Enabling exercises, education and research with a comprehensive cyber range

Maria Leitner[1,2*], Maximilian Frank[2], Gregor Langner[2], Max Landauer[2], Florian Skopik[2], Paul Smith[2], Benjamin Akhras[2], Wolfgang Hotwagner[2], Stela Kucek[2], Timea Pahi[2], Lenhard Reuter[2], and Manuel Warum[2]
[1]University of Vienna, Faculty of Computer Science, Vienna, Austria
maria.leitner@univie.ac.at
[2]AIT Austrian Institute of Technology, Center for Digital Safety & Security, Vienna, Austria
firstname.lastname@ait.ac.at

Received: April 30, 2021; Accepted: October 8, 2021; Published: December 31, 2021

Abstract

With the continuous rise of threat actors and attacks, it is even more imminent to create cyber ranges, for example as training and exercise environment, that contribute to the modern challenges of a digital society. New digital environments are needed to tackle information security and cyber security challenges. In the past 15 years, virtual environments that simulate a diverse set of information technology (IT) and operational technology (OT) infrastructures, further called cyber ranges, have been developed. These virtual platforms provide the operations of exercises, training, research and other activities. The sum of these activities contribute to science and strengthen the practice of organizations. In this paper, we introduce the design and implementation of the AIT Cyber Range and outline its building blocks: computing platform, infrastructure provisioning, software provisioning and scenario engine. Furthermore, we describe five use cases: cyber exercises, training as well as security research and development on industrial control systems and intrusion detection systems. For future work, we aim to address federation and interoperability with other cyber ranges.

Keywords: Cyber Range, Cyber Security, Training, Cyber Security Exercises, Anomaly Detection

1 Introduction

Cyber-attacks and incidents have become an integral part of organizations' businesses. New business models such as *cybercrime as a service* have led to an increased number of attacks to organizations. With this constantly evolving threat and attack environment it has become highly relevant to create rooms where organizations can increase their preparedness or where simulations can be conducted that lead to a better understanding of threats and attacks. This development has evoked the emergence of cyber ranges. Cyber ranges are virtual environments that can simulate a diverse set of information technology (IT) and operational technology (OT) infrastructures for various purposes. The European Cyber Security Organisation (ECSO) [1] defines cyber ranges as "*a platform for the development, delivery and use of interactive simulation environments. [...]*." Literature often refers to cyber ranges as virtual environments using virtualization software but others may include also physical components (see, e.g., [2, 3, 4, 1]).

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 12(4):37-61, Dec. 2021
DOI: *10.22667/JOWUA.2021.12.31.037*

*Corresponding author: University of Vienna, Faculty of Computer Science, Waehringerstrasse 29, 1090 Vienna, Austria, Tel: +43-1-4277-79112 and AIT Austrian Institute of Technology, Center for Digital Safety & Security, SCT, Giefinggasse 4, 1210 Vienna, Austria

Cyber ranges have been actively used as learning and teaching environments in the past 15 years. For example, many cyber security exercises and training courses have been developed to increase capabilities, skills and competences of individuals and strengthen the resilience and preparedness of organizations against threats and attacks. Often, these cyber exercises are conducted on top of virtual environments. Cyber ranges support experiential learning by for example hosting training (e.g., [5, 6, 7]) or cyber exercises (e.g., [8, 9]). An advantage is that cyber ranges provide a safe environment for trainees. Production systems are normally unaffected by training or exercises. All actions are happening within the cyber range testbed.

This paper is an extended version of the conference proceeding in [10]. As in the original contribution, we updated and extended the architecture, implementation and use cases of the AIT Cyber Range. We describe the four building blocks: computing platform, infrastructure provisioning, software provisioning and scenario engine. We also demonstrate that we focus primarily on Open Source technology, as it allows for the rapid and modular deployment of infrastructure and software services. Specifically in this paper, we focus on an extended overview of five use cases. Hence, the contribution of this paper is an in-depth view on how we utilize the infrastructure to enable exercises, education and research. In particular, we elaborate five use cases:

- Cyber exercises: Cyber exercises can be designed as discussion-oriented or action-oriented exercises. We will discuss the general methodology that we apply and sketch an example exercise.
- Professional training: The cyber range can be utilized in professional trainings in the area of information security, awareness, OT security and many others. We describe several examples of professional trainings that we have conducted with information security experts and beginners.
- Higher education: We describe how cyber security education can benefit from utilizing a cyber range for teaching and learning digital skills. In this paper, we outline potential application scenarios for higher education.
- Simulation and Detection of Attacks in Industrial Control Systems: We demonstrate how a cyber range can provide a test environment for cyber-physical systems such as in a nuclear facility. We describe our approach and deployment with examples.
- Evaluation of Intrusion Detection Systems: We outline how the cyber range is utilized for setting up a model-driven testbed for a log data generation approach. We depict our approach and give an example.

Each use case outlines the versatility of the cyber range. Overall, the use cases contribute to building competencies and skills but also to investigate and contribute to new research directions (e.g., simulation and detection of attacks).

The content of the paper can be of high interest to several target groups:

- Cyber range providers may benefit to learn about the architecture and the deployment of other cyber ranges. Also they can be interested in setting up a variety of use cases and extend their own cyber range.
- Organizers of training courses may be introduced to cyber ranges for the first time. This may shift their view on how lectures and exercises can be designed for local or remote training courses.
- Organizers of cyber exercises may be interested in learning on the variety of cyber exercises and the complex scenario building. The example can also help people who would like to start hosting their own cyber exercises.

- Researchers benefit from learning how such environments can be used to develop and enrich their own scientific work by using dynamic and realistic environments.

The rest of the paper is structured as follows. Section 2 outlines related work within the context of cyber ranges and cyber security testbeds. Section 3 describes the requirements and architectural decisions. Section 4 outlines the implementation. Section 5 demonstrates the cyber range using five use cases. Lastly, Section 7 concludes the paper.

2 Background

This section summarizes related work in the area of cyber ranges and exercises.

2.1 Cyber Ranges

In a survey by [2], several cyber ranges are reviewed and the term "cyber range" is investigated. Furthermore, the NIST [3] defines that "*cyber ranges are interactive, simulated representations of an organization's local network, system, tools, and applications that are connected to a simulated Internet level environment*". The European Cyber Security Organisation (ECSO) [1] defines cyber ranges as "*a platform for the development, delivery and use of interactive simulation environments. A simulation environment is a representation of an organisation's ICT, OT, mobile and physical systems, applications and infrastructures, including the simulation of attacks, users and their activities and of any other Internet, public or third-party services which the simulated environment may depend upon. [...].*" Furthermore, authors in [4] conduct a systematic review of cyber ranges and, for example, the technology used. It provides an extensive overview of cyber ranges (e.g., utilized technology and tools). Also, another review by [11] investigates cyber ranges and testbeds for education, research and training. Design considerations when hosting such cyber security testbeds have to be taken into account (e.g.,[6, 12]). Several cyber ranges have published their design and use cases such as [13, 14, 15].

As the survey in [4] shows, there are a vast number of testbeds or cyber ranges. It is challenging to compare our development to other cyber ranges without more details. First of all, not every testbed or cyber range has publicly available information on architecture or implementation. Also identifying potential testbeds and cyber ranges might be cumbersome. As a starting point to compare architectural building blocks, the functional architecture of cyber ranges given in [4] could be utilized. For example, most of the building blocks are also represented in the AIT cyber range. Furthermore, we use a very similar vocabulary as outlined in the taxonomy suggested by [4] in the AIT Cyber Range specification. For example, we use the same terminology for "*Scenarios*", "*Storylines*". However, "*Environment*" is an infrastructure in our design specification. These considerations support that functional aspects or designs could be compared to other cyber ranges. In particular, the details are essential. A comparative analysis, for example, might need more in-depth exchange between cyber range providers. This publication aims to share architectural designs and implementation specifications in order to start exactly this conversation and foster the exchange of results.

Furthermore, several reviews have investigated cyber-physical systems testbeds, in particular for industrial control systems (ICS) [16, 4, 17]. The reviews show that there are already many designs, concepts and schemes how ICS are embedded or connected to cyber ranges. In the use case in Section 5.4.1, we utilize ICS components to simulate components of a nuclear plant.

2.2 Cyber Exercises

A variety of cyber exercises has emerged in the past 15 years. Cyber security exercises may aim at different goals (e.g., to build competences, to assess competences or to just have fun). Cyber security

exercises can be structured and designed in various ways (cmp. [18]). For example, Capture-The-Flag exercises (e.g., iCTF [19], DEFCON CTF, NYU-CSAW) are designed so that participants (teams or individuals) capture a certain flag (e.g., a file or text). CTFs often use specifically designed platforms [20]. Other examples are cyber security exercises or cyber defense exercises (CDX) (e.g., [21, 8]) that are often hosted in virtual environments such as cyber ranges. How to investigate the performance of trainees in cyber ranges has been proposed in [22]. Lastly, table-top exercises are also a common method to conduct cyber exercises with participants from various domains (cmp. [23, 24]). They often use cards, board games or apps to support their game.

