITON attack campaign
- Design and implementation of a testbed that enables the reproduction of TEMP.Veles TTPs from the TRITON attack campaign
- A Safety Instrumented System (SIS) component for the open-source project GRFICSv2 that simulates an OT process

This thesis is structured into eight chapters:

- **Chapter 1 - Introduction** provides the context, motivation, problem statement, and contributions
- **Chapter 2 - Background and state of the art** introduces the fundamental concepts of an OT/ICS environment, explores the current landscape of CTI and cyber ranges, and frameworks for attack behavior
- **Chapter 3 - Methodology** outlines the selection of the CTI reports, the selection of the campaign, the kill chain mapping method, and the design approach
- **Chapter 4 - Cyber Threat Intelligence of the TRITON campaign** provides a detailed account of the threat campaign, the threat actor, and the attack reconstruction through ICS Kill Chain mapping
- **Chapter 5 - Scenario Design and Architecture** presents CTI and the context from the TRITON attack contribute to the range design
- **Chapter 6 - Cyber-range Implementation** details the architecture components, the simulation, and the implementation steps
- **Chapter 7 - Scenario Execution & Discussion** provides a walk-through of the cyber range and reflects on the scenario and the balance between realism and feasibility of the created scenario
- **Chapter 8 - Conclusion & Future Works** summarizes the contribution of the thesis and proposes future enhancements and future directions for the scenario

# Chapter 2
# Preliminaries and state of the art

## 2.1 OT/ICS fundamentals

OT differs from IT mostly in its purpose. While IT's main purpose is the management and transmission of data, OT controls physical processes, such as manufacturing processes, energy grid infrastructure, and chemical processes [1]. The main driver towards the convergence of both environments is the neccesity of real-time information from the OT to the IT environment for reasons such as production scheduling, inventory management, and cost calculations among others [1]. In modern organizations, both OT and IT are increasingly interconnected, enabling communication and data exchange between the two domains. While this convergence enhances operational efficiency and functionality, it also increases the organization's attack surface [9].

Key components of an ICS environment include Programmable Logic Controllers (PLC), Human-Machine Interface (HMI), Safety Instrumented Systems (SIS), and Engineering Workstations among others. PLCs serve as the primary control element in every ICS, by taking data from sensors and controlling actuators. Within an ICS, the HMI serves as the principal visualization component, presenting process data, control values, and alarms to operators [10]. SIS are the last line of defense in ICS environments. Their role is to constantly monitor the ICS processes, and when detecting an unsafe state such as a dangerously rising pressure in a chemical process, to bring it back to a safe state or shut down the system completely [11]. Lastly, engineering workstations are where operators can configure the components from. Together, these components form a Distributed Control System (DCS): an integrated set of subsystems that jointly perform a specific task. Communication between OT components is enabled by various protocols, including the widely used Modbus, a publicly available communication protocol [12]. At the same time, some vendors develop and commercialize their own proprietary protocols.

Importantly, the OT environment is inherently insecure [13]. The devices and protocols have been designed with availability in mind, without necessarily considering integrity and confidentiality. For example, the Modbus protocol offers no encryption, authentication, or authorization, indicating that it is highly susceptible to man-in-the-middle attacks [12]. Furthermore, the convergence of IT and OT has broadened the attack surface, as vendors and engineers have increasingly adopted IT protocols, such as Ethernet, Internet Protocol (IP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP), as the underlying transport for OT communications. This convergence improves interoperability and efficiency, but it also exposes formerly isolated OT devices to remote network access, widening the attack surface [1].

Taken together, these characteristics define the security posture of modern ICS environments. The architecture is functionally layered, mapping onto the Purdue model, yet it rests on components and protocols that were engineered for availability and reliability rather than security. The prevalence of protocols such as Modbus, which lack encryption and authentication, combined with the growing convergence of IT and OT, leaves these systems exposed to a broad range of threats. Understanding this inherent insecurity motivates the need for realistic cyber ranges, in which these environments can be safely replicated to

study both attacks and defensive measures without risk to operational systems.

## 2.2 Purdue Model

To model an OT/ICS network, an industry-adopted reference model called the Purdue model is used [14]. It illustrates where different components of the ICS network are located based on their function and role. The Purdue model uses four distinct zones and seven levels to model ICS networks.

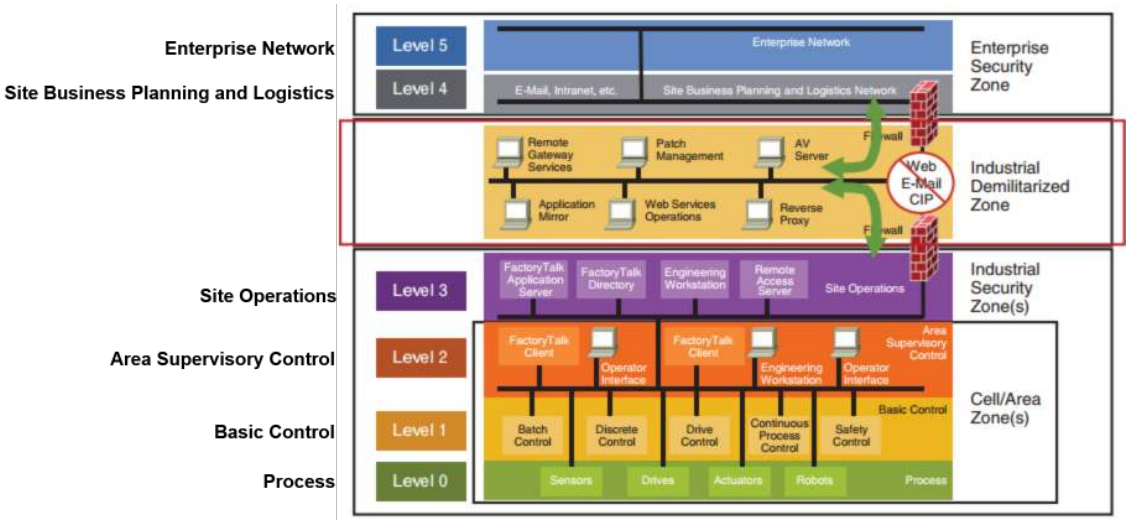


Figure 2.1: OT/ICS environment architecture based on Purdue Model [1]

The Enterprise Zone comprises Levels 5 and 4. Level 5, the Enterprise Network, hosts corporate systems with a business purpose, using data acquired from the plant to report on overall production status. Below it, Level 4, Site Business Planning and Logistics, houses systems that support production, such as supervisor desktops and databases. Between the enterprise and industrial domains lies the IDMZ (Level 3.5), which contains intermediary systems such as proxies, web servers, update servers, and broker services.

The Industrial Zone spans Levels 0 to 3. At Level 3, Site Operations, are the systems that directly support the production processes, such as operator workstations, HMIs, and servers that report data to the Enterprise Zone. Level 2, Area Supervisory Control, mirrors Level 3 but at a narrower scope, dedicated to a smaller part of the overall system. Level 1, Basic Control, is where the systems that directly control the ICS processes reside, such as PLCs. Finally, Level 0, the Process level, comprises the physical systems, for example sensors, actuators, and robots.

Linking back to Section 2.1, the components described map directly onto the Purdue Model. PLCs reside at Level 1 (Basic Control), as they directly control the process, while some controllers with a supervisory role sit at Level 2. The SIS also resides at Level 1, since it independently monitors the process and acts to restore a safe state when an unsafe condition is detected. Supervisory systems belong to Level 2, and engineering workstations fall at Level 2 or 3, depending on whether they serve a specific system or a larger part of the plant.

## 2.3 Frameworks

### 2.3.1 MITRE ATT&CK

ATT&CK is a knowledge base developed by the MITRE organization which can be used to model adversary behavior throughout an attack life cycle [15]. It is structured through Tactics, Techniques and Procedures [16]:

- Tactics represent the immediate goal of a Technique used by the attacker
- Techniques describe how the attacker achieves their tactical goal. Sub-techniques further break down techniques into more granular behavior
- Procedures are specific implementations of the techniques used by the attacker

Furthermore, the ATT&CK knowledge base comprises three distinct matrices: Enterprise, Mobile, and ICS. This separation reflects fundamental differences in adversarial behavior across domains; for instance, the ICS matrix contains tactics and techniques with no Enterprise equivalent (e.g., Impair Process Control, or techniques targeting physical processes), necessitating a dedicated matrix rather than an extension of the Enterprise matrix [17]. Although these knowledge bases are structurally distinct, they can nonetheless be used in conjunction to characterize different phases of a single attack.

The ATT&CK framework is widely used in Cyber Threat Intelligence as a common language used to profile threat actors, detail their attack behavior in different campaigns, and compare different threat actors or campaigns.

### 2.3.2 SANS ICS Kill Chain

Traditionally, to illustrate the necessary steps undertaken by an attacker to achieve their objectives, the Lockheed Martin Cyber Kill Chain is used [18]. It represents the seven stages of an Advance Persistent Threat (APT), with each preceding stage enabling the following and towards an objective. While this framework is widely used, and portrays IT cyber attacks faithfully, it does not fully capture the characteristics of an ICS cyber attack.

Attacks on ICS differ from IT intrusions in attacker intent, capability and familiarity with the environment. ICS attacks often require specialized knowledge of proprietary systems and physical processes. To demonstrate this in a kill chain, researchers from the SANS institute introduced the concept of the ICS Kill Chain [2]. Stage 1 of the SANS ICS Kill Chain closely mimics the Lockheed Martin Cyber Kill Chain. It consists of five phases, namely "Planning", "Preparation", "Cyber Intrusion", "Management & Enablement", and "Sustainment, Entrenchment, Development & Execution". The Planning phase is composed of the Reconnaissance step, which is activity to gain information on the target environment. The Preparation phase includes Weaponization and Targeting steps, which involve preparing a malicious deliverable, such as a seemingly benign file that conceals malicious functionality, and selecting the intended target. The Cyber Intrusion phase involves the process of Delivery, Exploit, and Install/Modify steps. These three steps portray the process the attacker uses to deliver the file in the target environment, the means the attacker uses to perform malicious actions, and the process of installing or modifying available capabilities, such as deploying a Remote Access Trojan. The Management & Enablement phase involves C2 communications, which is the way the attacker will ensure continuous access and communication with the target environment. Lastly,

the Sustainment, Entrenchment, Development & Execution entails the Act step. There are many different actions an attacker can undertake in this step, ranging from Discovery of assets, moving laterally in the environment, capturing traffic, collecting and exfiltrating information, and cleaning any traces. The goal of this stage is often to access OT environment devices, which in turn will set the stage for Stage 2 of the attack.

Stage 2 is where the attacker uses the knowledge gained in Stage 1 to carry out the actual ICS attack. Stage 2 consists of three phases, namely the "Attack Development & Tuning", "Validation", and "ICS Attack". It is important to note that the first two phases of the kill chain do not necessarily take place in the live environment which they have infiltrated. In the "Attack Development & Tuning" phase, the attacker uses the information gathered to develop a capability tailored to the specific target systems and protocols to affect the ICS in the desired way. In the Validation phase, the attacker tests their new capability on a similar or identically configured system. Ultimately, the ICS Attack phase consists of delivering the capability, installing or modifying the existing system functionality, and lastly executing this attack. The attack execution can be categorized into enabling, initiating or supporting to achieve the desired effect on the ICS.

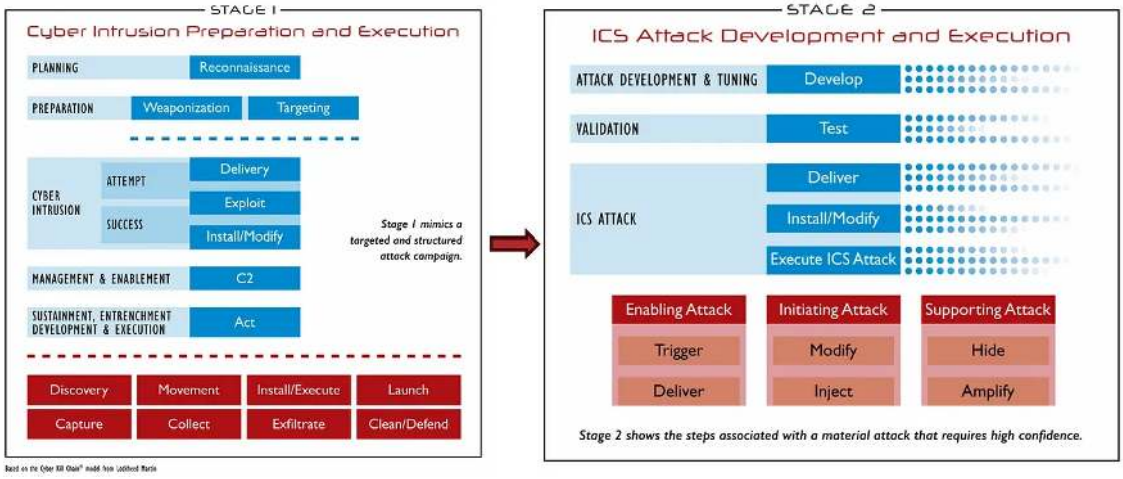


Figure 2.2: The SANS ICS Kill Chain illustrating Stage 1 and Stage 2 and their respective steps [2]

## 2.4 Cyber ranges

Cyber ranges have become a well-established tool for training, testing, and research [19]. They offer an environment that replicates the networks, systems, and tools found in real organizations, where users can train offensive and defensive skills, test defenses against attacks, and conduct cyber security research. While the concept is not newly developed, its importance and relevance grows in step with the increase of the rising number and sophistication of cyber attacks.

Although the majority of the available cyber ranges target IT environments, a smaller but growing number of cyber ranges target OT and ICS environments [20]. The complexity of OT and ICS cyber ranges lies in the feasibility of recreating such an environment. OT components are supplied by multiple competing vendors, and as a result, detailed information regarding their internal operation and communication protocols is rarely publicly available. This scarcity of information poses a central challenge for reproducing OT

environments faithfully in a cyber range.