3 Design of the Cyber Range

3.1 Initial Motivation and Requirements

The motivation to start and develop our own infrastructure and software provisioning was driven by several aspects that we wanted to address and at the time, about 5 years ago, there was no open source infrastructure or software available. The motivation to develop a cyber range was: (1) to develop individual threat scenarios that can be hosted and executed in a planned way; (2) to establish industrial control systems (ICS) on the cyber range; (3) to enable a flexible simulation infrastructure that could be customized for different occasions and customers; (4) to enable scalability from small to large scenarios and infrastructures; and (5) to utilize open source technology and to contribute to this community. Based on these requirements, we developed an architecture (see Section 3.2) and implementation (see Section 4).

3.2 Architecture

In this section, the architecture of the Cyber Range is described. In particular, we outline its components solely from a conceptual point of view without addressing implementation details which are discussed in Section 4. The architecture consists of four system modules (i.e. building blocks): computing platform, infrastructure provisioning, software provisioning and scenario engine. These modules in combination are able to fulfill the above mentioned requirements. Modules all have a distinct purpose and are only loosely dependent on each other, so as to make changing the underlying technologies or implementation as easy as possible.

3.2.1 Computing Platform

At the core of every cyber range is the ability to simulate and integrate systems to build complex networked infrastructure setups. The computing platform is the module that facilitates and makes this possible. This role can be fulfilled by most modern Infrastructure as a Service platforms (e.g., OpenStack, AWS) or virtualization stacks (e.g., VMware). Selecting which computing platform to use is very important as it will directly influence what can be simulated and integrated into the cyber range.

3.2.2 Infrastructure Provisioning

Cyber security testbeds are one of the core features of a cyber range. The infrastructure provisioning module is the component that is used to create these testbed configurations and orchestrate them on the computing platform. For this a software solution is needed with which it is possible to efficiently design, create, store and orchestrate complex infrastructure networks.

Many of the technologies that can be used as computing platforms already have capabilities which can be used to implement this module integrated in their feature set (e.g., OpenStack Heat Templates), but

there also exist solutions which provide the required features for multiple different computing platform technologies (e.g., Pulumi or Terraform). Supporting multiple computing platforms on the infrastructure provisioning level has the advantage of being more flexible when choosing a computing platform. It can also make it easier to port a testbed from one platform to another if the target platform is also supported by the chosen infrastructure provisioning technology.

3.2.3 Software Provisioning

Using the computing platform and infrastructure provisioning modules it is possible to create large complex networked infrastructure. The software provisioning module is used to add actual functionality to machines in these cyber security testbeds. This means while the computing platform and infrastructure provisioning is used to model and create the connections and system parameters (e.g., CPU count) for a testbed, the software provisioning module is used to model the roles the machines have within the scenario.

For example a small testbed might consist of three machines connected to a router as defined within the infrastructure provisioning module. Now one of the three machines is designated as an mail server while the other two are client machines. Using the software provisioning module it must be possible to define which software must be installed on these machines based on their roles and also which software versions and configurations to use.

This makes the software provisioning module an essential application deployment and configuration management tool for use within the cyber range. Therefore, it can be realized with one of the many already available software solutions for this use case (e.g., Ansible, Puppet).

3.2.4 Scenario Engine

The scenario engine is the module that is used to define the flow of a cyber range scenario. It turns the static infrastructure created by the other three modules into a living system. This module is mainly used to extend the functionality of the AIT Cyber Range to support not only static use cases, such as security testbeds, but also dynamic cyber range activities, such as cyber exercises.

The scenario engine must support two basic features to achieve this. First, it needs to be possible to define a series of injects (i.e., actions within the context of a scenario, e.g., sending of a message) during development of a scenario. Second, during the execution of a cyber exercise these injects need to be automatically executed to establish the dynamic scenario to which cyber exercise participants react.

Depending on the level of sophistication of the scenario engine, it might be possible to employ it in other cyber range use cases such as security research. A scenario engine that supports the definition of complex cyber attacks as injects can be used to automate an attack chain. Such automation can, for example, be useful for security research where repeatability is important for the verification of research results.

4 Implementation of the Cyber Range

This section gives a brief description of the technological design decisions and implementation of the AIT Cyber Range. The AIT Cyber Range code repositories can be found at `https://github.com/ait-cs-IaaS/`.

4.1 Computing Platform

As already mentioned in Section 3.2.1, there are many suitable software solutions available that can fulfill the function of a computing platform. Alternatively, it would also be possible to develop such a platform specifically for usage in a cyber range, if the resources for this are available. The AIT Cyber Range uses a self hosted OpenStack cluster as its compute engine. OpenStack was chosen as our computing platform due to its open source nature and high level of adoption.

The AIT Cyber Range consists of a mostly default OpenStack (`https://www.openstack.org/`) configuration run on multiple Ubuntu-based nodes. Within our research group we have multiple teams that require access to the cyber range for use in various research topics. Testbeds of these teams needed to be isolated from each other. For this we utilized OpenStacks tenant isolation features. Each team is setup within their own OpenStack domain that is only able to access and view resources allocated to them. Within a team it is possible to create multiple testbeds by separating them into projects.

4.2 Infrastructure Provisioning

As mentioned above, OpenStack already has all the features required for the infrastructure provisioning module. Namely, the OpenStack Heat project makes it possible to define infrastructure using so-called Heat templates. While OpenStack Heat would provide everything we need, we decided to use the infrastructure as code tool Terraform (`https://www.terraform.io/`) instead, as it supports a wide variety of computing platforms.

Using Terraform, we are able to define complex infrastructure modules which can be reused across multiple testbed configurations. Due to its infrastructure as code nature it can also version and store our testbed infrastructure definitions using common code versioning systems (e.g., GIT). This modularized infrastructure approach allows us to quickly develop complex testbeds only using a few hundred lines of Terraform code. Listing 1 shows an excerpt of an example configuration that creates two networks. Module "*internet*" creates a network with parameters and module "*locnet*" creates a local network with a router and connects to the internet.

Listing 1: Example Terraform Network Configuration

```
module "internet" {
  source       = "redacted/terraform-modules/openstack-inet.git?ref=v1.1"
  cidr         = "240.128.0.0/10"
  router_name  = var.router_name
  dns_image    = var.image
  external_dns = ["IP redacted"]
}

module "locnet" {
  source          = "redacted/terraform-modules/openstack-vmnets?ref=v1.5.2"
  host_name       = "fw-${var.locnet_name}"
  host_image      = var.loc_fwimage
  host_flavor     = var.loc_fwflavor
  host_size       = var.loc_fwsize
  host_use_volume = var.loc_fw_use_volume
  host_tag        = "firewalls"
  sshkey          = var.sshkey
  extnet          = var.extnet
  ext_subnet      = var.ext_subnet
  networks = {
    locnet = {
      network            = var.locnet_name
      subnet             = "subnet-${var.locnet_name}"
```

```
      cidr                = var.loc_cidr
      dns                 = var.loc_dns
      host_address_index  = var.loc_gw
      host_as_dns         = false
      routes = [
        {
          cidr = var.sec_cidr
          gw   = cidrhost(var.loc_cidr, var.fw_sec_loc_gw)
        }
      ]
    }
  }
}
```

In Terraform actual infrastructure instances and its code representations are synchronized using the Terraform State. This state can be either stored as a local file or on a remote storage server. The AIT Cyber Range uses a HashiCorp Consul server for remote state storage. States for the various testbeds and their modules are stored in key prefixes relative to the OpenStack domain and project the testbed is instantiated in. This state storage scheme allows us to reuse testbeds across teams and projects without the need to duplicate infrastructure code.

Infrastructure components are assigned labels based on their role within the scenario as part of the testbed configuration. This labeling is achieved by configuring OpenStack compute node metadata as part of the Terraform configuration. The labeling can then be used by the software provisioning module to map machine instances to their roles and apply software configuration accordingly.

4.3 Software Provisioning

The AIT Cyber Range software provisioning module is implemented using the configuration management tool Ansible (`https://www.ansible.com/`). Ansible provides us with the ability to define software deployment and configuration as templatable code. Similar to Terraform, it also allows us to modularize our Ansible code using Ansible Roles.

Using Ansible Roles we can define software configurations (e.g., for a Postfix server) which can be reused for multiple machines and testbeds. Many members of the Ansible community also provide their Ansible Roles under open source licenses, making it possible for us to integrate their roles as part of cyber range testbeds. Tasks such as the configuration of multiple users on a system can be achieved in a few lines of Ansible code. Listing 2 shows an example Ansible Playbook used to configure a web server.

Listing 2: Example Ansible Configuration of a Web Server

```
name: Setup and configure the organisation webservers
  hosts: webservers
  become: true
  roles:
    - role: wordpress
      vars:
        wp_options: "{{ wp_options_all | combine(wp_options_group | default({})) | 
            combine(wp_options_host | default({})) }}"
  [...]
```

As mentioned above in the AIT Cyber Range all machines are assigned their roles through metadata labels as part of the Terraform infrastructure configuration. The labels and connection information needed for applying Ansible configuration to the deployed infrastructure is read using the Ansible OpenStack inventory provider. For the software provisioning process we use a special management host which has been configured to be connected to all network zones that are contained within a testbed while also

being assigned a public IP from the OpenStack domain's public network. This allows our cyber range to connect to testbed machines without impacting the simulated network and its various zones.

The usage of labels, the OpenStack inventory provider, and the management host allows us to decouple the software provisioning from the actual infrastructure configuration to some degree. Since all the relevant information can be read at runtime, there is no need to know the actual network structure or IP addresses of specific machines in the context of the software provisioning configuration. This allows us to create complex configurations which can be applied to multiple differently structured testbed infrastructures, as long as the contained roles can be mapped to a machine by use of the labeling system.