## 2.5 State of the art

### 2.5.1 CTI-driven cyber ranges and scenarios

Thus far, research endeavors on CTI-driven cyber range scenarios focuses largely on automating the generation process of cyber range environment scenarios. In their survey, Garg et al. [21] summarize existing approaches and describe the common CTI pipeline: CTI data is ingested automatically from various threat feeds and passed into a system that generates the scenario with minimal manual intervention.

The CTI data typically includes Indicators of Compromise (IoC), TTPs, malware names, and exploited vulnerabilities. The automated pipeline takes the CTI, and either fills predefined templates with domain names, IPs, and malware names [22], or take TTPs such as **T1566 - Phishing** and create an automated script that sends a phishing mail [23]. Another emerging technique is to use Artificial Intelligence (AI) to generate scenarios from reports or natural language, without necessarily deploying the complete range [24].

While these techniques make effective use of CTI, they pay little attention to the context surrounding the campaigns, such as the environment the attack took place in, or the components of the network that were compromised. Additionally, this lack of context is even more evident in scenarios that merge IT and OT environments. Apart from the context, the survey also indicates that the more the solution is automated, the wider the trade-off between reality and feasibility grows OR the more automated the solution, the wider the trade-off between reality and feasibility.

### 2.5.2 OT testbeds and scenarios

Turning from CTI-driven approaches to the environments themselves, a number of OT and ICS testbeds have been proposed by researchers and companies. Their collective aim is to provide a testbed for practicing attacks, testing defenses, and conducting research. Approaches in their implementation vary from fully virtualized approaches to hybrid models.

CR-ICS is a cyber range that proposes a simulation of an industrial gas turbine [25]. It relies on full virtualization of the OT components for simulation. The authors propose an intended scenario and attack steps for the environment, but they do not mention any specific basis from real attacks or threat actors. While the scenario is legitimate and follows a logical kill chain, it lacks the context of a real historic attack or real attacker behavior.

SWaT and LICSTER are testbeds that aim to emulate ICS environments for attack and defense purposes [26, 27]. These two solutions propose a hybrid approach that uses virtualization and hardware to create the range. While LICSTER aims to be low-cost and uses minimal hardware, SWaT uses real OT components such as PLCs and water treatment equipment. The use of hardware in this case contributes to the realism of a cyber range, but trades off practicality by making them less portable and reusable elsewhere. The authors of both ranges describe attacker models for their respective ranges, however, as previously noted, there is a lack of use of any CTI. Even if loosely inspired by real incidents, the absence of an explicit, documented CTI basis means the scenarios cannot be traced to or validated against real adversary behavior.

Lastly, GRFICS is an open-source cyber range developed by Fortiphyd [28]. GRFICSv2 simulates an ICS environment and is composed of a PLC, HMI, workstation, a process

simulation, and a router each in its own Virtual Machine (VM). Additionally, GRFICSv3 proposes the same components, but with the simulation containing a 3D environment and also a MITRE CALDERA server [29]. The project is silent on any attacker model or any specific scenario to follow, though it offers a complete environment to practice. It is characterized by the use of open-source software and is fully virtualized, and offers a graphical simulation showing the process in action. GRFICS provides a capable environment, yet it leaves both scenario design and any grounding in real adversary behavior entirely to the user.

| **Name** | **Implementation** | **IT/OT Integration** | **Scenario basis** | **CTI-Driven?** |
|---|---|---|---|---|
| CR-ICS [25] | Virtualized | Yes | Generic attacker model | No |
| SWaT [26] | Physical | No | Generic attacker model | No |
| LICSTER [27] | Hybrid | No | Generic attacker model | No |
| GRFICS [28] | Virtualized | No | None | No |
| **CTI-CRS** | Virtualized | Yes | Threat Actor Profile, CTI reports | Yes |

Table 2.1: Comparison of implementation, IT/OT integration, scenario basis, and CTI-driven design across OT cyber ranges/testbeds

As summarized in Table 2.1, none of the reviewed environments combine CTI-driven scenario design with a realistic OT/ICS environment, or reproduce a documented real-world campaign. It is precisely this gap, the translation of cyber threat intelligence into realistic, context-aware OT/ICS scenarios, that this thesis addresses.

# Chapter 3
# Methodology

This chapter presents the methodology followed to translate cyber threat intelligence into a deployable OT/ICS cyber range scenario. It is organized around a pipeline that begins with intelligence analysis and concludes with a deployed environment. Notably, this approach does not rely on, and neither does it aim to propose an automated approach. Figure 3.1 illustrates the pipeline which will be detailed in the following sections.

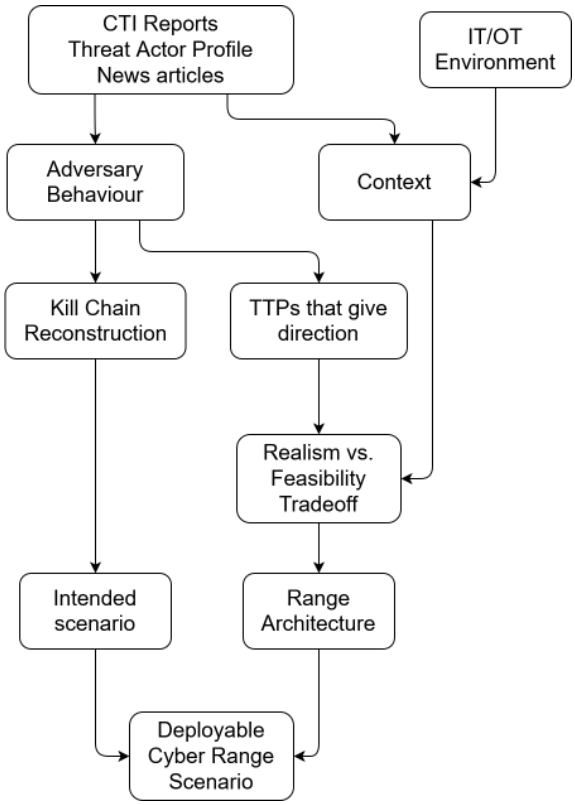


Figure 3.1: The methodology pipeline for translating cyber threat intelligence into a deployable cyber range scenario. Intelligence sources yield adversary behavior and context; behavior drives the kill chain reconstruction (producing the intended scenario) and the directional TTPs, which, with context and the realism-versus-feasibility trade-off, inform the range architecture. Scenario and architecture combine into the final deployable range

## 3.1 Cyber Threat Intelligence

The first step and basis of the whole approach is gathering available CTI. This includes Attack Campaign Reports, Threat Actor Profiles, and news articles. The list is not exhaustive, since information on attack behavior is derived through various resources, often not openly available to the public, such as paid information from vendors or information from the dark web. Against this background, two categories of information central to this approach are attacker behavior and context.

Attacker behavior can be extracted and mapped to the MITRE ATT&CK framework. In many publicly available CTI reports, this mapping has already been performed, allowing analysts to directly leverage the documented TTPs. Additionally, information from the actual attacks is sometimes not fully available due to the confidential nature of such incidents, so there may be gaps in TTPs for some of the Tactics. Although this may pose a challenge for the kill chain mapping, the missing techniques are often inferred from the surrounding context or the attacker's position at that stage, allowing the researcher to reason about plausible techniques for a given tactic. At the same time, context is crucial for the output to be faithful to the attacker and campaign. While CTI reports tend to be technical and focus on attack behavior, they may lack details surrounding the attack or the environment it took place in. This contextual information must therefore be collected from complementary sources, such as incident write-ups, sector or vendor reports, or general news coverage, to establish the nature of the target or the likely goal of the attacker.

## 3.2 Kill Chain Reconstruction

The first step in mapping the Kill Chain, specifically the SANS ICS Kill Chain, is understanding the attacker's objective and the steps required to achieve it. This mapping serves two purposes: it contributes to the intended scenario for the cyber range, and it establishes the evaluation direction, since success is defined as being able to act as the attacker and reach that objective within the environment. Each observed action is mapped to the kill chain phase it belongs to and to its corresponding ATT&CK technique, based on the objective the action serves and the method used to achieve it. It should be noted that information on behavior or context may be lacking for certain phases; for instance, it is not certainly known how the attacker gained initial access. While this is a limitation, the remaining phases can still be mapped, and missing steps can be reasoned about as discussed in section 3.1.

## 3.3 TTP Scoping

For the range to be realistic and faithful towards the attacker and the target environment, the design must draw directly on the CTI and context gathered. In this step, the CTI gives direction towards the final architecture, which is later combined with the context to produce the final result. Importantly, this step does not discard any TTPs; it extracts any architectural detail capable of being provided by each, although some contribute more than others.

A TTP is architecturally relevant to the extent that it constrains or reveals a specific property of the target environment. For example, taking **T1068 - Exploitation for Privilege Escalation** from the Tactic Privilege Escalation, then it it reveals little about the environment beyond the existence of an exploitable privilege-escalation vector. On the other hand if instead we consider **T1055.001 - Process Injection: Dynamic-link Library Injection** from the same Tactic, it tells us that the privilege escalation is done in an environment where there is at least one Windows Operating System (OS) since Dynamic-link Libraries are specific to the Windows OS environment. To take it one step further, we may then have **T1210 - Exploitation of Remote Services** from the Tactic Lateral Movement, which in combination with the Windows OS detail from the previous

technique, can tell us that Remote Desktop Services are available in the environment. Thus, while individual techniques each reveal a property of the environment, their combination constrains the architecture more precisely than any single technique alone.

A challenge arises when available TTPs are too broad, like **T1068**. The gaps create by these TTPs could still be addressed in through context previously gathered. For example, if the attack took place in a corporate environment, then the author can assume with a high degree of certainty that the environment with Windows operating systems. Although such inference introduces interpretation and reduces certainty, it still yields an environment in which the attacker's scenario can be reproduced, albeit with lower fidelity.

## 3.4 Realism versus Feasibility Tradeoff

Recreating an IT environment is readily achievable through virtualization, since it easily allows the deployment of the necessary machines and servers. The same is not true for OT components, which are most often hardware with closed documentation, making them difficult to virtualize and deploy. In addition, there are vendor-specific protocols that are used and are also not necessarily documented and found easily. For these reasons, a deliberate balance must be struck between realism and feasibility at this step.

While complete realism is difficult to achieve for an OT environment, open-source software can approximate a realistic scenario. A number of open-source tools exist for this purpose; OpenPLC, for instance, is a programming environment that allows a PLC to be simulated in software [30]. However, while open-source tools can reproduce some components, others, such as safety systems with proprietary protocols and hardware, cannot be identically simulated without acquiring the physical equipment. In these cases, a custom simulator can be developed, or the component omitted entirely, depending on how central it is to the attack path.

## 3.5 Range Architecture

At this step, the architecture is derived from three inputs: the CTI, the campaign context, and the realism-versus-feasibility balance. A third input also contributes at this stage: reference knowledge of how IT and OT environments are typically composed. Because standard environments follow well-established architectural patterns, the presence of one component often implies others by convention. For instance, if Windows is inferred to be present, a Windows Domain Controller is also likely; similarly, a controlled physical process necessarily implies the presence of a PLC. In the OT context, the Purdue Model serves as the primary reference, as it defines the components expected at each level of an industrial environment. These reference architectures allow the environment to be composed with components that the CTI itself may not explicitly mention, though such inferred components carry lower certainty than those directly documented in the intelligence.

## 3.6 Scenario Design

Alongside the range architecture, the methodology produces an intended scenario to be conducted by the user, so that the attacker behavior can be reproduced. The scenario is an ordered execution of TTPs, drawn from the previously reconstructed kill chain, that the user manually performs to progress and reach the defined objective.

Not all TTPs from the kill chain are included. This gives rise to two distinct selection criteria applied to the same set of TTPs: whereas architectural relevance depends on whether a TTP reveals a detail about the environment, scenario relevance depends on whether the TTP represents behavior performed inside the target environment. TTPs describing attacker-side preparation therefore fall outside the scope the scenario; for instance, **T1587.001 - Develop Capabilities: Malware** from the Tactic Resource Development, takes place on the attackers own infrastructure, thus it cannot be recreated in the range. Conversely, a TTP that does not contribute to the architecture, such as **T1068**, may still form an essential step in the scenario.

The scenario and the architecture are therefore co-designed: The architecture must expose the conditions that allow the TTPs to be executed, while the scenario defines the path the user follows in that architecture.

## 3.7 Evaluation Approach

The main goal of the methodology is to produce an environment faithful to the target from which the CTI is drawn, in which the attacker's behavior can be reproduced. Evaluation therefore addresses the two properties named in the research question: feasibility and realism.

Feasibility has two dimensions in this evaluation. The first, construction feasibility, concerns the practicality of building each component of the range. For instance, a physical Triconex safety controller is costly and difficult to obtain, whereas a custom simulator provides a feasible and portable alternative, at the cost of some realism. The second dimension, execution feasibility, concerns whether the intended scenario can be carried out to reproduce the attacker's behavior and reach the objective.

Realism is assessed by how faithfully the environment and the reproduced attack path correspond to the documented campaign. This includes the coverage of the reconstructed kill chain phases, the fidelity of the reproduced TTPs to their documented counterparts, and the correspondence of the architecture to the known characteristics of the target environment.

It should be noted that this evaluation approach has inherent limits. Because the environment is a reconstruction, incorporating inferred components and simulated systems, based on intelligence that may not always be complete, the evaluation can demonstrate that the scenario and range is realistic and feasible, but not that it is identical to the original incident.

# Chapter 4
# Threat Intelligence Analysis of the TRITON attack

To demonstrate the proposed methodology, a real-world attack campaign was selected as a case study. For a case to effectively exercise the propose method, the following conditions are evaluated:

- Availability of Cyber Threat Intelligence
- Two-stage structure spanning IT and OT
- Well-documented TTPs across Enterprise and ICS matrices of MITRE ATT&CK

The TRITON attack campaign satisfied the conditions, thus was chosen for demonstration. First, TRITON is supported by a substantial body of publicly available threat intelligence, which is essential for extracting context. Second, the campaign exhibits a clear two-stage structure spanning both IT and OT environments, making it a multidisciplinary range. Third, its TTPs are well-documented across the Enterprise and ICS matrices, enabling the use of both for mapping. Lastly, the target of the attack, the SIS, represents exactly the kind of proprietary component where the trade-off between realism and feasibility is most pronounced.

Stuxnet, while a landmark ICS attack, relies heavily on autonomous, self-propagating behavior driven by multiple zero-day exploits [31]. Thus, making it less suitable for a manually reproducible scenario, in which the user is intended to perform each step deliberately. Industroyer targets electrical grid infrastructure through specialized power-system protocols, the faithful reproduction of which would demand simulating grid components and communication standards that are difficult to represent with available tooling [32]. This places the case at an impractical extreme of the realism–feasibility trade-off. TRITON, by contrast, is a human-operated campaign with a clear two-stage IT/OT structure and a target that, while proprietary, can be approximated to a workable degree, making it demanding enough to exercise the methodology yet feasible enough to reproduce.

## 4.1 Incident Overview

In August of 2017, two emergency shutdown systems were activated in the Petro Rabigh petrochemical facility in Saudi Arabia [33]. After initial investigations, a malware named TRITON was discovered in the environment which interfered with the Triconex line of safety systems. The TRITON malware was specifically designed to hijack the Triconex SIS and create an unsafe environment that would allow a major accident to go through. The significance of this attack lay in the fact that it specifically targeted the SIS devices present in the OT environment, being the first of its kind.

From the available intelligence, it is not exactly known how the attackers gained initial access into the environment, but they gained access to corporate computers. Afterwards they moved laterally and gained remote access to the OT environment, specifically a SIS

engineering workstation, where they were able to gather information deployed the custom TRITON malware which enabled them to communicate with the Triconex SIS [5]. It is widely believed that the attackers inadvertently tripped a SIS controller, while modifying its application memory while using TRITON.

## 4.2 Threat Actor and Objectives

Further investigation by FireEye (now Mandiant, part of Google Cloud) assessed with high confidence, later supported by a U.S. Department of Justice indictment, that the attackers were linked to Russia, specifically to a research institution known as the Central Scientific Research Institute of Chemistry and Mechanics (CNIIHM) [34, 35]. This attribution was based in part on clues found within the malware code itself. The involvement of a nation-state-affiliated institution is consistent with the resources the attack required: identifying that the target used Triconex controllers, reverse-engineering the proprietary TriStation protocol, and developing malware capable of communicating with the safety controllers all demand capabilities characteristic of a well-resourced, state-backed actor.

Although the attack was not successful in achieving the probable objective the attacker had, the following attack options model illustrates the potential impact of this attack through the options available to an attacker who has hijacked a SIS [5]:

- Attack Option 1: Use the SIS to shutdown the process
  - The attacker can reprogram the SIS logic to cause it to trip and shutdown a process that is, in actuality, in a safe state. In other words, trigger a false positive.
  - Implication: Financial losses due to process downtime and complex plant start up procedure after the shutdown.
- Attack Option 2: Reprogram the SIS to allow an unsafe state
  - The attacker can reprogram the SIS logic to allow unsafe conditions to persist.
  - Implication: Increased risk that a hazardous situation will cause physical consequences (e.g. impact to equipment, product, environment and human safety) due to a loss of SIS functionality.
- Attack Option 3: Reprogram the SIS to allow an unsafe state – while using the DCS to create an unsafe state or hazard
  - The attacker can manipulate the process into an unsafe state from the DCS while preventing the SIS from functioning appropriately.
  - Implication: Impact to human safety, the environment, or damage to equipment, the extent of which depends on the physical constraints of the process and the plant design.

## 4.3 Attack Workflow and SANS ICS Kill Chain Mapping

Building on the overview, this section formally reconstructs the attack against SANS ICS Kill Chain, mapping each step to its phase along the corresponding ATT&CK technique. Table 4.1 and 4.2 describe all of the identified TTPs taken from the MITRE ATT&CK

description of the TRITON attack campaign, alongside evidence from various sources [36].

| ID | ATT&CK Tactic | MITRE Technique | Reference |
|---|---|---|---|
| T1595 | Reconnaissance | Active Scanning | The IP 87.245.143.140 has engaged in network reconnaissance against targets of interest to TEMP.Veles [34] |
| T1587.001 | Resource Development | Develop Capabilities: Malware | The attack framework was built to interact with Triconex Safety Instrumented System (SIS) controllers [5] |
| T1588.002 | Resource Development | Obtain Capabilities: Tool | TEMP.Veles' lateral movement activities used a publicly-available PowerShell-based tool, WMImplant [34] |
| T1059.001 | Execution | Command and Scripting Interpreter: PowerShell | TEMP.Veles' lateral movement activities used a publicly-available PowerShell-based tool, WMImplant [34] |
| T1053.005 | Execution/Persistence/Privilege Escalation | Scheduled Task/Job: Scheduled Task | A ZIP archive recovered during investigations, schtasks.zip, contained an installer and uninstaller of CATRUNNER that includes two versions of an XML scheduled task definitions for a masquerading service 'ProgramDataUpdater' [34] |
| T1036.005 | Stealth | Masquerading: Match Legitimate Resource Name or Location | The malware was named to masquerade as the legitimate Triconex Trilog application (trilog.exe) [5] |
| T1027.005 | Stealth | Obfuscated Files or Information: Indicator Removal from Tools | Analysis of these cryptcat binaries indicates that the actor continually modified them to decrease AV detection rates. One of these files was deployed in a TEMP.Veles target's network [34] |
| T1056.003 | Credential Access/Collection | Input Capture: Web Portal Capture | The updated phone numbers redirected to websites controlled by the hackers, enabling them to capture and use any login codes sent to the devices via text message [33] |
| T1003.001 | Credential Access | OS Credential Dumping: LSASS Memory | Traditional malware backdoors, Mimikatz distillates, remote desktop sessions, and other well-documented, easily-detected attack methods were used throughout these intrusions [34] |
| T1573 | Command and Control | Encrypted Channel | Analysis of these cryptcat binaries indicates that the actor continually modified them to decrease AV detection rates. One of these files was deployed in a TEMP.Veles target's network [34] |

Table 4.1: ATT&CK Matrix for Enterprise Mapping of TEMP.Veles Activity

| ID | ATT&CK Tactic | ATT&CK Technique | Reference |
|---|---|---|---|
| T0886 | Initial Access/Lateral Movement | Remote Services | In the Triton Safety Instrumented System Attack, TEMP.Veles utilized remote desktop protocol (RDP) jump boxes, along with other traditional malware backdoors, to move into the ICS environment [37] |
| T0807 | Execution | Command-Line Interface | Trilog.exe took one option from the command line, which was a single IP address of the target Triconex device [5] |
| T0853 | Execution | Scripting | TEMP.Veles' lateral movement activities used a publicly-available PowerShell-based tool, WMImplant [34] |
| T0859 | Persistence/Lateral Movement | Valid Accounts | Traditional malware backdoors, Mimikatz distillates, remote desktop sessions, and other well-documented, easily-detected attack methods were used throughout these intrusions [34] |
| T0872 | Evasion | Indicator Removal on Host | In the Triton Safety Instrumented System Attack, TEMP.Veles would programmatically return the controller to a normal running state if the Triton malware failed. If the controller could not recover in a defined time window, TEMP.Veles programmatically overwrote their malicious program with invalid data [37] |
| T0867 | Lateral Movement | Lateral Tool Transfer | Made attempts on multiple victim machines to transfer and execute the WMImplant tool [34] |
| T0843 | Lateral Movement | Program Download | Downloaded control logic on SIS controllers through the malware [37] |
| T0830 | Collection | Adversary-in-the-Middle | The updated phone numbers redirected to websites controlled by the hackers, enabling them to capture and use any login codes sent to the devices via text message [33] |
| T0855 | Impair Access Control | Unauthorized Command Message | Used TRITON to send unauthorized command messages to the SIS controllers [37] |
| T0828 | Impact | Loss of Productivity and Revenue | Tripped a SIS controller into a failed state, which caused the plant to shutdown, and paused operation for a week [5] |

Table 4.2: ATT&CK Matrix for ICS Mapping of TEMP.Veles Activity

### 4.3.1 SANS ICS Kill Chain: Stage 1

In the Reconnaissance step, the threat actors continuously scanned potential targets using the IP 87.245.143.140. This behavior maps to **T1595**.

In the Preparation phase, specifically the Weaponization step, the threat actors obtained the tools WMImplant and Mimikatz to be used in the target environment for Lateral Movement activities. Additionally, they modified cryptcat binaries to evade Anti-Virus (AV) software detection. This behavior maps to **T1588.002** and **T1036.005**. While the threat actor chose their specific entrance into the environment after targeting, the specific behavior is not documented publicly.

During the cyber intrusion phase, the attacker gained access to the target environment. The specific way the attacker achieved this is not publicly disclosed, nor their exploitation steps. The only information on this phase is that the attackers used scheduled tasks as a means of gaining persistence in the target environment. During investigations, the researchers found a ZIP folder containing files with scheduled task definitions [34]. This behavior maps to **T1053.005**.

After the attacker gained access, they established C2. In the Management & Enablement phase, it is believed the attacker established their C2 operations through cryptcat binaries. The investigators found cryptcat binaries in the target environment. This behavior maps to **T1573**.

Once established inside the target network, the attackers used many TTPs to finally reach the SIS engineering workstation. They used the tool WMImplant to move laterally inside the environment. Additionally, they used the tool Mimikatz to dump credentials from the LSASS Memory. The attacker steps are not exactly known, but the credentials might have then been used for more lateral movement using Remote Desktop Protocol sessions, and then also transfer WMImplant to other endpoints inside the network. This is all that is known of the attacker's activity inside the network to then reach the SIS engineering workstation, which starts Stage 2. This behavior is translated into the following TTPs: **T1059.001**, **T0853**, **T1003.001**, **T0859**, **T0886**, **T0867**.

### 4.3.2 SANS ICS Kill Chain: Stage 2

Stage 2 started when the attacker gained remote access to the SIS engineering workstation. This allowed the attacker to start gaining information on the target SIS in the environment. In the Attack Development & Tuning phase the attacker developed the TRITON malware for communicating and hijacking the SIS. As discussed before this step of the Kill Chain does not necessarily happen inside the target environment. The malware being advanced and proprietary to the Triconex model of SIS, indicates that the attackers had access to a similar model outside the target environment. Additionally, the attackers named the malware executable *trilog.exe* to masquerade it. This behavior maps to **T1587.001** and **T1036.005**.

Before deploying the malware inside the target environment, the attackers most probably tested on a similar or identical SIS acquired. There is no specific evidence of these steps undertaken, thus there is no documented behavior to map.

After preparing the malware, the attacker undertakes the ICS Attack phase. The attacker delivered the malware to the SIS engineering workstation, enabling them to communicate with the SIS. In the Install/Modify step, the attackers downloaded control logic to the SIS, with the malware only requiring the IP address of the SIS as a parameter. Additionally, the malware would return the controller to a normal running state if an error occurred. This behavior maps to **T0843** and **T0872**.

While the attacker's intent might not have been to inadvertently trip the SIS while delivering control logic to them, they still caused damage to the ICS. The attackers sent

unauthorized command messages to the SIS, with the intent of modifying its functionality, which is an enabler to a potentially larger attack. By delivering control logic, the attackers tripped a SIS, shutting down the petrochemical plant for a week which caused loss of productivity and revenue. This behavior maps to **T0828** and **T0855**.

| **Stage** | **Phase** | **Step** | **ID** |
|---|---|---|---|
| 1 | Planning | Reconnaissance | T1595 |
| | Preparation | Weaponization | T1587.001 |
| | | | T1027.005 |
| | | Targeting | N/A |
| | Cyber Intrusion | Delivery | N/A |
| | | Exploit | N/A |
| | | Install/Modify | T1053.005 |
| | Management & Enablement | C2 | T1573 |
| | Sustainment, Entrenchment, Development & Execution | Act | T1059.001 |
| | | | T0853 |
| | | | T1003.001 |
| | | | T0859 |
| | | | T0886 |
| | | | T0867 |
| 2 | Attack Development & Tuning | Develop | T1587.001 |
| | | | T1036.005 |
| | Validation | Test | N/A |
| | ICS Attack | Deliver | N/A |
| | | Install/Modify | T0807 |
| | | | T0843 |
| | | | T0872 |
| | | Execute ICS Attack | T0828 |
| | | | T0855 |

Table 4.3: Mapping of the TEMP.Veles behavior against the SANS ICS Kill Chain

# Chapter 5
# Scenario & Range Design

This chapter applies the design stages of the methodology to the TRITON case, translating the preceding analysis into a cyber range. The following sections will describe the architecturally relevant TTPs, and in conjunction with the context, provide the range architecture. Lastly, it will discuss the selected TTPs for the intended scenario of the range.

## 5.1 Architecturally Relevant TTPs

As described in 3.3, a TTP is architecturally relevant if it reveals or constrains a specific property of the environment. In the case of TRITON several TTPs satisfy this criterion. The TTPs **T1059.001** along with **T1003.001** imply the presence of at least one Windows OS host in the environment. Both TTPs describe behavior using technologies that are characteristic of the Windows OS environment, namely PowerShell and LSASS. **T0886** indicates that remote services were used for lateral movement. On its own this does not specify which service, however, combined with the established presence of Windows OS, it can reasonably be inferred that Remote Desktop Services (RDS) were used, as RDS is the standard remote-access service in Windows environments. This compositional inference is also corroborated by direct evidence from the CTI that RDP was used by the threat actors.

As for the OT environment, the TTPs **T0855** and **T0807** together indicate that CLI-based communication with the SIS is required.