4.4 Scenario Engine

In the AIT Cyber Range the scenario engine is currently redesigned and developed in two components: the Scenario Builder and the Scenario Runner. The Scenario Builder is a web-based application that supports the specification of scenarios (see Figure 1) define the injects of a script (see Figure 2) and manage the list of scenarios (see Figure 3).

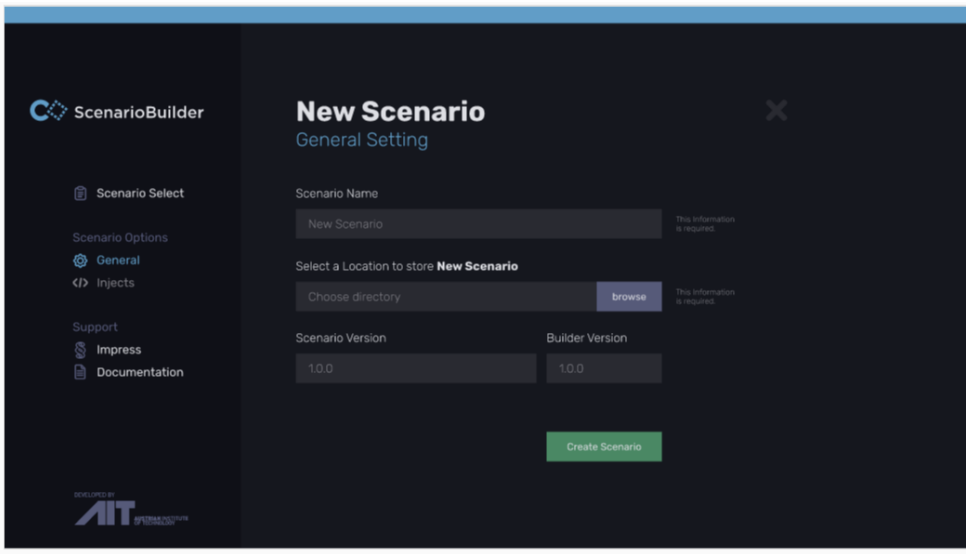

Figure 1: Scenario Builder: Creation of a Scenario (Mockup)

The Scenario Runner, shown in Figure 4, is a web-based application used to control the scenario flow as well as to execute injects within the context of a cyber range testbed. The Scenario Runner is deployed as part of the above mentioned management host during the software provisioning process. It has network access to all systems within a testbed.

The previous version as outlined in [10] was limited in its capabilities and was mostly used to direct non-technical flow of cyber exercises, e.g., participant instructions or predefined non-technical events, such as email communication from a Computer Security Incident Response Team (CSIRT). The new Scenario Engine will allow for more complex technical injects, e.g., running fully automated attack chains against a testbed infrastructure.

5 Use Cases

This section summarizes the main use cases of the cyber range.

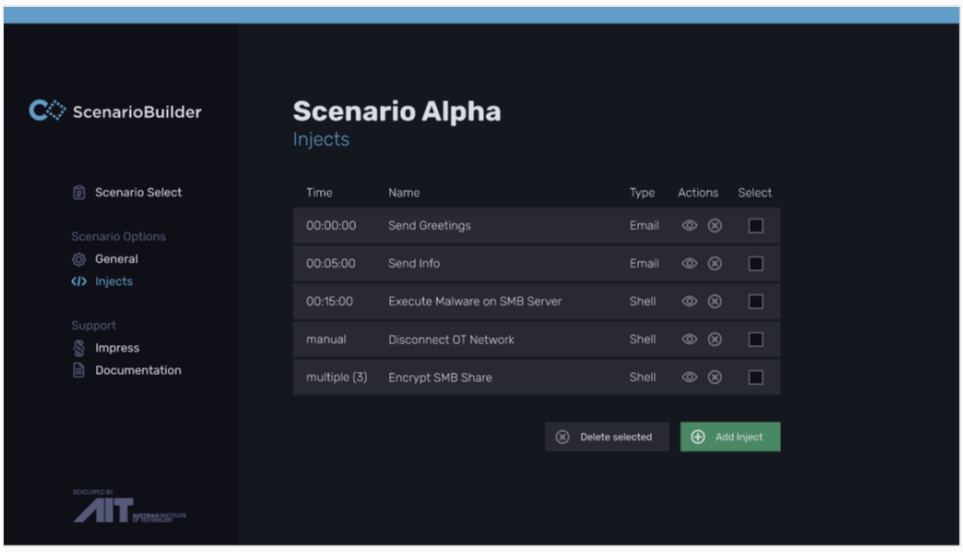

Figure 2: Scenario Builder: Example Scenario Alpha (Mockup)

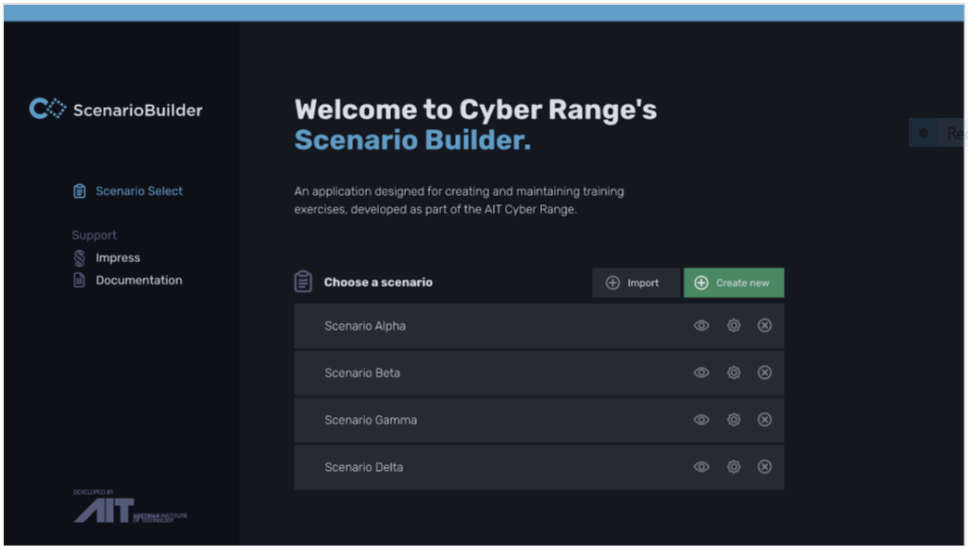

Figure 3: Scenario Builder: Overview of Scenarios (Mockup)

5.1 Cyber Exercises

Cyber security exercises can be utilized to test, train and validate the organizational and technical competencies of organizations and individuals. Cyber security exercises can be structured and designed in various ways (cmp. [18]). In the AIT Cyber Range, several exercises have been conducted that are summarized in the following:

- The *intra-organizational cyber security exercise* emphasizes on validating strengths and capabilities of a single organization. The goal is to increase the organizations incident response management. The exercise consists of up to 20 participants that work in different departments of the same organization (e.g., IT, Legal, General Data Protection Regulation (GDPR), Human Resources, IT Security).

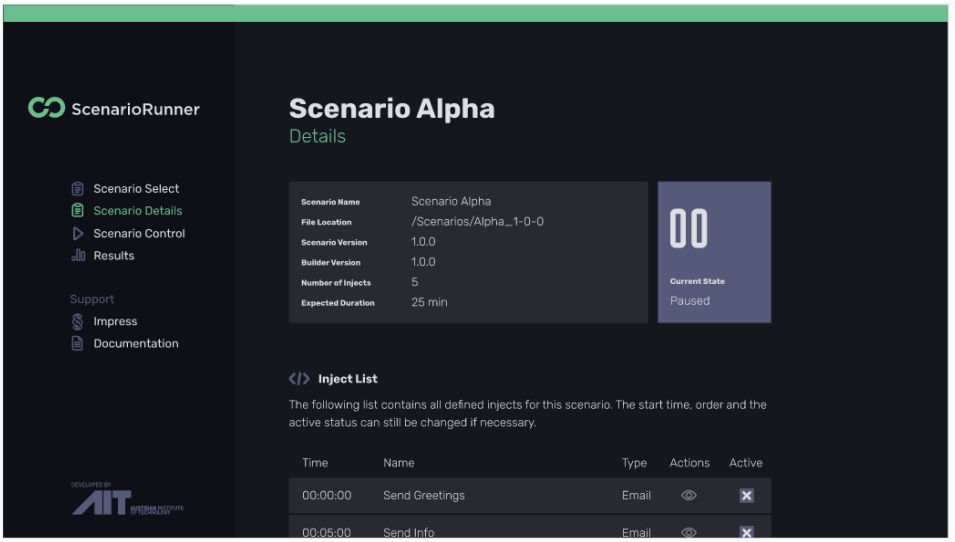

Figure 4: Scenario Runner: Execution of the Example Scenario (Mockup)

- The aim of a *national, cross-sector cyber security exercise* is to raise awareness and increase the capabilities of several stakeholders (e.g., operators of essential services (OES), public authorities, large organizations, Small and Medium Enterprises (SMEs)). The exercise is designed to establish and maintain technical skills and competences, enable cyber security awareness and information sharing (cmp. [25]).