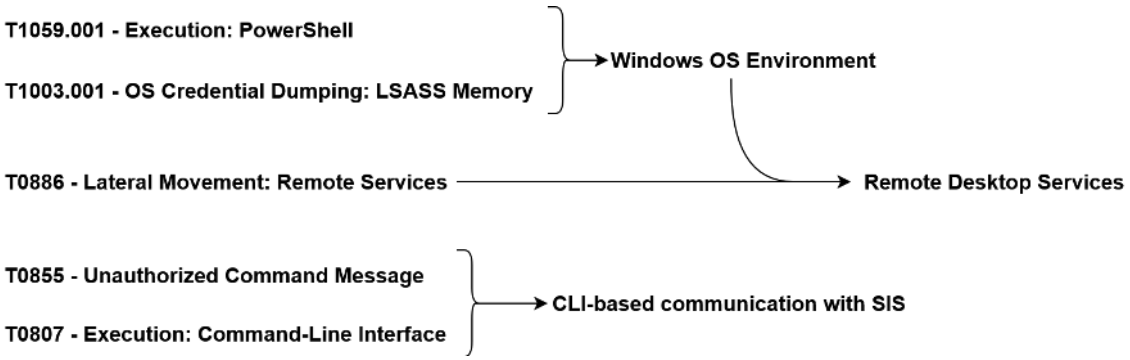


Figure 5.1: Illustration summarizing TTPs that give direction

## 5.2 Campaign context

Campaign context is crucial for the range to be as realistic as possible. The context that is important in this case study is the attack path followed by the threat actor. TEMP.Veles breached the corporate environment and gained access to a corporate computer, then moved laterally into the OT environment and eventually reached the SIS engineering workstation to ultimately hijack the SIS. From this context, it can be determined that the range needs a SIS, a SIS engineering workstation, an IT-to-OT escalation, and a corporate computer. Additionally, we know that they used RDP jump boxes for escalation, thus it can

| Range Input | Inferrence Basis |
|---|---|
| Windows OS | TTP-inferred |
| RDS | TTP-inferred |
| SIS CLI-based communication | TTP-inferred |
| SIS presence | Context-inferred |
| SIS engineering workstation | Context-inferred |
| IT-to-OT escalation through RDP jump box | Context-inferred |
| IT Workstation | Context-inferred |
| Domain Controller | Context-inferred |

Table 5.1: The range inputs inferred from cyber threat intelligence, showing each required component or property and whether it was inferred from a specific TTP (TTP-inferred) or from the campaign context (context-inferred)

be inferred that there is an RDP jump box that allows the IT-to-OT escalation through RDS as inferred in the previous section. Lastly, the threat actors used credential-based lateral movement, which combined with the inference of a Windows environment, reasonably implies domain-based authentication through a Domain Controller.

## 5.3 Realism versus Feasibility trade-off: TRITON Case

Not every element of the target environment can be reproduced with full fidelity, especially in the OT environment. This also applies to the TRITON case reproduction. The most significant trade-off concerns the Triconex SIS controllers. This model of SIS is closed-source and cannot be virtualized. Additionally, acquiring one costs several thousand dollars, making it infeasible even for a hybrid implementation. For this reason, the SIS is reproduced through a custom simulator, developed as part of this work (detailed in Chapter 6.4). The simulator reproduces the behavior necessary for the environment: a communication path from the engineering workstation via CLI commands, and continuously monitoring the OT process to ensure that it remains in a safe state. Although the simulator is not an exact copy of the Triconex SIS, it provides the necessary functionality and a target for the attack path.

Furthermore, the exact components of the petrochemical facility are not publicly known, owing to the sensitive nature of the environment. Since an OT process is nonetheless essential to the cyber range, a comparable chemical process is implemented using GRFICSv2 [28]. This again favors feasibility over full realism. The substitution is acceptable because the attack targets the safety instrumented system rather than the specific process it protects; the exact process is therefore not essential to reproducing the attacker's behavior, and the fidelity lost is confined to the environment rather than the attack path. As a result, the attack paths are preserved, and the intended scenario can still be executed as planned.

## 5.4 Range Architecture

The architecture is derived from the TTPs detailed in 5.1, the campaign context, and reference knowledge of typical IT/OT environments based from the book *Industrial Cybersecurity* [1]. Figure 5.2 presents the architecture of the range that has been designed

for the TRITON malware threat actor reproduction, structured according to the Purdue Model.

In the Enterprise network, there is a Windows OS computer (ENT-WS1), and a Domain Controller (ENT-DC). As it has been inferred in 5.1, the environment comprises Windows OS endpoints, and also since it is an enterprise network there is a Domain Controller to manage the Active Directory Service of the environment. There is an additional Windows OS computer (ENT-WS2) in Level 4 of the network, which will be the first target of the intended scenario, also it is placed in Level 4 because it is from there that a user can connect to the OT environment, thus having a role fitting to the level..

**Information**

Active Directory (AD) is Microsoft's directory service for Windows environments, used to centrally manage users, computers, and other resources within a network [38]. It provides authentication and authorization across the domain, allowing organizations to define identities and enforce access policies from a single point rather than configuring each machine. The Domain Controller (DC) is the server that hosts Active Directory and carries out these functions: it stores the directory database, validates login requests, and applies security policies for the domain [39]. Because it authenticates users and governs access across the environment, the Domain Controller is a high-value target for attackers and compromising it can yield control over the entire domain, including the ability to move laterally and access other systems using legitimate credentials.

In the Industrial Demilitarized Zone (IDMZ), also considered level 3.5 of the Purdue Model, there is an RDP Jump Box. It acts as an intermediary between two network zones, so that there is no direct connectivity. In this case, it prevents the Enterprise Zone from being directly connected to the Industrial Zone. For this specific range, it will allow the escalation from the IT to the OT environment.

Lastly, the Industrial Zone is made up of seven components. In the Site Operations Level, there is a Windows-based computer (PLC-ENG-WS) which has the purpose of controlling the PLC, and is infered from the context of an OT environment. Additionally, there is also a Domain Controller (IND-DC), which implements a separate industrial domain and Active Directory that is separate from the Enterprise domain. In the Area Supervisory Control Level there is a HMI which monitors the process, and also a Windows-based computer (SIS-ENG-WS) which allows the configuration of the SIS. The presence of the HMI comes from the context of an OT environment, and the SIS engineering workstation from previous inference. The Basic Control Level consists of a PLC and the SIS. There is no OT process without a PLC and the SIS is the main target of the range. Lastly, the Process Level consists of the chemical process simulation.

The range is intentionally kept minimal, reproducing only the segmentation essential to the attack path, notably the separation between the Enterprise and Industrial zones, enforced by firewalls on either side of the IDMZ, without modeling the broader network complexity of a full production environment. Because these firewalls block direct connectivity between the two zones, the RDP jump box in the IDMZ provides the only sanctioned path between them, which is the path the attacker traverses during escalation. This preserves the reproducibility of the attacker's escalation while keeping the range feasible to build and operate.

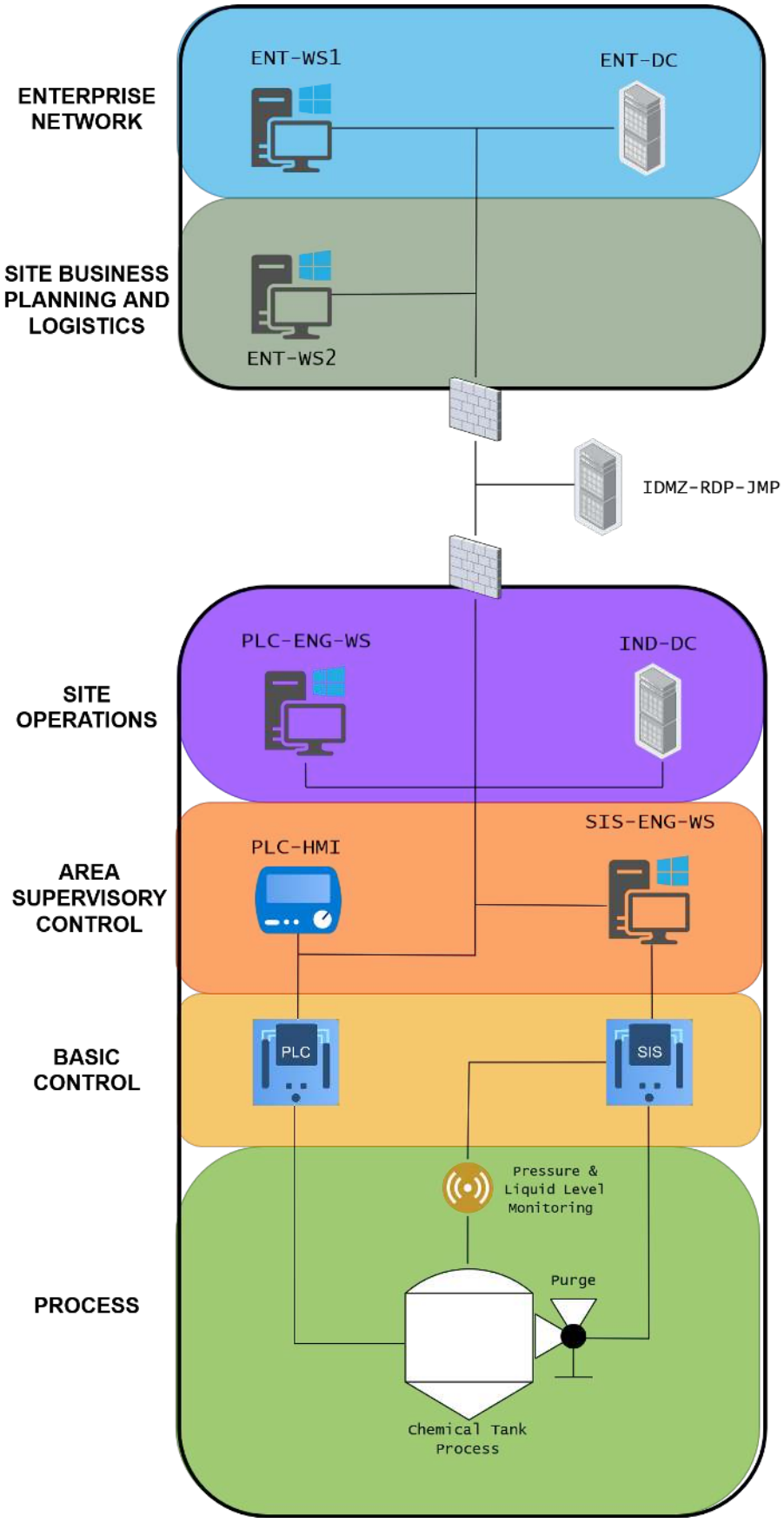


Figure 5.2: The resulting cyber range architecture from the TRITON attack CTI

| Component | Purdue Level | Basis | Reproduction approach |
|---|---|---|---|
| ENT-WS1 (Windows OS) | 5 | Table 5.1 | Virtualised |
| ENT-WS2 (Windows OS) | 4 | Table 5.1 | Virtualized |
| ENT-DC | 5 | IT Enterprise network context | Virtualized |
| IDMZ-RDP-JMP | 3.5 | Table 5.1 | Virtualized |
| PLC-ENG-WS | 3 | Typical OT environment [1] | Virtualized |
| IND-DC | 3 | Typical OT environment [1] | Virtualized |
| PLC-HMI | 2 | Typical OT environment [1] | Virtualized |
| SIS-ENG-WS | 2 | Table 5.1 | Virtualized |
| PLC | 1 | Typical OT environment [1] | Simulated/Virtualized |
| SIS | 1 | Table 5.1 | Simulated/Virtualized |
| Process | 0 | Typical OT environment [1] | Simulated/Virtualized |

Table 5.2: Components of the architecture with their basis and reproduction approach

## 5.5 Attack Scenario Design

With the architecture defined, the intended attack scenario specifies the sequence of actions and the TTPs the user is intended to perform. It mimics the SANS ICS Kill Chain as a series of manually executed steps, although not using all the TTPs that are in it. The reason being that not all TTPs are directly relevant to the attack path, for example **T1587.001 - Develop Capabilities: Malware** which describes the attacker developing malware outside the target environment. The following scenario uses the TTPs to describe an ordered, executable attack path.

The scenario consists of an intended assumed breach scenario placing the attacker in the target environment with known credentials. The reasoning is that it is not known how the attacker gained initial access into the environment, thus to stay faithful to the known information, the user will have credentials to access the ENT-WS2 workstation.

After the attacker has accessed the workstation, they will need to discover information on accessing the OT environment through artifacts already in the workstation. The information enables the user to use the IDMZ-RDP-JMP to access the Industrial Network, specifically to the PLC-ENG-WS. The TTPs performed during this first lateral movement step are **T0886**, **T0859**, and **T0867** from Table 4.3. Noticeably, the intended attack path is not using the TTPs from Planning, Preparation, and Cyber Intrusion steps of the mapped SANS ICS Kill Chain. The reason being that **T1595**, **T1587.001**, **T1027.005** describe activity behavior that is peripheral to the intended scenario. With the assumed breach, the user will not need to do active infrastructure scanning, and the user will have access to a Kali Linux machine with Mimikatz available. Lastly, the simulator does not require any malware since it will provide a communication gateway on the SIS engineering workstation.

On the PLC-ENG-WS the user will have to do LSASS memory credential dumping to acquire new credentials that will be used to move to the SIS-ENG-WS. The user will need to transfer the tool Mimikatz to the PLC-ENG-WS and then perform the dumping. This behavior maps to **T1003.001**, **T1059.001**, **T0867**, and **T0853**. The TTPs from the Develop step of Stage 2 of the ICS Kill Chain are purposefully excluded since they

concern behavior outside of the environment.

With the newly acquired credentials, the user can access the SIS-ENG-WS, where they will find documentation on the workings of the simulator and how it functions, which allows them to achieve the goal of manipulating the SIS. This interaction with the SIS maps to TTPs **T0807** and **T0855**. The SIS hijack is the main goal of the scenario, but as a further confirmation of the hijack, the user can move back laterally to the PLC-ENG-WS and upload a malicious configuration to the PLC which would cause physical damage to the process, the TTP being **T0828**. Further details on exact manipulations needed will be explained in the following chapter.

The intended scenario describes one attack path and the behavior the user must perform to reach the objective. Not all TTPs from the SANS ICS Kill Chain mapping appear in this path. However, since the goal of the methodology is to provide an environment where the attacker's behavior can be reproduced, the range supports the reproduction of these additional TTPs as well. Beyond the guided path, a user may establish persistence through scheduled tasks (**T1053.005**), establish C2 communications (**T1573**), obtain additional tools not present on the provided Kali Linux machine (**T1588.002**), or manipulate the SIS to return it to a safe state (**T0872**). These capabilities follow from the components of the environment, allowing the range to serve both as a guided scenario and as an open environment for reproducing the broader campaign behavior, supporting its use for training, testing, and research.

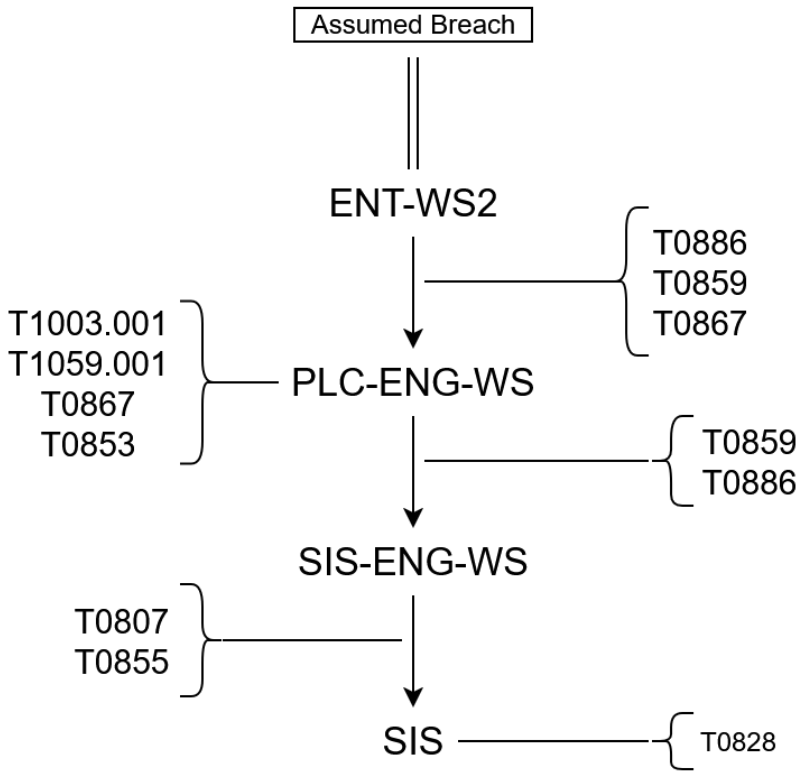


Figure 5.3: Intended attack path starting with provided credentials upon assumed breach, alongside the TTPs performed through each step and on endpoints

# Chapter 6

# Cyber Range Implementation

This chapter describes the implementation of the cyber range designed in Chapter 5. The following sections detail the deployment process, the Enterprise and Industrial Zone components, the Safety Instrumented System (SIS) simulator, and the manner in which the range enables the intended attack path and associated tactics, techniques, and procedures (TTPs). The implementation builds on existing tooling for the underlying infrastructure; the primary contribution of this work lies in the realization of the range itself and the development of a custom SIS simulator.

The range was deployed using Ludus, a tool for building and managing virtualized environments for cyber security testing and training, which operates on top of Proxmox [40, 41]. Ludus enables the declarative definition of multi-machine environments through YAML configuration files. Based on this configuration, Ludus automatically provisions the virtual machines, configures their operating systems, establishes network segmentation, and applies the necessary Ansible roles, which allow you to automatically load related variables, files, tasks, handlers, and other Ansible artifacts [42]. Ludus was selected for its ability to define and reproducibly deploy multi-VM environments, a capability that directly addresses the reproducibility and scalability requirements of this range.

## 6.1 Environment Deployment

The environment was provisioned as a set of virtual machines organized according to the Purdue model zones defined in the architecture presented in Chapter 5. To provision the virtual machines, Ludus used templates as the basis for each VM; these templates were built directly from ISO files, allowing for clean customization of the specific environment. Each zone from the architecture was configured as a separate network segment, with each zone assigned a distinct Virtual Local Area Network (VLAN): the Enterprise zone used VLAN 20, the IDMZ used VLAN 30, and the Industrial zone used VLAN 40, matching the network segmentation prescribed by the Purdue model, in which enterprise, DMZ, and industrial traffic are strictly isolated. Ludus additionally provisioned a router VM, through which firewall rules were implemented to enforce the boundaries defined in the architecture. Specifically, direct traffic between the Enterprise and Industrial zones was blocked, forcing all inter-zone traversal through the IDMZ jump box. Only RDP traffic on port 3389 was permitted from the Enterprise zone to the IDMZ jump box, with all further Industrial-zone access requiring an authenticated session originating from that host.

Ludus uses Ansible roles to automate content deployment, Active Directory (AD) configuration, task execution, and variable assignment. In this range, Ansible roles were used to configure the AD environment, transfer and install files and software, set up and run services, and enforce login policies across the virtual machines. The deployment relied on roles provided by the Ludus development team via Ansible Galaxy, as well as custom roles developed specifically for this range.

## 6.2 Enterprise zone

The Enterprise zone consists of four virtual machines: two workstations, a domain controller (DC), and an attacker machine representing the adversary's foothold for the assumed-breach scenario. The workstations were provisioned using a Windows 11 (version 22H2) template, the DC using Windows Server 2022, and the attacker machine using Kali Linux.

The Active Directory content for the Enterprise domain (ent.triton.local) was configured using the ludus_bulk_ad_content role, provided by Bad Sector Labs, in custom mode to allow explicit definition of the directory structure, groups, and users. A single organizational unit, Staff, was created under the domain root, containing one global security group named Standard Users. Three user accounts were then created within the Staff OU. The account *ent_poseidon* is a member of the Domain Admins group and represents a privileged account in the domain. The accounts *bob* and *alice* are standard users belonging to the Standard Users group. All three accounts were configured with non-expiring passwords to keep the environment stable across the lifetime of the range.

As *bob* is the victim of a phishing attack in this scenario, their credentials are provided to the attacker on the Kali machine for the assumed-breach scenario. To support the escalation towards the Industrial Zone, **ENT-WS2** includes a simulated onboarding document detailing the connection process through the IDMZ-RDP-JMP box to the Industrial Zone. This supports the recreation of the TTPs **T0886**, and **T0859**. The Enterprise Zone also has ENT-WS1, which is present for realism purposes, providing two avenues of exploration for the user.

## 6.3 Industrial zone

The Industrial Zone consists of seven VMs, specifically the Industrial Domain Controller, the PLC engineering workstation, the SIS engineering workstation, an HMI, a PLC, a SIS, and the chemical process simulation. The DC uses a Windows Server 2022, while the workstations use Windows 10 (Version 22H2), since OT environments tend to have older systems in place to avoid continous updates and risk availability. The OT components all run on headless Debian 12 VMs.

The Industrial zone is governed by a separate Active Directory domain (ind.triton.local), hosted on its own primary domain controller. This separation reflects the administrative and identity isolation commonly maintained between IT and OT environments. As with the Enterprise domain, the content was configured using the ludus_bulk_ad_content role in custom mode. A single organizational unit, Operators, was created, containing two global groups, Service Accounts and Staff. Four accounts were defined: *ind_poseidon*, a Domain Admins member representing the privileged account in the industrial domain, *PLC_Operator* and *SIS_Operator*, operator accounts belonging to the Staff group, and *svc_engineering*, a service account. The *svc_engineering* account is configured with a Service Principal Name (SPN) and a weak password, and belongs to the Service Accounts group. All accounts have non-expiring passwords to maintain a stable environment.

To set up and install the necessary software according to the GRFICSv2 project, custom Ansible roles were used to automate the process. The range uses a combination of software from GRFICSv2, and the custom created SIS addition (6.4) that consists of the simulator (sis simulator) and the client (sew_client.py). The GRFICSv2 project uses software that is open-source, but their proprietary configurations were implemented in the range. The

table below details the software placed in VMs of the Industrial Zone, and the Ansible roles used for each host.

| VM | Software | Ansible role |
|---|---|---|
| PLC-ENG-WS | OpenPLC Editor | plc_eng_ws_prep |
| PLC-HMI | ScadaBR | scadabr_vm_prep |
| SIS-ENG-WS | sew_client.py | sis_eng_ws_prep |
| SIS | sis simulator | sis_vm_prep |
| PLC | OpenPLCv2 Runtime | plc_vm_prep |
| PROCESS-SIM | Chemical Process Simulator | plant_simulation_role |

Table 6.1: Software deployed on each virtual machine in the Industrial Zone, together with the Ansible role used to install and configure it. The OpenPLC, ScadaBR, and chemical process simulator components are based on the GRFICSv2 project, while the SIS simulator and its client (sew_client.py) were developed as part of this work. Details on the roles are provided in Appendix C


Figure 6.1: GRFICSv2 visual simulation of chemical process

## 6.4 SIS Simulator

In the stock configuration of the GRFICSv2 project, the only protection against a hazardous over-pressure condition is control logic embedded within the PLC itself. Thus, an attacker that gains access to the PLC engineering workstation can drive the process into an unsafe state unopposed. To add an additional layer of security and create an environment that can reproduce the TRITON attack, an independent SIS was designed and implemented as an additional component for the GRFICSv2 project.

The SIS system is composed of 4 cooperating elements: An independent measurement path, a logic solver, a dedicated independent relief valve, and the engineering channel with

the workstation client. The independent measurement path has a dedicated sensor process that has been added to the simulation configuration of GRFICSv2, that exposes the reactor pressure and liquid level as its own Modbus/TCP device with a distinct network address. The core of the SIS is a control loop that, on a fixed scan cycle, reads the process variables, evaluates them against a set of configurable safety trips (high-pressure, high-level, and low-level thresholds), and when a limit is violated, commands the process to a safe state. The trip logic is **latching**, meaning that once a protective function activates it, it remains engaged until an operator resets it and the underlying condition has cleared. Additionally, the solver applies a fail-safe policy, if the feed becomes unreadable. To guarantee that the safe-state command cannot be overridden by the (potentially compromised) PLC, a dedicated relief/blowdown valve was introduced into the process physics itself. This actuator vents the reactor vapor space and is exposed as a Modbus device that only the SIS controls. Opening it on a trip is therefore an uncontested action; the shared control valves are additionally commanded as defense-in-depth but are not relied upon for the guarantee. The SIS exposes its safety configuration — the thresholds, enable flags, and bypasses — through a REST (HTTP) engineering interface. A companion command-line client, run from a Safety Engineering Workstation, downloads, inspects, and modifies this configuration. Crucially, configuration changes take effect on the next scan cycle, and every change is written to an audit log. This is the interface an engineer uses legitimately and, in the threat scenario, the interface an intruder abuses to raise or disable a trip threshold and so silently defeat the safety layer before manipulating the process.

Instead of replacing the conventions of GRFICSv2, the extension integrates seamlessly with the existing ones. Process values and actuator commands travel over Modbus/TCP, the same protocol GRFICSv2 already uses between the PLC and the simulation's remote-I/O bridges, therefore the SIS participates as an additional, independent Modbus master on the ICS network. The new sensor and relief-valve devices are implemented in the same remote-I/O style as the stock feed, purge, and tank devices, and are launched alongside them. The relief valve required a minimal, additive change to the simulation's process model, leaving normal operation unaffected until a trip occurs. The SIS host occupies its own address on the ICS subnet, and the engineering workstation reaches it across the testbed's existing topology. In effect, the contribution inserts a second, independent protection layer into GRFICSv2's control hierarchy, between the physical process and the operator. This enables the testbed to represent, and to be used to study, both safety-system defense and safety-system-targeted attacks within a single coherent environment, including the recreation of a TRITON-style attack.

## 6.5 Attack Path Enablement

This range configuration enables the intended attack path detailed in Chapter 5. The credentials of **bob** being given in the attacker machine support the assumed breach scenario start. Afterwards, on ENT-WS2 the attacker will find an onboarding document for the **bob** on how to access the Industrial Zone through the IDMZ-RDP-JMP onto the PLC-ENG-WS, thus enabling TTPs **T0886**, and **T0859**. Additionally, the user must transfer the Mimikatz tool onto the target machine to dump LSASS memory and extract credentials, corresponding to TTP T0867. This transfer can be accomplished by attaching a local drive over the RDP connection. To make the execution more feasible, and to remove the need for AV evasion techniques, Microsoft Defender has been disabled on both IDMZ-RDP-JMP

and PLC-ENG-WS.

After escalation into the OT environment, the user can use the tool Mimikatz to dump credentials of **svc_engineering** account. This corresponds to TTP **T1003.001**, **T1059.001**, **T0853**. Furthermore, the credentials of svc_engineering can be used to access SIS-ENG-WS, with the TTPs being **T0859** and **T0886**. While the attacker may find other credentials on PLC-ENG-WS, only **svc_engineering** credentials allow connection to the SIS-ENG-WS.

Stage 2 of the ICS Kill Chain starts when the attacker gains access to the SIS-ENG-WS, presenting the opportunity to control the SIS. On the SIS-ENG-WS, the attacker will find documentation on how to use the SIS client to be able to configure safety settings. The attacker can then change configurations or disable the SIS, so that malicious attacks can be executed. This behavior enables TTPs **T0807** and **T0855**. The disabling or the change of configurations of the SIS, leads to an impact corresponding to loss of productivity and revenue of the organization, thus TTP **T0828**.