- The *international, cross-organizational, cross-sector cyber security exercise* focuses on the global challenge of incident response, incident management and information sharing across countries and sectors. This exercise consists of various teams (e.g., SMEs, CSIRTs, OES or public authorities). Together, the teams work on mitigating and minimizing threats and attacks.

With the flexibility of the architecture, the cyber range provides the infrastructure, participant access, technical scenario and injects to host these exercises.

5.1.1 Example: Intra-organizational Cyber Exercise

A typical intra-organizational cyber security exercise in the cyber range is described as an example in the following. Please note that this is only a brief summary. Exercise script and detailed injects were not included but are summarized in the following paragraph.

Technical Scenario In this example, we develop a technical scenario that consists of several story lines. In each story line, we focus on one specific incident or attack. The story lines are:

1. Ransomware attack: The ransomware attack story line is our mostly valued and requested story. It is about a ransomware attack that starts with emails to several employees with a malicious attachment. While several employees recognize the motivation, a few employees do not and install malware on their computers. This allows the attacker to gain access to the network and then continues to investigate potential targets, harvests information, accounts and passwords and tries to access other networks. This story is continued until the attacker has gained access to a highly secured network and can access highly sensitive and business-relevant data.

2. Data leak: This story focuses on an incident that leads to the (unintentional) disclosure of personal data which needs to be reported according to the EU General Rule of Data Protection (GDPR).

3. Spear phishing: Several employees receive sophisticated emails in an organization that requests information on current accounts or other confidential business-relevant information (e.g., top customers, etc.). Unfortunately, a few employees do not see that this request is from an external entity and reply with confidential documents attached.

Participant Setup In the example exercise, we expect several participant roles. The white team (exercise control) hosts the exercise and monitors its progress. It starts, pauses and stops the exercise when needed and is supported by the green team, i.e., the team that sets up and monitors the infrastructure during the exercise. The blue team consists of the participants of the exercise. Often, the participants in this example exercise represent security operations center (SOC) teams or CSIRTs.

Scenario Infrastructure While the underlying cloud infrastructure has been described in Section 4, this section refers to the infrastructure that the blue team uses. For example, in OpenStack this is a project that is set up. Figure 5 displays the example scenario infrastructure. The scenario infrastructure consists of two networks *(IT network* and *security zone*). In the IT network, several employee laptops/computers are connected as well as other server infrastructure (e.g., shares, email) and services (e.g., processing, backup) are running. The security zone is a network with higher standards regarding the availability, confidentiality and integrity. In the network, we have several running instances of servers that host confidential data (e.g., personal data, business relevant data). While Figure 5 only indicates a few interconnections between some servers, there are others that are not visible in this diagram.

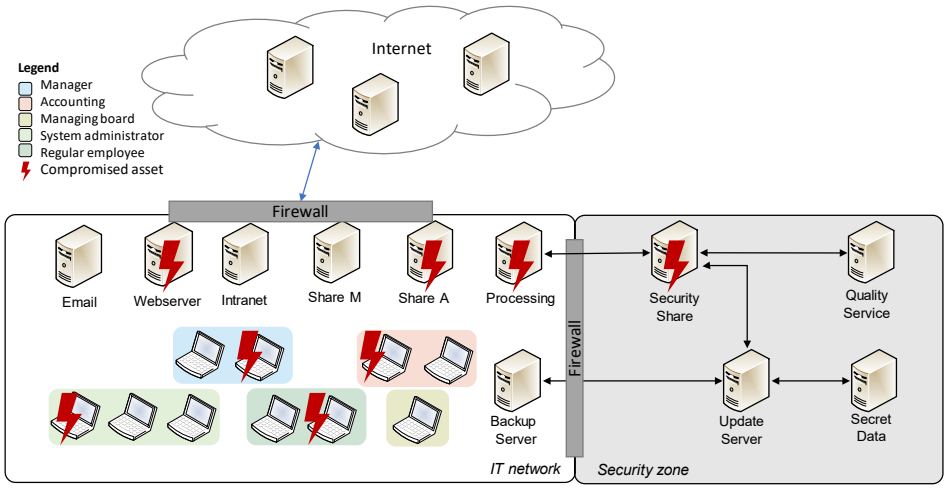

Figure 5: Scenario Infrastructure Example

Figure 5 also highlights the attack path in the ransomware story line using the flash symbol. Employees receive emails with malicious attachments and unfortunately several install malware on their computers. In the following, the attackers gain access to the network and start investigating other servers. They acquire user access to different machines and utilize a common vulnerability exposure (CVE) to gain access to the secure network zone.

5.2 Professional Training

The cyber range can be set up to host regular training courses for educational purposes (e.g., professionals). For example, the infrastructure in the cyber range can be set up to reflect the production systems of SMEs or large organizations in various sectors (e.g., IT, energy, manufacturing, finance, healthcare). Often, training content is a combination of a theoretical elaboration and a practical exercise (e.g., hands-on practice). In our experience, this is a very efficient way to adopt new methods or practices. So far, the cyber range has provided a training platform in the following training courses:

- In the *Information security* course, injects that are used in the cyber security exercises are utilized for e.g., incident response, network security, malware analysis or forensics. These hands-on training courses target specifically professionals, researchers or others who are interested in information security training (cmp. [6]) and improve their dexterity.
- The *Computer security in industrial control systems* training course provides an overview of industry-specific protocols and technologies. Furthermore, it addresses important aspects of ICS security such as network security, access control or physical security.
- The cyber range has hosted customized *Industrial training* courses to increase skills and competences of employees in Industry 4.0. These training courses contribute to the skill development of the workforce. The training was developed as an additional way to support the digital transformation of work places.
- The *Strategic cyber security awareness* course focuses on a discussion-oriented setting of cyber incidents and attacks on an organization. While most of the course is originally not action-oriented as it focuses on the participants, their perspectives and a shared strategy, the demonstrations and contents are prepared in the cyber range. Figure 6 shows an example question on investments that is directed to participants. Figure 7 outlines a question that addresses decision making as Chief Information Security Officer (CISO). The utilization of questions aims at stipulating discussions and active decision-making. So far, we have been utilizing Mentimeter[1] but are currently working on a new software for strategic cyber security exercises.

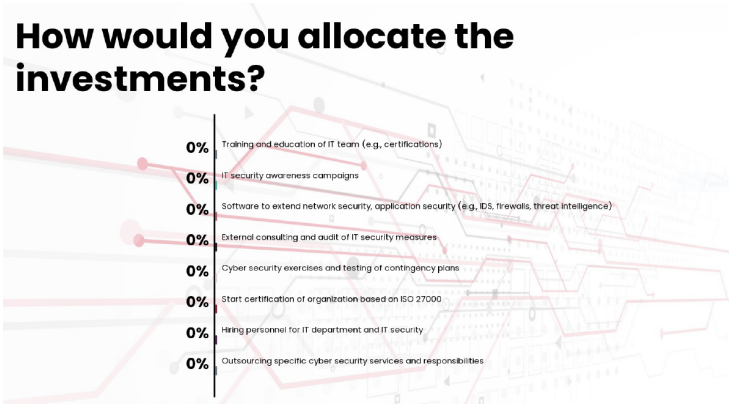

Figure 6: Example Strategic Cyber Security Awareness Training (1)

[1] http://www.mentimeter.com

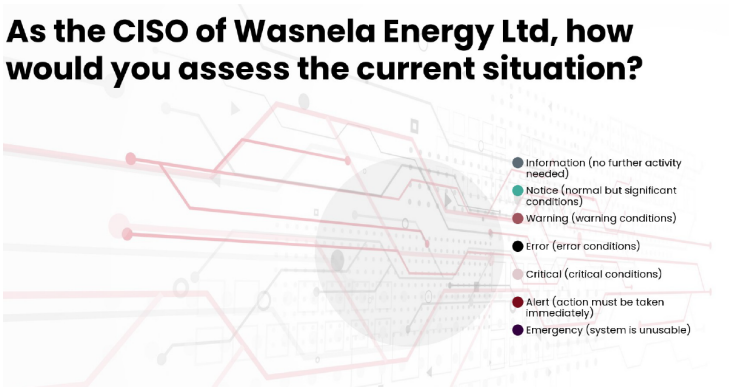

Figure 7: Example Strategic Cyber Security Awareness Training (2)

5.3 Higher Education

Cyber security is an increasingly important domain in university education. There is clearly a broad demand in educational programs for a competence and capacity building, but cyber security requires a holistic and multidisciplinary perspective. As such, it presents a challenge for institutions and educators to create an effective offering and an appropriate experience for learners. The AIT Cyber Range can address this gap in teaching. For this purpose, it can be used as a training and learning tool to meet different requirements and needs of both target groups. To achieve this, the Collaborative Cybersecurity Awareness Learning (COLTRANE, `https://coltrane.ait.ac.at/`) project is developing a methodology that will combine new and existing approaches and methods from teaching and learning research with the potential teaching strategies offered by the Cyber Range. This enables the teachers to provide a real-life and practice-oriented education in the field of cyber security by applying problem-based, multi-perspective and communication-oriented lessons. This leads to the acquisition of cross-disciplinary competences and skills by the students. In particular, the Cyber Range enables students to develop collaborative and self-reflective approaches and to enhance their understanding and awareness of cyber security through problem-based and independent learning of the content. With the integration of a cyber range and harmonization of education and training, different needs and goals of the target groups (students and teachers) can be achieved. This means that students are better prepared for today's and tomorrow's challenges in cyber security and can compete on the national and international professional market. Through the use of the teaching method and the Cyber Range, teachers can provide and deliver teaching content tailored to the target group, focused on the level of knowledge and specific to the topic. These dynamics will minimise the workload and the preparation of the teachers, by providing a scenario in an abstract manner, it is possible to combine different teaching concepts (theoretical exercises, collaborative hands-on examples, interactive learning content) and to facilitate the elaboration of possible solutions. This allows a uniform integration of educational methods and tools for teaching cyber security in order to create a common standard to enable comparability and cooperation between different groups of people and thus ensure security in the digital space.