# Chapter 7
# Scenario Execution & Discussion

This chapter demonstrates the execution of the intended scenario within the deployed range from Chapter 6, and discusses the results against the criteria of realism and feasibility.

## 7.1 Scenario Execution

The scenario starts from the assumed breach position established in the design, with bob's credentials present in a text file on the desktop of the Kali Linux machine.

```
# Credentials present on the attacker machine
bob
1_g0t_phi5hed:(
```

The credentials allow for an RDP connection to ENT-WS2. After gaining access to the workstation, an HTML file on the Desktop of bob, provides instructions on how to access the RDP jump box present in the IDMZ.

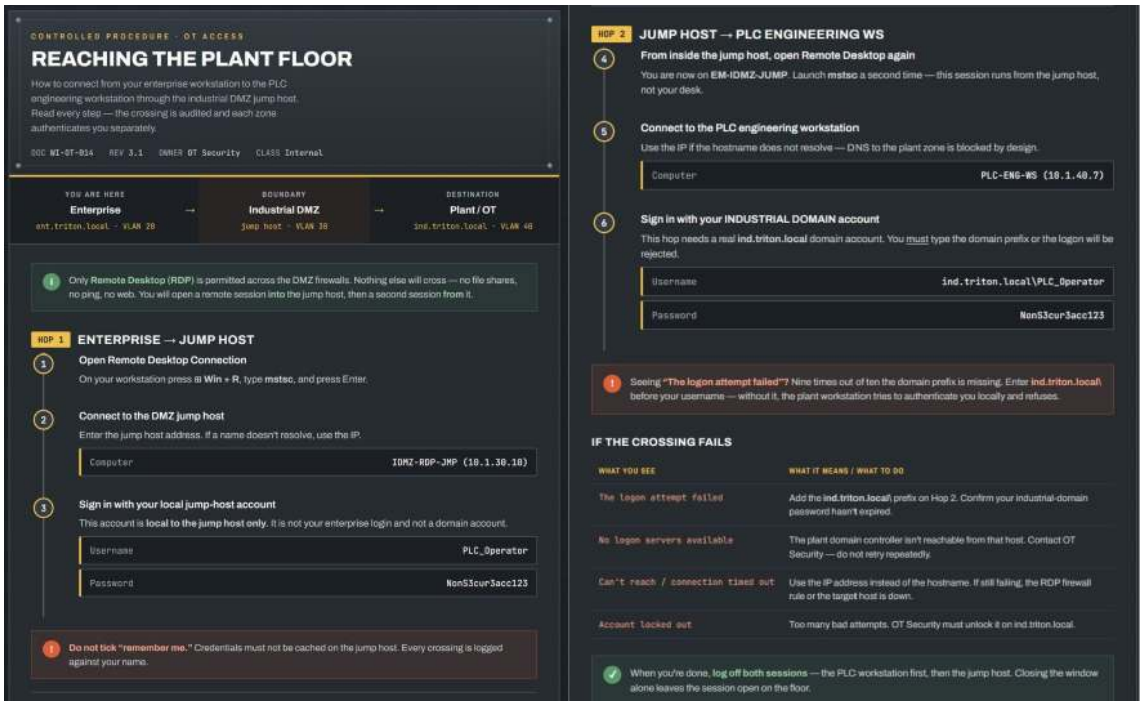


Figure 7.1: HTML file present on ENT-WS2 with instruction to access the Industrial Zone

Those new credentials are used to escalate to the Industrial Zone through the IDMZ jump box, while also transferring the tool Mimikatz. This hop towards the Industrial Zone is exactly the behaviour described by TTP **T0886**, **T0859**, and **T0867**, which is lateral movement using a valid account for remote services and transferring a tool. Using the credentials and the IPs in the HTML file, access is firstly gained on IDMZ-RDP-JMP and then from there access to PLC-ENG-WS. To be able to transfer Mimikatz to PLC-ENG-WS, the location of the tool is attached as a network shared drive to the RDP connection.

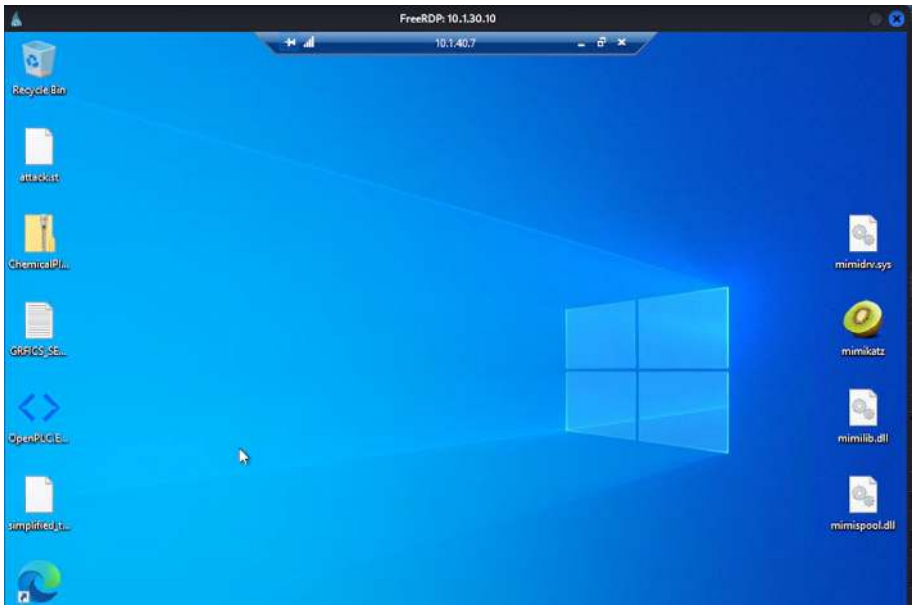


Figure 7.2: Screenshot of PLC-ENG-WS (10.1.40.7) showing Mimikatz transferred successfully

After copying the Mimikatz files on IDMZ-RDP-JMP, in a similar manner through the RDP software present in Windows, the user can attach the C:/ drive as a network share to PLC-ENG-WS, and in that way Mimikatz will successfully be transferred on the PLC-ENG-WS. PLC_Operator being a local administrator, Mimikatz can run successfully and dump credentials, with the output shown below:

```
User Name         : svc_engineering
Domain            : ind
Logon Server      : IND-DC
Logon Time        : 8/10/2026 12:44:42 AM
SID               : S-1-5-21-3431372852-1861592885-1049375497-1113
        msv:
         [00000003] Primary
         * Username : svc_engineering
         * Domain   : ind
         * NTLM     : 4868dfe18a10995edc9fa9d6cc7c447c
         * SHA1     : 31d5b9e8e4e7324ff78204c517cf6e891dcb45c5
         * DPAPI    : 595339621ffc32e2f583724aaef645b4
        tspkg :
        wdigest :
         * Username : svc_engineering
         * Domain   : ind
         * Password : (null)
        kerberos :
         * Username : svc_engineering
         * Domain   : IND.TRITON.LOCAL
         * Password : (null)
        ssp :
        credman :
```

```
cloudap :
```

The output shows the password hashes of the password of the svc_engineering account. The hash can be cracked offline in Kali with hashcat, or through an online website (depending on the complexity of the password). The hash effectively corresponds to the password *maintenance*.

These credentials are then used to establish an RDP connection from IDMZ-RDP-JMP to SIS-ENG-WS successfully. As described on the documentation present on the Desktop of the user on SIS-ENG-WS. the communication with the SIS is done through the client, as well as a batch file (.bat) describing the instruction and the flags the client takes. Specificaly, the batch file describes the command used to change the pressure point trip of the safety configuration of the SIS, which can be set for example at 4000 psi. This is an unsafe configuration since at 3100 psi, the chemical process is hazardous and explodes. To demonstrate the explosion, the user can upload, from PLC-ENG-WS, the file *attack.st* to the PLC web interface on *10.1.40.2*. That file is an unsafe configuration which drives the chemical process to an explosion.

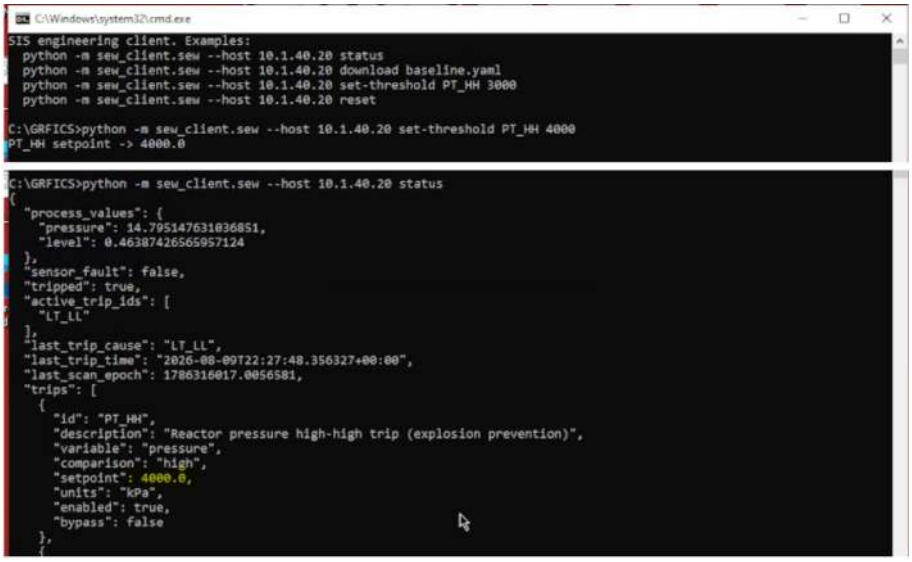


Figure 7.3: Screenshot showing manipulation command and SIS status with new pressure trip set at 4000 psi (over hazardous limit of 3100 psi)

## 7.2 Discussion

The construction of the range required trade-offs and choices to gain in feasibility. Notably, the trade-off is localized in the OT components rather than spread uniformly across the environment: the corporate hosts, domain infrastructure, and network segmentation could be reproduced with high fidelity through standard virtualisation, at little cost to feasibility, whereas the fidelity loss was concentrated almost entirely in the proprietary OT elements. The clearest instance is the Triconex SIS. As discussed in Chapter 5, a physical unit is expensive and closed source, so a custom simulator was developed to replace it. Although this required programming effort, an understanding of the GRFICSv2 simulation mechanism, and knowledge of how a SIS functions, the result was a realistic mechanism that integrated well with the process simulation: it continuously monitored the simulation and tripped when it detected an unsafe state, as defined by its safety configuration. The

feasibility of this step depends on the developer's skill, since building a custom simulator requires programming ability and domain understanding, though AI-assisted coding lowers this barrier. Importantly, even where a component could not be reproduced faithfully, the simulator reproduced the behaviour that mattered for the attack, the command interface and safety monitoring, if not the exact proprietary form.

In the particular case of this range, the integration of the GRFICSv2 components proved challenging due to the fact that the project uses old versions of Debian OS, an older version of the OpenPLC project, and the documentation was scarce at times, thus a lot of effort was required to integrate it in the Ludus environment. While this is purely a challenge resulting from choice, a range creator may use other ready-made solutions to adapt to the range which would be more feasible.

The range execution proved feasible, with an attack path that follows the ICS Kill Chain, and artifacts proving useful to guide the user to the end goal with no need to guess the next step. A user can clearly make use of the TTPs provided in the attack path, to follow and behave like the attacker, in this case being TEMP.Veles. Execution feasibility also depends on the user's expertise: experienced red teamers can complete the path readily, whereas less experienced users require more time to understand the environment and technologies involved.

The cyber range scenario closely reproduces the attack chain attributed to TEMP.Veles, encompassing the initial access on a corporate computer, escalation into the OT environment, lateral movement via RDS, and the ultimate goal of manipulating the SIS. As the exact steps and timing of the attacker's behavior are not publicly documented, the scenario is faithful specifically to the TTPs, tools, and sequence of actions reported in the available CTI, rather than to the incident in its entirety. Similarly, while the precise configuration of the targeted petrochemical facility remains unknown, the architecture reproduces a minimal environment sufficient to recreate this behavior: an ongoing chemical process and the SIS as TEMP.Veles' primary target. Overall, the range covers the phases of the attack and reproduces the TTPs of the intended attack path, while the underlying architecture supports the execution of the attacker's broader objective, the manipulation of the safety system governing the physical process.

Beyond the intended attack path, the user can expand their activity by using additional tools, applying stealth techniques, establishing persistence on hosts, or setting up C2 communications. The range provides an environment where a lot of other TTPs can also be performed for the purposes of training, defensive and academic research. While the range is built to adhere to the CTI of the TRITON attack, the environment is still built on IT and OT components that are found in many organizations, and with which other TTPs can be executed. For example, in this implementation, the service account svc_engineering for which we gained credentials through Mimikatz, the user can also perform Kerberoasting and perform lateral movement without the need of the password since the account also has a Service Principal Name (SPN).

Beyond its fidelity to the TRITON attack campaign, this work demonstrates the applicability of the underlying CTI-driven methodology more broadly. The process followed here, translating publicly available threat intelligence into a structured attack path, mapping and using ATT&Ck TTPs, and reproducing the minimal environment necessary to execute it, is not specific to the TRITON incident. The same approach could, in principle be applied to other publicly documented ICS-targeting campaigns (e.g., Stuxnet or Industroyer), provided sufficient CTI is available to reconstruct a plausible attack path.

This suggests that the methodology, rather than the range itself, constitutes the more transferable contribution of this work: individual ranges will always be constrained by the availability and completeness of public CTI, but the process by which CTI is converted into an executable scenario can be reused across different threat actors and attack chains.