Figure 8 illustrates a potential approach to cyber security education, which can be used as a basis for a course in cyber security. A higher education module in the field of cyber security is shaped by the objectives that need to be achieved, but also by the methodology of the lecturer. This methodology can involve a conceptual approach or a combined approach depending on the type of lesson and the context of cyber security that it is intended to address. It is important for the understanding and awareness building phase that the examples shown are related to or applied in the real world. This approach enables students to define realistic scenarios themselves and evaluate and test those scenarios in an infrastructure.

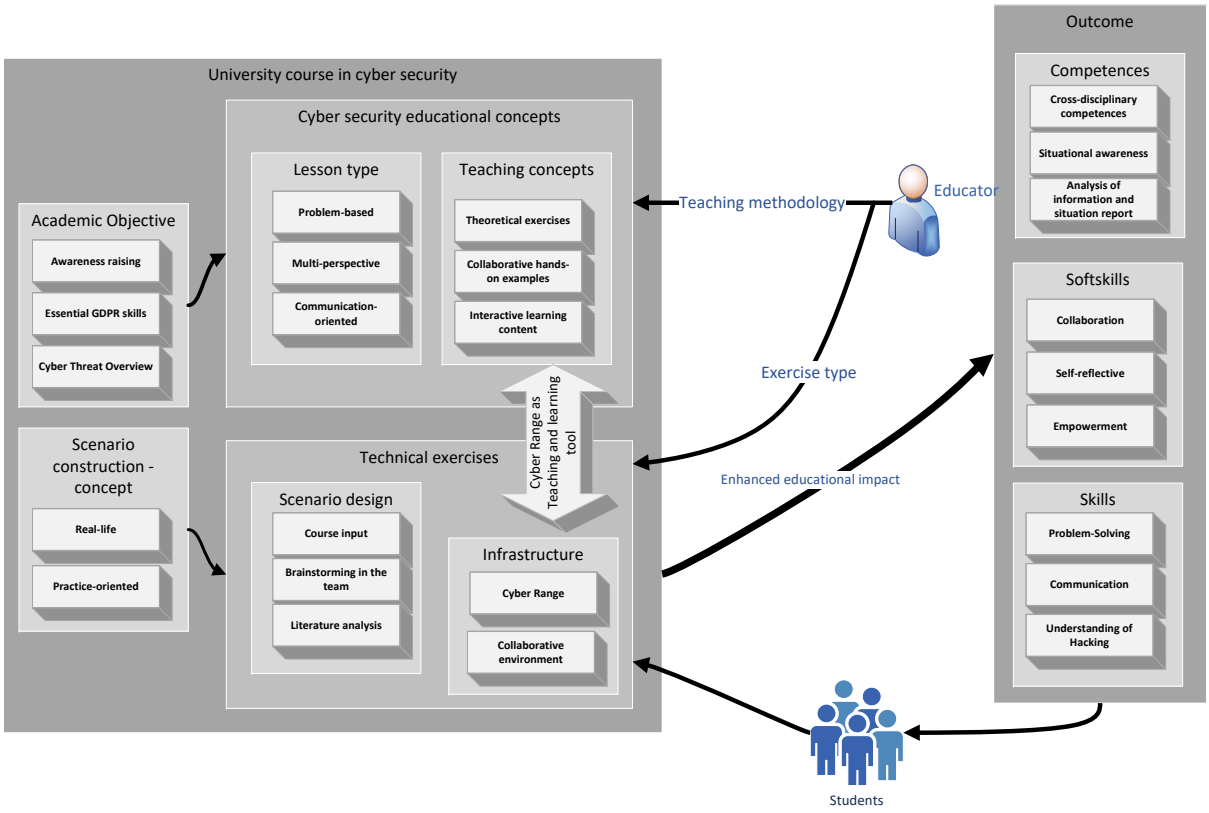

Figure 8: Cyber Security Education Example

As a result, the students not only learn the specified objectives in a practical way, but also transfer competences beyond their area of expertise, soft skills which are increasingly being demanded in the working environment, and strengthen their skill set.

5.4 Security Research and Development

For research and development, the cyber range (1) is used as testbed to develop and test new approaches and methods (e.g., defense methods) and (2) is the basis for specific cyber range research (e.g., federation, scenario generation, measurement). In the following, two research areas are summarized.

5.4.1 Simulation and Detection of Attacks in Industrial Control Systems

A major challenge is evaluating cybersecurity solutions for industrial control systems (ICSs) that support critical infrastructures. It is not possible to perform tests on production environments, because of their high availability requirements. Consequently, it is desirable to perform research into cybersecurity solutions using cyber ranges that include representative ICS equipment and the processes they control. In many cases, it is important to simulate a process under control to determine the potential consequences of a cyber-attack whose purpose is the sabotage of that process. (The canonical example of this form of sabotage is the Stuxnet virus, which resulted in damage to centrifuges at an Iranian nuclear facility [26].)

In the context of an IAEA-funded research project, we have used the AIT Cyber Range to support research into computer security incident response and analysis at nuclear facilities. In the project, we have investigated approaches to detecting cyber-attacks on edge devices, such as Programmable Logic

Controllers (PLCs) [27], and approaches to root cause analysis that could facilitate computer security incident response [28]. To perform this research, it was necessary to develop threat scenarios – largely targeted at sabotaging the operation of a nuclear facility – that can be executed on a testbed that is representative of a nuclear facility. We used the AIT Cyber Range to realize this testbed.

The testbed consists of three main parts: *(i)* a virtualized environment that consists of representative IT and Supervisory Control and Data Acquisition (SCADA) systems, which are arranged into a defensive computer security architecture; *(ii)* a collection of ICS equipment, including several PLCs and Human Machine Interfaces (HMIs) that control and visualize simulated processes, respectively; and *(iii)* a MATLAB/Simulink simulation model of a Pressurized Water Reactor (PWR), which was developed in the project, called Asherah [29]. The PLCs in our testbed control the pressure level in the pressurizer vessel of the simulated Asherah PWR.

The virtualized nuclear facility environment consists of several security zones that are used to support functions that have different levels of criticality, e.g., from enterprise systems through to those that are being used to engineer and control key processes, such as reactor cooling. (More specialized SCADA systems that have been implemented include engineering workstations that are used to program devices, such as PLCs, data historians, and systems that implement a main control room HMI, for example.) The implementation of zones is realized using virtualized security controls, such as firewalls. The configuration of these zones and the systems within them are described using Ansible scripts that have been defined by partners in the project. The use of Ansible has enabled partners – and Member States that wish to leverage the results of the project – to easily share configurations that can be instantiated on their testbeds. This significantly reduces the effort that is required for partners to implement their testbeds and has supported the realization of a common environment in the project, which enables the sharing of useful artefacts, such as datasets and tools, between partners.

5.4.2 Evaluation of Intrusion Detection Systems

Intrusion detection systems (IDS) are tools that analyze computer system or network behavior and report suspicious activities. They contribute to system security by recognizing patterns that are known to relate to adversaries [30] within large amounts of complex data or disclose deviations from normal behavior without human intervention. The ability to measure and compare the performance of IDS objectively in a safe and secure manner is essential for improving their algorithms and providing new research directions [31]. However, most IDS are designed, or at least specifically configured, for application in particular environments and focus on the detection of pre-determined attack techniques. Accordingly, objective IDS benchmarking for selection and deployment in real world applications is not trivial [32]. For this reason, research groups have developed testbeds that resemble real networks and allow IDS deployment as well as attack execution in controlled environments [33].

Testbeds are essential to validate, evaluate, and compare the capabilities of IDSs. They offer analysts the opportunity to challenge IDS with a wide variety of attack scenarios, which is the basis for unbiased investigations. Otherwise, it is not possible to reliably assess whether the IDS under test performs with similar efficiency and effectiveness when deployed in productive operation. Setting up testbeds for particular use-cases is usually time-consuming. The main problem is that analysts are stuck with inflexible testbeds that are set up once by domain experts that could not predict the requirements that emerged after setup.