The range and the evaluation present limitation that should be acknowledged. Firstly, the range reproduces and environment for the TRITON attack based on publicly available CTI and news, which is incomplete: the exact method of initial access, the precise sequence and timing of the attacker's actions, and the specific components of the targeted facility are not publicly documented. The scenario therefore reproduces a plausible reconstruction of the attack rather than an exact replica, and steps that could not be sourced were either inferred from context or deliberately scoped out, as in the assumed-breach starting point. Second, the OT environment relies on approximations: the Triconex safety controller is represented by a custom simulator rather than the physical device, and the industrial process is a substitute rather than the actual petrochemical process, since neither could feasibly be reproduced faithfully. As a result, aspects of the real systems' behavior are not represented, and the evaluation demonstrates that the attack path is executable and plausible, not that it is identical to the original incident. Finally, the range is intentionally minimal, reproducing only the segmentation and components necessary for the attack path; it does not capture the full complexity, monitoring, or defensive depth of a production environment. These limitations notwithstanding, the range demonstrates that available CTI can be translated into an executable, plausible reproduction of the attack, which was the aim of this work.

# Chapter 8
# Conclusion & Future works

This thesis set out to address the gap between the availability of CTI and its limited use in OT/ICS cyber range scenarios. As IT and OT domains become increasingly interconnected, the attack surface and the occurrence of attacks has grown substantially. Yet, cyber ranges that reproduce this interconnection for training, defensive testing, and research remain limited.

To this end, a methodology was developed for translating CTI into a deployable cyber range scenario. The methodology proposes a pipeline starting from gathering the CTI, reconstructing the ICS Kill Chain, scoping the TTPs, balancing realism and feasibility, and translating these into a range architecture alongside an intended scenario.

The approach was demonstrated by applying it to the TRITON attack campaign, reconstructing the attack, designing a corresponding range, and executing the intended scenario. The threat actor profile and ICS Kill Chain reconstruction were performed, providing a clear basis from which to scope the TTPs and derive the range architecture. An intended scenario was also developed, providing an attack path that reproduces the attacker's behavior. The range was implemented using Ludus, Proxmox, and Ansible roles. GRFICSv2 was integrated as the OT process simulation of the range, and a custom SIS was developed to serve as same target present in the TRITON attack campaign.

The primary contribution of this thesis is a methodology of translating CTI into a realistic and feasible cyber range scenario. In achieving this, the work also produced several secondary contributions, namely the custom SIS, attacker profile of TEMP.Veles, the SANS ICS Kill Chain reconstruction of the TRITON attack, and a deployable cyber range architecture where the attacker behavior can be recreated.

In answer to the research question, the work shows that available intelligence, combined with contextual knowledge and reference architectures, is sufficient to produce a scenario that is both faithful to the real attack and feasible to build and execute, subject to the realism–feasibility trade-offs inherent in reproducing proprietary OT systems. By bridging cyber threat intelligence and hands-on OT/ICS training, this work contributes a step towards more realistic and intelligence-driven defensive preparation.

## 8.1 Future Works

The objective of this section is to propose concrete improvements that could address current gaps and to outline future directions for research on the topic. The proposed directions fall into two categories: extending the methodology to other cases, and enhancing the range itself.

A first direction concerns the methodology. It could be applied to other documented ICS attack campaigns, such as the 2016 Industroyer attack or the 2015 Ukraine power grid attack, to assess its generality beyond the TRITON case and identify which aspects transfer across different attacks and OT domains. While reproducing these attacks would pose greater realism–feasibility challenges, given the more complex OT components and protocols of the energy sector, they would still provide useful avenues for extending the

methodology.

A second direction concerns the cyber range. The range currently focuses on reproducing the offensive attack path; it could be extended with defensive instrumentation such as logging, network monitoring, and a Security Information and Event Management (SIEM) system, enabling its use for blue-team training and detection research. This addition would allow for each TTP to be paired with its corresponding defensive signature, supporting the study of how the attack could be detected and mitigated at each stage.

A third direction concerns the architecture. The range currently uses a minimal architecture sufficient to achieve the intended objective. The current environment could be expanded toward a more complex, production-like environment to examine how the attack path scales and whether additional lateral-movement opportunities arise.

Taken together, these directions would extend the work from a single demonstrated scenario toward a broader and more versatile platform for OT/ICS security research and training. Whether by applying the methodology to further campaigns or enhancing the range with defensive instrumentation and greater architectural complexity, each direction builds on the foundation established in this thesis. As attacks on critical infrastructure continue to grow in sophistication, grounding training and research environments in real adversary behavior will remain an increasingly valuable pursuit, one to which this work contributes a concrete and extensible step."

# Bibliography


[1] P. Ackerman, *Industrial Cybersecurity, Second Edition.* Packt Publishing Pvt Ltd, 2021.

[2] M. J. Assante and R. M. Lee, "The industrial control system cyber kill chain," 2015.

[3] C. Peptan, "Considerations on some aggressions against critical infrastructure on the territory of ukraine during the "special military operation" conducted by the russian federation," *Annals of the "Constantin Brâncuşi" University of Targu Jiu, Engineering Series*, no. 1, pp. 37–48, 2022.

[4] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[5] B. Johnson, D. Caban, M. Krotofil, D. Scali, N. Brubaker, and C. Glyer, "Attackers deploy new ics attack framework "triton" and cause operational disruption to critical infrastructure," Dec. 2017.

[6] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the ukrainian power grid," Mar. 2016.

[7] M. A. I. Mallick and R. Nath, "Navigating the cybersecurity landscape: A comprehensive review of cyber-attacks, emerging trends, and recent developments," *World Scientific News*, vol. 190, no. 1, pp. 1–69, 2024.

[8] S. Ainslie, D. Thompson, S. Maynard, and A. Ahmad, "Cyber-threat intelligence for security decision-making: A review and research agenda for practice," *Computers Security*, vol. 132, p. 103352, 2023.

[9] R. Paes and others, "A guide to securing industrial control networks: Integrating IT and OT systems," *IEEE Industry Applications Magazine*, vol. 26, no. 2, pp. 47–53, 2020.

[10] K. Stouffer, M. Pease, C. Tang, T. Zimmerman, V. Pillitteri, S. Lightman, A. Hahn, S. Saravia, A. Sherule, and M. Thompson, "Guide to operational technology (ot) security," NIST Special Publication 800-82r3, National Institute of Standards and Technology, Gaithersburg, MD, 2023.

[11] R. Squillante, D. Jose dos Santos, F. Junqueira, and P. Eigi, "Development of control systems for safety instrumented systems," *IEEE Latin America Transactions*, vol. 9, no. 4, pp. 451–457, 2011.

[12] P. Huitsing, R. Chandia, M. Papa, and S. Shenoi, "Attack taxonomies for the modbus protocols," *International Journal of Critical Infrastructure Protection*, vol. 1, pp. 37–44, 2008.

[13] W. A. Conklin, "It vs. ot security: A time to consider a change in cia to include resilienc," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 2642–2647, 2016.

[14] T. Williams, "The purdue enterprise reference architecture," *IFAC Proceedings Volumes*, vol. 26, no. 2, Part 4, pp. 559–564, 1993. 12th Triennal Wold Congress of the International Federation of Automatic control. Volume 4 Applications II, Sydney, Australia, 18-23 July.

[15] MITRE ATT&CK, "Mitre att&ck," 2026.

[16] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre att&ck: Design and philosophy," 2020. Originally published July 2018, revised March 2020.

[17] O. Alexander, M. Belisle, and J. Steele, "Mitre att&ck for industrial control systems: Design and philosophy," 2020.

[18] Lockheed Martin, "Cyber kill chain®," 2026.

[19] M. N. Katsantonis, A. Manikas, I. Mavridis, and D. Gritzalis, "Cyber range design framework for cyber security education and training," *International Journal of Information Security*, 2023. Open access, CC BY 4.0.

[20] M. M. Yamin, B. Katt, and V. Gkioulos, "Cyber ranges and security testbeds: Scenarios, functions, tools and architecture," *Computers Security*, vol. 88, p. 101636, 2020.

[21] A. Garg, A. Boualouache, A. Imeri, and U. Roth, "A survey of cyber range training exercise scenario description, generation, and execution," 2025. Preprint, not peer reviewed.

[22] E. Russo, G. Costa, and A. Armando, "Building next generation cyber ranges with crack," *Computers Security*, vol. Volume 95, 04 2020.

[23] OpenAEV-Platform, "Openaev: Open adversarial exposure validation platform," 2026.

[24] A. Zacharis and C. Patsakis, "AiCEF: an AI-assisted cyber exercise content generation framework using named entity recognition," vol. 22, no. 5, pp. 1333–1354.

[25] S. Khan, A. Volpatto, G. Kalra, J. Esteban, T. Pescanoce, S. Caporusso, and M. Siegel, "Cyber range for industrial control systems (cr-ics) for simulating attack scenarios," 04 2021.

[26] A. P. Mathur and N. O. Tippenhauer, "Swat: a water treatment testbed for research and training on ics security," in *2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater)*, pp. 31–36, 2016.

[27] F. Sauer, M. Niedermaier, S. Kießling, and D. Merli, "Licster – a low-cost ics security testbed for education and research," in *Electronic Workshops in Computing*, BCS Learning & Development, 2019.

[28] Fortiphyd Logic, "Grfics: Open-source ot & ics security lab," 2026.

[29] The MITRE Corporation, "Caldera: Automated adversary emulation platform," 2026. Built on the MITRE ATT&CK framework.

[30] Autonomy-Logic, "Openplc editor: Ide capable of creating programs for the openplc runtime," 2026.

[31] A. Matrosov, E. Rodionov, D. Harley, and J. Malcho, "Stuxnet under the microscope," 2010.

[32] A. Cherepanov, "Win32/industroyer: A new threat for industrial control systems," 2017.

[33] B. Sobczak, "The inside story of the world's most dangerous malware," 2019.

[34] FireEye Intelligence, "Triton attribution: Russian government-owned lab most likely built custom intrusion tools for triton attackers," 2018.

[35] J. Slowik, "Zeroing in on xenotime: Analysis of the entities responsible for the triton event," in *Virus Bulletin Conference 2022*, (Prague, Czech Republic), Virus Bulletin, 2022.

[36] MITRE ATT&CK, "Triton safety instrumented system attack, campaign c0030," 2024. Version 1.1, last modified 12 May 2026.

[37] S. Miller and E. Reese, "A totally tubular treatise on triton and tristation," 2018.

[38] Microsoft, "Active directory domain services overview," 2025.

[39] Microsoft, "Active directory structure and storage technologies," 2014. Archived documentation; applies to Windows Server 2003–2012 R2.

[40] Bad Sector Labs, "Ludus: Cyber ranges for everyone," 2026.

[41] Proxmox Server Solutions GmbH, "Proxmox: Open-source server software solutions," 2026.

[42] Ansible Project Contributors, "Roles — ansible community documentation," 2026.

[43] Bad Sector Labs, "ludus_windows_utils: Windows & active directory utility roles for ludus cyber ranges," 2026.

[44] Cyblex Consulting, "ludus-local-users: Ludus role to create windows or linux local users," 2026.

[45] A. Oloruntola, "Ms-attack-range: Automated deployment tool for instrumented azure environments simulating attacks against microsoft sentinel," 2026. ludus_ms_sentinel/roles subdirectory.

# Appendix A

# Cyber Range Component Table

| **Asset Name** | **Zone** | **Role** | **OS** | **IP** | **Domain** |
|---|---|---|---|---|---|
| ENT-DC | Enterprise | Enterprise Domain Controller | Windows Server 2022 | 10.X.20.10 | ent.triton.local |
| ENT-WS1 | Enterprise | Workstation | Windows 11 (22H2) | 10.X.20.20 | ent.triton.local |
| ENT_WS2 | Enterprise | Workstation | Windows 11 (22H2) | 10.X.20.30 | ent.triton.local |
| ATTACKER-KALI | Enterprise | Attacker Machine | Kali Linux | 10.X.20.40 | N/A |
| IDMZ-RDP-JMP | IDMZ | RDP Jump Box | Windows Server 2022 | 10.X.30.10 | N/A |
| IND-DC | Industrial | Industrial Domain Controller | Windows Server 2022 | 10.X.40.6 | ind.triton.local |
| PLC-ENG-WS | Industrial | PLC Engineering Workstation | Windows 10 (22H2) | 10.X.40.7 | ind.triton.local |
| SIS-ENG-WS | Industrial | SIS Engineering Workstation | Windows 10 (22H2) | 10.X.40.8 | ind.triton.local |
| PROCESS-SIM | Industrial | Chemical Process Simulation | Debian 12 | 10.X.40.9, 10.X.40.10, 10.X.40.11, 10.X.40.12, 10.X.40.13, 10.X.40.14, 10.X.40.15, 10.X.40.16, 10.X.40.17 | N/A |
| PLC-SIM | Industrial | PLC | Debian 10 | 10.X.40.2 | N/A |
| SIS-SIM | Industrial | SIS | Debian 12 | 10.X.40.20 | N/A |
| PLC-HMI | Industrial | HMI | Debian 12 | 10.X.40.21 | N/A |

Table A.1: Table of components of the cyber range implementation including Asset Name, Zone, Role, OS, IP, and Domain

| **Username** | **Full Name** | **Domain** | **Password** | **Organizational Unit** | **Groups** |
|---|---|---|---|---|---|
| ent_poseidon | Poseidon Triton | Enterprise | `T3mp_V3l35` | OU=Staff,DC=ent,DC=triton,DC=local | Domain Admins |
| bob | Bob Smith | Enterprise | `1_g0t_phi5h3d:(` | OU=Staff,DC=ent,DC=triton,DC=local | Standard Users |
| alice | Alice Johnson | Enterprise | `1_d1dnt_g3t_phi5h3d:)` | OU=Staff,DC=ent,DC=triton,DC=local | Standard Users |
| ind_poseidon | Poseidon Triton | Industrial | `T3mp_V3l35` | OU=Operators,DC=ind,DC=triton,DC=local | Domain Admins, Staff |
| PLC_Operator | PLC Operator | Industrial | `NonS3cur3acc123` | OU=Operators,DC=ind,DC=triton,DC=local | Staff |
| SIS_Operator | SIS Operator | Industrial | `y0u_n33d_to_4irg4p_the_SIS` | OU=Operators,DC=ind,DC=triton,DC=local | Staff |
| svc_engineering | Service Engineering | Industrial | `maintenance` | CN=Users,DC=ind,DC=triton,DC=local | Service Accounts |

Table A.2: Active Directory user accounts configured across the Enterprise and Industrial domains

# Appendix B

# Intended Scenario Walkthrough

The following screen captures from the cyber range showcase the steps undertaken for the intended scenario.

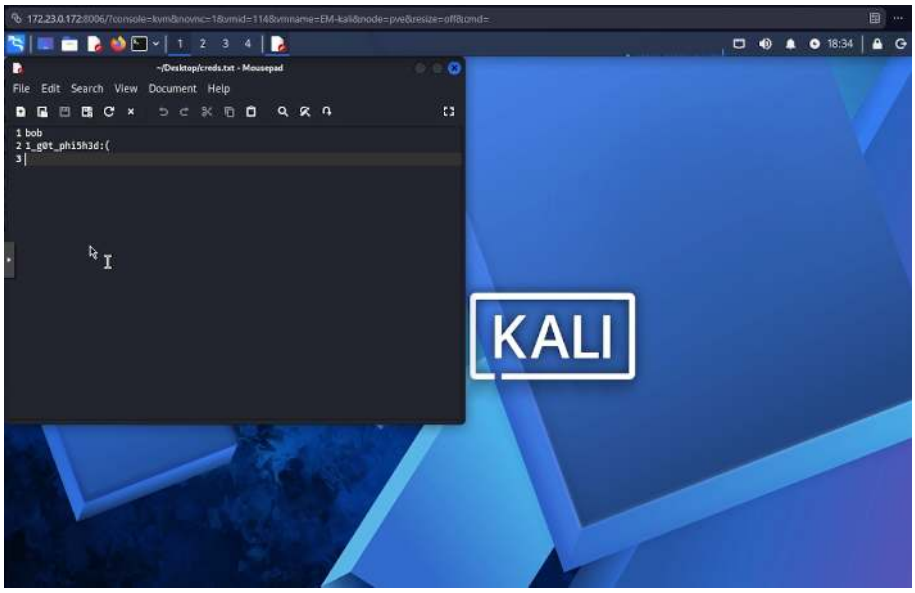


Figure B.1: Credentials present in ATTACKER-KALI for the assumed breach scenario

CLI command to establish an RDP connection with ENT-WS2

```
xfreerdp /u:bob /p:'1_g0t_phi5h3d:(' /v:10.1.20.30
```


Figure B.2: Desktop of ENT-WS2 through the RDP connection from ATTACKER-KALI

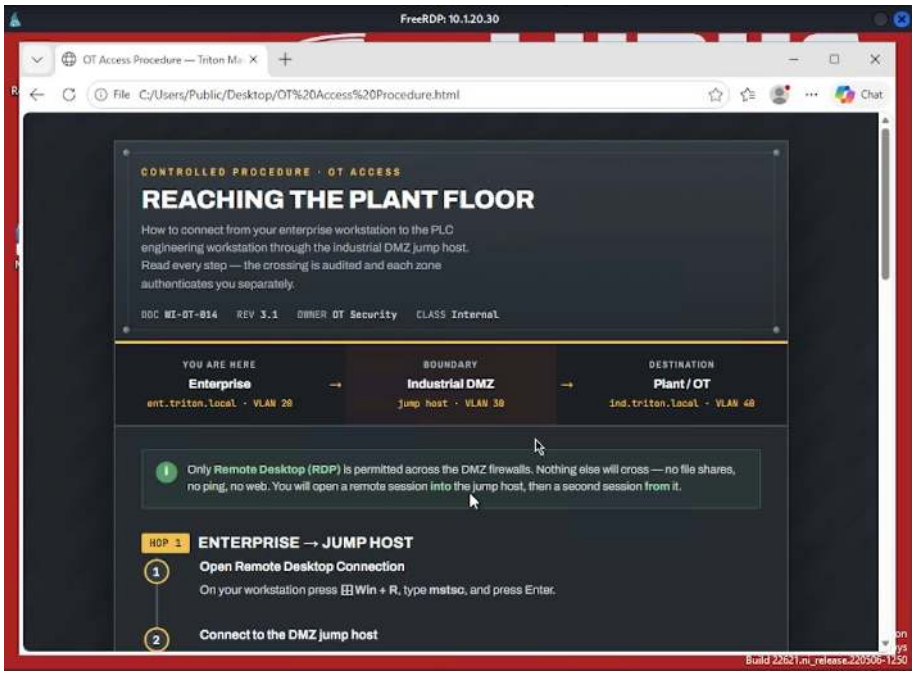


Figure B.3: On-boarding document located in ENT-WS2 detailing how to connect to the Industrial Zone environment through IDMZ-RDP-JMP

CLI command to establish an RDP connection with IDMZ-RDP-JMP with the location of Mimikatz attached as a network drive to transfer Mimikatz

```
xfreerdp /u:PLC_Operator /p:NonS3cur3acc123 /v:10.20.30.10  \
   /drive:/usr/share/windows-resources/mimikatz/x64
```

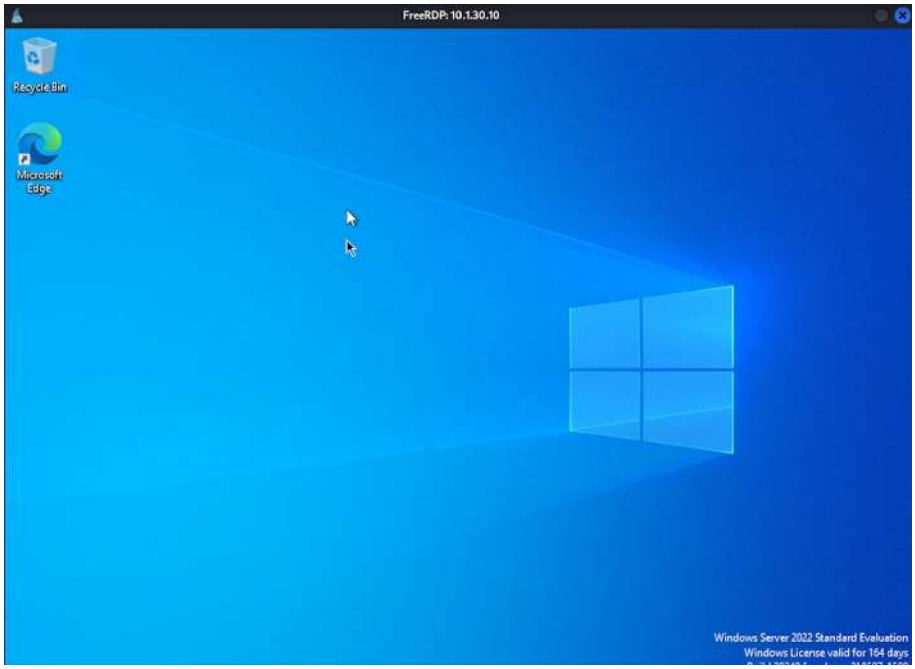


Figure B.4: Desktop of IDMZ-RDP-JMP through the RDP connection from ATTACKER-KALI


Figure B.5: RDP client on IDMZ-RDP-JMP with information to connect to PLC-ENG-WS


Figure B.6: Attaching the C: drive to the RDP connection from IDMZ-RDP-JMP to PLC-ENG-WS

Mimikatz commands to perform credential dump. Mimikatz needs to run with admin privileges.

```
privilege::debug
sekurlsa::logonPasswords full
```

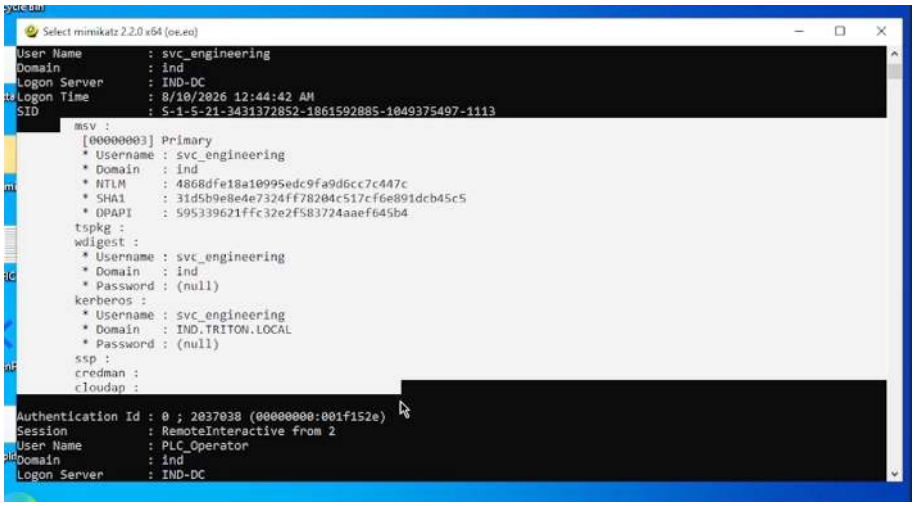


Figure B.7: Output of Mimikatz on PLC-ENG-WS. The highlighted portion shows the dumped hashes of the credential of svc_engineering.

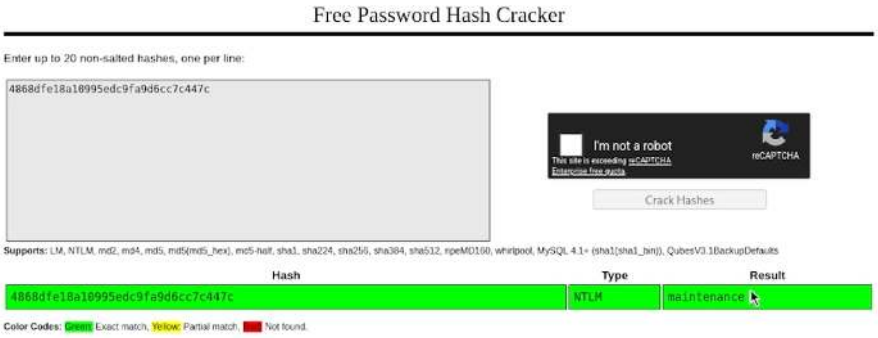


Figure B.8: Cracking the NTLM hash of the password of svc_engineering on crackstation.net. The password is *maintenance*

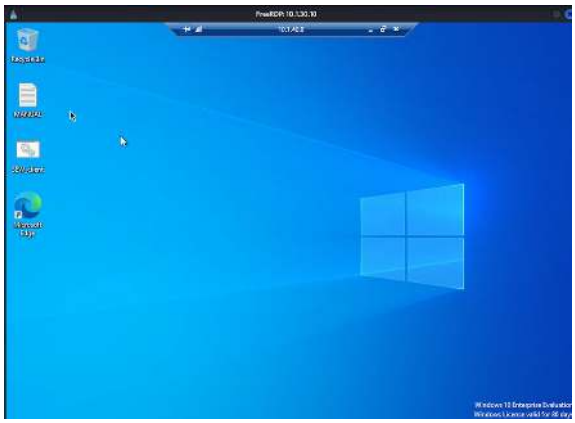

Figure B.9: Desktop of SIS-ENG-WS after RDP connection from IDMZ-RDP-JMP using the credentials of svc_engineering.

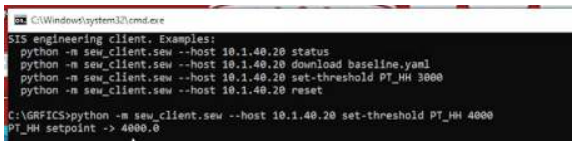


Figure B.10: Raising the pressure trip of the SIS to the unsafe level of 4000psi.

# Appendix C
# Ansible Roles

Ansible roles were used to automate content deployment across the cyber range, drawing on both open-source and independently developed roles. For the Active Directory content, the role *ludus_bulk_ad_content*, part of the *ludus_windows_utils* collection developed by Bad Sector Labs, was used [43]. To create the local account on IDMZ-RDP-JMP, the role *ludus-local-users*, developed by Cyblex Consulting, was used [44]. To disable Microsoft Defender on hosts, the role *disable_defender*, developed by Albert Timileyin, was used [45].

In addition to these third-party roles, eight custom roles were developed specifically for this range: *ent_ws_ot_access_doc*, *kali_tools*, *plant_simulation_role*, *plc_eng_ws_prep*, *plc_vm_prep*, *scadabr_vm_prep*, *sis_eng_ws_prep*, and *sis_vm_prep*. Each role's function is detailed in Table C.1.

| **Role** | **Description** |
|---|---|
| `ent_ws_ot_access_doc` | Drops an OT/DMZ access-procedure HTML document onto a Windows desktop (public/all-users desktop by default) for cyber-range scenarios. |
| `kali_tools` | Installs Kali Linux tools for use in Ludus cyber range scenarios |
| `plant_simulation_role` | Deploys and configures the chemical process simulation for the OT/ICS cyber range, based on the GRFICSv2 project. |
| `plc_eng_ws_prep` | Windows PLC engineering workstation — stages OpenPLC Editor, GRFICS control programs, and the IP-corrected HMI project |
| `plc_vm_prep` | Builds GRFICS OpenPLCv2 (vulnerable libmodbus) from source on Debian 10 |
| `scadabr_vm_prep` | Fresh ScadaBR install + GRFICS project with PLC IP corrected to the Ludus subnet |
| `sis_eng_ws_prep` | Deploys the GRFICS SIS engineering client (sew_client) on a Windows 10 SIS engineering workstation |
| `sis_vm_prep` | Deploys the GRFICSv2 Safety Instrumented System with IPs rewritten to the Ludus subnet |

Table C.1: Custom Ansible roles used to provision and configure the OT/ICS components of the cyber range