Another problem is that most existing testbeds are relatively static, because their configuration relies on manual input and domain knowledge. This prohibits fast instantiation of different testbeds with variable configurations. The AIT Cyber Range applies a model-driven approach and offers the possibility to obtain multiple testbeds with variations, which is highly beneficial for IDS evaluation. More available data representing different technical environments enable the generation of separate training, validation,

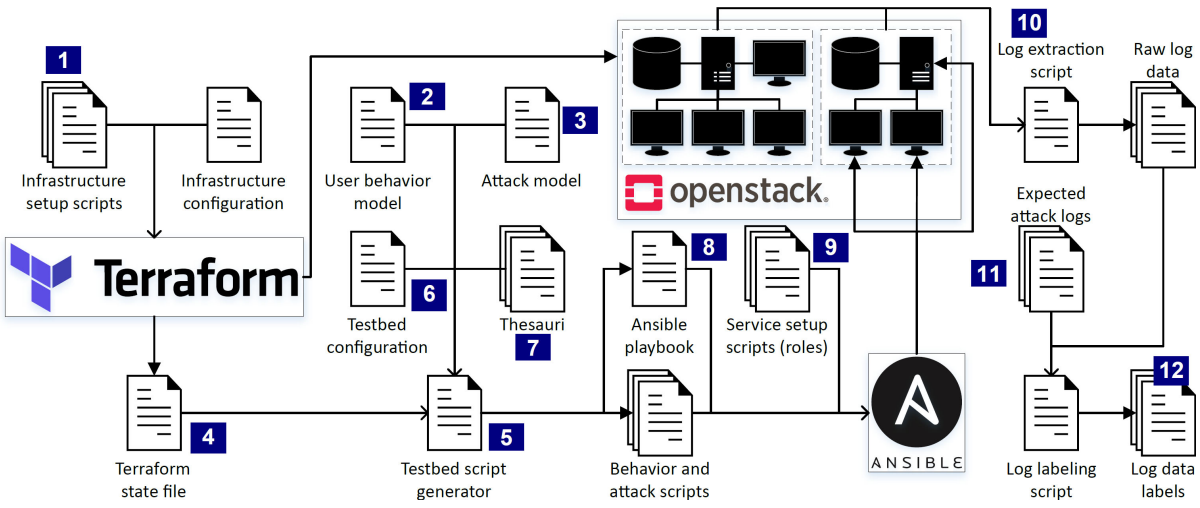

Figure 9: Technical implementation of the model-driven testbed and log data generation approach. Simple arrows indicate imports and filled arrows indicate generation of resources such as scripts, configuration files, or machines.

and test data sets, improve robustness of evaluation results, and support validation of approaches in different application environments.

In particular, we seek for commonly available infrastructures that are frequently subject to attacks, such as servers that are accessible over a network. We also look for frequently installed packages and examine the settings of the logging services. Given a real infrastructure, it is also possible to monitor the exhibited behavior and derive relevant characteristics of normal system usage, such as usage distributions over a period of time [33]. Finally, attacks are either observed on the real infrastructure or exist in documented form in online threat databases[2].

Our model-driven approach [34], as outlined in Figure 9, defines testbed-independent models (TIM). Regarding the infrastructure, this implies declarations of the setup routines (#1) for all involved components and services without specifying any concrete parameters. For example, we define how a component is connected to the network, but do not allocate IP addresses, assign names, or specify the number of users, but only the type and range of these parameters. Similarly, we design a model of the system behavior (#2) as a state machine without fixed transition probabilities between its states, and a model for the attack scenario (#3) that consists of the basic steps that are necessary for carrying out the attack. All TIMs function as templates, i.e., they are scripts that represent specific routines, but are configurable through consciously placed parameters throughout the code.

Our transformation engine that generates testbed-specific models (TSM), processes the templates and inserts all parameters to produce executable code. The parameters are thereby selected randomly based on their type specified in the TIM. For example, the number of simulated users is selected from a predefined range, their names and passwords are picked from predefined lists, and IP addresses are automatically assigned from a pool. Since transformation of TIMs to TSMs is fully automatic, it is possible to generate arbitrary amounts of TSMs at the same time, where each TSM exhibits variations depending on the settings for random selection.

As discussed earlier, we use the infrastructure-as-a-service tool Terraform to instantiate the testbed infrastructures as virtual machines on an Openstack cloud platform using our predefined setup scripts. Configurations at this point involve the total number of machines, operating systems, and computational resources, e.g., memory. Building the machines with Terraform yields a so-called state file (#4) that con-

[2]For example, `https://www.metasploit.com/`

tains deployment information, such as IP addresses. The testbed script generator (#5) implemented in Python that acts as the transformation engine of our model-driven proof-of-concept implementation imports the state file together with a configuration file (#6), system behavior and attack TIMs, and thesauri (#7), i.e., word lists arranged by topics such as usernames, passwords, and host names. The configuration contains lower and upper limits for parameters that are randomly chosen when generating TSMs, i.e., files and executable scripts. Moreover, the transformation function generates a playbook (#8) that specifies the services to be installed, which are referred to as roles. Examples for such roles are PHP, Apache for web server setup, MariaDB for database setup, suricata IDS (`https://suricata.io/`), or Internet browsers. Each role requires a setup script that states (#9) a list of tasks to be carried out. Thereby, it is possible to use variables in the playbook to specify random modifications of the setup process, e.g., install different versions, or replace them with alternative roles altogether. We then use the application-deployment tool Ansible to distribute all generated files, set up services, and start the execution of user and attack scripts.

Notice, for the user behavior, we specify a number of profiles with ranges for transition frequencies that the transformation engine translates into probabilities. Regarding the attack scenario, optional parameters of individual steps as well as their order and delays are randomly selected. Note that modeling may be based either on attacks or vulnerabilities, i.e., an attack model scenario may focus on a single malicious action or involve several vulnerability exploits and diverse attack vectors.

The right side of Fig. 9 shows that once the simulation is complete, another script (#10) collects all log files from the virtual machines and stores them on disk. We then automatically label the logs using attack execution information extracted together with the other logs as well as a predefined dictionary of expected log lines (#11) for each attack step. We store the lists of generated labels in separate files (#12).

While with the generated raw log data a wide range of intrusion detection systems are challenged, their detection results, i.e., whether they classify a log line as malicious or not, can be compared to the ground truth given through the log data labels. This way, the detection performance of different IDS, algorithms and configurations can be objectively rated thanks to our testbed running on top of the AIT cyber range.

6 Discussion

6.1 Main Results

In Sections 3 and 4 we show the AIT Cyber Range platform design and implementation. The platform consists of four loosely coupled components (Computing Platform, Infrastructure Provisioning, Software Provisioning and Scenario Engine). The AIT Cyber Range implements these components using OpenStack, terraform, Ansible and our in-house developed Scenario Engine. The loosely coupled design makes it possible to replace any of the chosen technologies (e.g., using AWS instead of OpenStack as compute platform) to better fit an organization wishing to operate a cyber range. The strong focus on infrastructure and software provisioning as a code also makes created cyber range testbeds easy to reproduce and instantiate. Thus making it possible and encouraging other people to re-use or built on top of previously defined cyber range testbeds.

In addition, we outline use cases for the infrastructure in Section 5: Cyber exercises, professional training, higher education as well as security research and development. As the use cases show, different domains and applications have certain requirements and our infrastructure can be used as a foundation and structure but needs to be prepared and specified in order to be able to accommodate various use cases (e.g., cyber exercises, intrusion detection systems or industrial control systems).

6.2 Lessons Learned

Computer networks simulated in cyber ranges can become very complex; utilizing multiple network zones and many different software configurations. Provisioning such systems manually can often times require multiple days of work. By using terraform and Ansible to automate this process, it is possible to significantly reduce the required provisioning time. Also the amount of labor required during the provisioning is significantly lower as cyber range operators only have to start and verify the provisioning process. For example, we recently conducted a large scale cyber security exercise with a testbed setup containing 165 hosts across 40 network zones. Provisioning of this setup took around one work day of run-time, but it only required around 3-4 hours work time to start, monitor and verify the process. Furthermore the nature of some cyber range use cases (e.g., IDS or professional training) requires a fresh system state, i.e., the simulated environment should not contain any artifacts of previous utilization. Such a clean state can in some cases only be efficiently achieved by completely destroying and re-provisioning the cyber range testbed. Thus it is vital for a cyber range to support automation of the infrastructure and software provisioning.

While automation can significantly cut down the time required to provision a cyber range testbed environment, it also adds complexity to the initial development process. As mentioned above the AIT Cyber Range utilizes Ansible for software configuration and provisioning. Ansible works by creating so called playbooks, a set of configuration tasks defined using a YAML (`https://yaml.org/`) and Jinja2 (`https://jinja.palletsprojects.com/en/latest/`) syntax. Each task describes a certain configuration action (e.g., changing a value a configuration file). These actions are achieved through Ansible modules which often times simply abstract commonly used CLI tools to provide the desired functionality. While most software can be easily automatically provisioned using such an approach some software (e.g., software only providing GUI based setup) requires a bit more initial effort. This extra effort though is quickly offset if a cyber range testbed is setup more than once.

Developing complex cyber range environments and scenarios is a process usually requiring the collaboration of multiple experts. For the AIT Cyber Range this means that the infrastructure and software provisioning configuration must be structured in a way so that it supports this distributed development effort. For this, we have learned to adopt and heavily utilize a modularized approach to our provisioning configuration. This means that we always try to use and create reusable configuration blocks (i.e., Ansbible Roles and terraform modules) when appropriate and then build complex testbed definitions from these base modules. While we keep configuration files for a testbed in a single code repository, we also structure these repositories in a modular way. For example, network layout and host definitions are created as separate terraform modules. In case of especially complex designs, we further split the host configuration module into multiple sub modules reflecting the various network zones. On the software configuration side, we opted for a feature based approach, e.g., DNS and webserver configuration each have their own Ansible playbook. Also variable configuration follows the same approach, i.e., following the Ansible paradigm of host and group variables (vars), we create separate host and group var files for each feature. Our experience has shown that using this modular approach, it is not only possible for multiple people to efficiently work on the same testbed definition, but it also makes it easy to re-use certain aspects of old testbed definitions in new scenario and environment designs.

As already mentioned above the AIT Cyber Range is hosted as an on-premise OpenStack cluster. Staying on-premise has many advantages, such as, the ability to be a 100% sure where and how once data is stored or processed. This can be vital for some customers or research projects as security and privacy policies might disallow usage of certain cloud providers. Also, hosting the platform on-premise gives one the ability to test new compute platform features (e.g., new OpenStack releases or extension projects) easily. This is something which would not be as easily possible in externally hosted platforms as we would have no control over the underlying compute platform configuration. That being said using an

externally hosted platform also has its own advantages. For example, an on-premise platforms resources will always be more limited than externally hosted platforms which allow on demand scaling. Also operating an on-premise platform comes with additional costs (e.g., maintenance, power consumption) which can be avoided by using an on-demand externally hosted service. Generally speaking both options are equally valid and supported by the AIT Cyber Range technology stack. Which option is best highly depends on the use case and team operating the cyber range platform.

6.3 Future Work

Cyber Exercises As mentioned in Section 5.1, we are currently building a new scenario engine that allows to inject different artifacts into the simulation (e.g., cyber-attacks but also emails). This will allow the multifaceted injection of events into cyber exercises and allows the AIT Cyber Range to host cyber exercises from discussion-oriented to action-oriented exercises as well as to support different cyber exercise types such as red team/blue team or king of the hill exercises. Furthermore, we continuously develop injects and content of cyber exercises to build and provide an improved experience for all participants. This includes also integrating new technologies (e.g., [35]), new attacks and new systems and infrastructures. The challenge is here to further develop the technologies and new artifacts that they can be reusable in new scenarios or infrastructures. In addition, we will focus also on scientific efforts to assess and develop complementary aspects of cyber exercises such as gender and diversity management or skills and competencies to investigate and identify how to design, develop and delivery cyber exercises to a broad audience.

Industrial Control Systems For cybersecurity exercises and research, it is often desirable to understand the operational consequences of an attack on a target critical infrastructure. For example, for electrical energy systems, it can be useful to determine whether an attack could result in power system issues, such as a blackout or instabilities. For a cyber range, this can be supported by coupling real industrial control systems equipment with a simulated process, as has been done with the Asherah Nuclear Simulator (see Section 5.4.1). However, critical infrastructures are becoming increasingly interconnected and coupled – this can be seen, for example, with Smart Grids and Smart Cities, wherein ICT systems are used to support the operation of electricity, gas, and transportation networks. In this setting, new challenges arise when attempting to understand and simulate the operational consequences of cyber-attacks, e.g. to determine whether there are cascading effects that propagate between coupled systems. Challenges include addressing timing constraints between simulations of different domains (e.g. electrical energy and gas for coupled energy systems) and the heterogeneity of simulation interfaces. Future work will investigate approaches to so-called *multi-domain simulation* for cyber range applications, such as those discussed in this paper. The intention is to build on existing research that has been undertaken on co-simulation, which has largely focused on process optimization tasks [36].

Intrusion Detection Systems Our testbed for generating log data that is useful for evaluating intrusion detection systems, is an evolving project. New use-cases may require to shift the focus of the overall network layout in different directions, e.g., by adding new components or services. Alternatively, to adequately evaluate new detection techniques, we may adjust or extend the attack chain to include actions that generate particular types of traces in log data. Moreover, all installed software will eventually be outdated and needs to be replaced by alternatives that are commonly used at that point in time in order to ensure that our testbed is representative for real-word situations. For future work, we therefore plan to extend the testbed by adding new modules that may be used as part of the simulations. In addition, we plan to increase the authenticity of the user simulation by deriving relevant parameters, such as the distribution of click delays, idle times, and page visit frequencies, and add them as ranges to our TIMs.

Moreover, our goal is to publish all code, which includes scripts for deployment of all machines as well as simulation scripts for users and attackers, as open-source software so that anyone in the research community may reproduce our results and add their own extensions to the testbed.

Conceptually, we also plan to improve the labeling strategy due to the fact that our dictionary-based technique for identifying logs related to different attack steps is difficult to maintain and prone to errors. Our idea is to introduce a querying language that allows analysts to define rules that match specific logs with high accuracy and thus avoid false classifications. Thereby, these queries should make use of facts extracted directly from the testbed, e.g., IP addresses or user names. This enables the design of labeling rules in an abstract way as part of the TIMs and thus allows to carry out labeling without human intervention on all testbeds that may be generated from the TIMs.

7 Conclusion

This paper introduced the AIT Cyber Range, specifically its design and deployment. The main principles of the architecture are to achieve flexibility, scalability and support five use cases. In particular, it consists of the building blocks computing platform, infrastructure provisioning, software provisioning and scenario engine. With the cyber range, many exercises and training courses have been successfully held. More than 350 participants have joined or competed in one of our exercises or training courses. Furthermore, we have outlined two use cases from security research and development to demonstrate how a cyber range can contribute to the scientific community. With this paper, the authors aim to contribute to the overall understanding and information sharing on cyber ranges.

Acknowledgments

This work was partly funded by two projects: The SIREN project has received funding from the IAEA as part of the CRP J02008 on Enhancing Computer Security Incident Analysis at Nuclear Facilities. The INDUCE project is funded by the National Foundation for Research, Technology and Development. Laura Bassi 4.0 is a research, technology and innovation funding programme processed by the Austrian Research Promotion Agency, with kind support of the Federal Ministry for Digital and Economic Affairs (BMDW).

References

[1] ECSO. Understanding cyber ranges: From hype to reality. Technical report, European Cyber Security Organisation, March 2020.

[2] J. Davis and S. Margath. A survey of cyber ranges and testbeds. Technical Report DSTO -GD -0771, Cyber Electronic Warfare Division, DSTO Defence Science and Technology Organisation, 2013.

[3] NIST. Cyber ranges. Technical report, National Institute of Standards and Technology, 2018.

[4] M. M. Yamin, B. Katt, and V. Gkioulos. Cyber ranges and security testbeds: Scenarios, functions, tools and architecture. *Computers & Security*, 88:101636, January 2020.

[5] C. Pham, D. Tang, K.-i. Chinen, and R. Beuran. Cyris: a cyber range instantiation system for facilitating security training. In *Proc. of the 7th Symposium on Information and Communication Technology (SoICT'16), Ho Chi Minh City, Vietnam*, pages 251–258. ACM Press, December 2016.

[6] M. Frank, M. Leitner, and T. Pahi. Design considerations for cyber security testbeds: A case study on a cyber security testbed for education. In *Proc. of the 3rd IEEE Conference on Cyber Science and Technology (CyberSciTec'17), Orlando, Florida, USA*, pages 38–46. IEEE, November 2017.

[7] S. Kucek and M. Leitner. Training the human-in-the-loop in industrial cyber ranges. In *Proc. of the 1st European Advances in Digital Transformation Conference (EADTC'18), Zittau, Germany, and 2nd European*

Advances in Digital Transformation Conference (EADTC'19), Milan, Italy, volume 670 of *Lecture Notes in Electrical Engineering*, pages 107–118. Springer, Cham, May 2020.

[8] J. Vykopal, M. Vizvary, R. Oslejsek, P. Celeda, and D. Tovarnak. Lessons learned from complex hands-on defence exercises in a cyber range. In *Proc. of the 47th IEEE Frontiers in Education Conference (FIE'17), Indianapolis, Indiana, USA*, pages 1–8. IEEE, October 2017.

[9] B. Ferguson, A. Tall, and D. Olsen. National cyber range overview. In *Proc. of the 34th IEEE Military Communications Conference (MILCOM'14), Baltimore, Maryland, USA*, pages 123–128. IEEE, October 2014.

[10] M. Leitner, M. Frank, W. Hotwagner, G. Langner, O. Maurhart, T. Pahi, L. Reuter, F. Skopik, P. Smith, and M. Warum. Ait cyber range: Flexible cyber security environment for exercises, training and research. In *Proc. of the 1st European Interdisciplinary Cybersecurity Conference (EICC'20), Rennes, France*, pages 1–6. ACM, November 2020.

[11] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, and M. A. Ferrag. Cyber ranges and testbeds for education, training, and research. *Applied Sciences*, 11(4):1809–1831, February 2021.

[12] A. Brilingaitė, L. Bukauskas, and E. Kutka. Development of an educational platform for cyber defence training. In *Proc. of the 16th European Conference on Cyber Warfare and Security (ECCWS'17), Dublin, Ireland*, pages 73–81. Academic Conferences International Limited, June 2017.

[13] T. Benzel. The science of cyber security experimentation: The deter project. In *Proc. of the 27th Annual Computer Security Applications Conference (ACSAC'11), Orlando, Florida, USA*, pages 137–148. ACM Press, 2011.

[14] J. Vykopal., R. Oslejsek., P. Celeda., M. Vizvary., and D. Tovarnak. Kypo cyber range: Design and use cases. In *Proc. of the 12th International Conference on Software Technologies (ICSOFT'17), Madrid, Spain*, pages 310–321. SciTePress, July 2017.

[15] M. Karjalainen and T. Kokkonen. Comprehensive cyber arena; the next generation cyber range. In *Proc. of the 4th IEEE European Symposium on Security and Privacy Workshops (EuroS&PW'20), Genoa, Italy*, pages 11–16. IEEE, September 2020.

[16] H. Holm, M. Karresand, A. Vidström, and E. Westring. A survey of industrial control system testbeds. In *Proc. of the 20th Nordic Conference on Secure IT Systems (NordSec'15), Stockholm, Sweden*, volume 9417 of *Lecture Notes in Computer Science*, pages 11–26. Springer, Cham, 2015.

[17] M. Conti, D. Donadel, and F. Turrin. A survey on industrial control system testbeds and datasets for security research. *IEEE Communications Surveys & Tutorials*, 23(4), July 2021.

[18] ENISA. The 2015 report on national and international cyber security exercises. Technical Report b09c680b-a2f7-11e5-b528-01aa75ed71a1, European Union Agency for Network and Information Security, 2015.

[19] G. Vigna, K. Borgolte, J. Corbetta, A. Doupé, Y. Fratantonio, L. Invernizzi, D. Kirat, and Y. Shoshitaishvili. Ten years of ictf: The good, the bad, and the ugly. In *Proc. of the 1st USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE'14), San Diego, California, USA*, pages 1–7. USENIX Association, August 2014.

[20] S. Kucek and M. Leitner. An empirical survey of functions and configurations of open-source capture the flag (ctf) environments. *Journal of Network and Computer Applications*, 151:1–19, February 2020.

[21] J. Kim, Y. Maeng, and M. Jang. Becoming invisible hands of national live-fire attack-defense cyber exercise. In *Proc. of the 3rd IEEE European Symposium on Security and Privacy Workshops (EuroS&PW'19), Stockholm, Sweden*, pages 77–84. IEEE, June 2019.

[22] M. Andreolini, V. G. Colacino, M. Colajanni, and M. Marchetti. A framework for the evaluation of trainee performance in cyber range exercises. *Mobile Networks and Applications*, 25(1):236–247, February 2020.

[23] T. Denning, A. Lerner, A. Shostack, and T. Kohno. Control-alt-hack: the design and evaluation of a card game for computer security awareness and education. In *Proc. of the 20th ACM SIGSAC conference on Computer & communications security (CCS'13), Berlin, Germany*, pages 915–928. ACM Press, November 2013.

[24] S. Frey, A. Rashid, P. Anthonysamy, M. Pinto-Albuquerque, and S. A. Naqvi. The good, the bad and the ugly: A study of security decisions in a cyber-physical systems game. *IEEE Transactions on Software Engineering*, 45(5):521–536, May 2019.

[25] M. Leitner, T. Pahi, and F. Skopik. *Situational Awareness for Strategic Decision Making on a National Level*. CRC Press, 2017.

[26] R. Langner. Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security Privacy*, 9(3):49–51, May 2011.

[27] D. Allison, P. Smith, K. McLaughlin, F. Zhang, J. Coble, and R. Busquim. Plc-based cyber-attack detection: A last line of defence. In *Proc. of the 1st IAEA International Conference on Nuclear Security: Sustaining and Strengthening Efforts (ICONS'20), Vienna, Austria*, pages 1–10. IAEA, February 2020.

[28] M. Findrik, I. Friedberg, E. Piatkowska, P. Smith, and J.-G. Song. Trustworthy computer security incident response for nuclear facilities. In *Proc. of the 4th International Symposium on Future Instrumentation and Control for Nuclear Power Plants (ISOFIC'17), Gyeongju, South Korea*, pages 1–10. KNS, November 2017.

[29] R. Busquim e Silva, K. Shirvan, J. Piqueira, and R. Marques. Development of the asherah nuclear power plant simulator for cyber security assessment. In *Proc. of the 1st IAEA International Conference On Nuclear Security: Sustaining and Strengthening Efforts (ICONS'20), Vienna, Austria*, pages 1–10. IAEA, February 2020.

[30] F. Skopik. *Collaborative cyber threat intelligence: detecting and responding to advanced cyber attacks at the national level*. CRC Press, 2017.

[31] C. Thomas, V. Sharma, and N. Balakrishnan. Usefulness of darpa dataset for intrusion detection system evaluation. In *Proc. of the 2008 SPIE Defense and Security Symposium (SDASS'08), Orlando, Florida, United States*, pages 164–171. SPIE, March 2008.

[32] M. Wurzenberger, F. Skopik, G. Settanni, and W. Scherrer. Complex log file synthesis for rapid sandbox-benchmarking of security- and computer network analysis tools. *Information Systems*, 60(C):13–33, August 2016.

[33] F. Skopik, G. Settanni, R. Fiedler, and I. Friedberg. Semi-synthetic data set generation for security software evaluation. In *Proc. of the 12th Annual International Conference on Privacy, Security and Trust (PST'14), Toronto, ON, Canada*, pages 156–163. IEEE, July 2014.

[34] M. Landauer, F. Skopik, M. Wurzenberger, W. Hotwagner, and A. Rauber. Have it your way: Generating customized log datasets with a model-driven simulation testbed. *IEEE Transactions on Reliability*, 70(1):402–415, March 2021.

[35] G. Schneider, M. Wendl, S. Kucek, and M. Leitner. A training concept based on a digital twin for a wafer transportation system. In *Proc. of the 23rd IEEE Conference on Business Informatics (CBI'21), Bolzano, Italy*, pages 20–28. IEEE, September 2021.

[36] P. Palensky, A. van der Meer, C. Lopez, A. Joseph, and K. Pan. Applied cosimulation of intelligent power systems: Implementing hybrid simulators for complex power systems. *IEEE Industrial Electronics Magazine*, 11(2):6–21, June 2017.

Author Biography

Maria Leitner is Professor of Computer Science in the Workflow Systems and Technology research group at the Faculty of Computer Science, University of Vienna and Scientist at AIT Austrian Institute of Technology. Her research interests focus on secure information systems, cyber security exercises and business process automation. Before, she was topic lead of the AIT Cyber Range and has designed and implemented strategic and technical training courses and cyber security exercises with altogether more than 350 participants. She is a member of IEEE and ACM, has edited 2 books and published more than 35 peer-reviewed articles and conference papers.

Maximilian Frank is a research engineer at the AIT. His current focus at AIT is development of cyber-attack simulations for research and training purposes on the AIT Cyber Range using state of the art virtualization and provisioning technologies. He is also part of the AIT penetration testing team and focuses on penetration testing of Web-Applications. In the summer of 2021 he finished his masters on Software Engineering and Internet Computing with a focus in IT-security at the Technical University of Vienna. In his free time, he is interested in IT security topics and participates in Capture the Flag events. Most notably, he participated in the European Cyber-Security Challenge two times, reaching the Austrian national finals both times (2015, 2016) and also reaching the European final as a member of the Austria Team in 2016.

Gregor Langner has graduated in business informatics and since 2017 he works in the sector of cyber security. He is presently working as a Research Engineer in the field of cyber security at the AIT Austrian Institute of Technology. Especially he is involved in the development and realization of cyber exercises with the cyber range. He has obtained experiences in national, international and EC-funded projects in numerous areas, where he managed critical activities in key positions. One of his strengths is the analysis and modelling of complex network infrastructures, associated information flows in order to produce and evaluate data exchange models. Beyond his research background, he is a lecturer on a theoretic and practical level for cyber security awareness, especially using cyber ranges as educational method. In addition, he provides his knowledge in the field of digital forensics and scenario building.

Max Landauer joined the Austrian Institute of Technology in 2017, where he carried out his Master's Thesis on clustering and time-series analysis of system log data. He started his PhD studies as a cooperative project between the Vienna University of Technology and the Austrian Institute of Technology in 2018. For his dissertation, Max is working on an automatic threat intelligence mining approach that extracts actionable CTI from raw log data. He is currently employed as a Scientist in the center for Digital Safety and Security at the Austrian Institute of Technology. His main research interests are log data analysis, anomaly detection, and cyber threat intelligence.

Florian Skopik is head of the cybersecurity research program at the Austrian Institute of Technology. He coordinates national and large-scale international research projects, as well as the overall research direction of his team. His main interests are centered on critical infrastructure protection, intrusion detection, and national cybersecurity. Florian received a Ph.D. in computer science from Vienna University of Technology. He is a member of various conference program committees, editorial boards, and standardization groups, such as ETSI TC Cyber, IFIP TC11 WG1, and OASIS CTI. He frequently serves as a reviewer for numerous high-profile journals, including Elsevier's Computers & Security and ACM Computing Surveys. He is a Senior Member of IEEE.

Paul Smith is a Senior Scientist with the Center for Digital Safety and Security at AIT Austrian Institute of Technology and a Visiting Researcher at Lancaster University, UK. He received his PhD in Computing from Lancaster University in September 2003. Paul's research is targeted at developing applied solutions to ensuring the security and resilience of critical information infrastructures, including computer networks and digitalized energy systems. Solutions include approaches to risk management, anomaly detection, secure architecture specification, incident response, and resilience measurement. For each of these solutions, his interest lies in addressing the cyber-physical aspects of security and resilience. He has participated in several international research projects in this area and has published articles on various aspects that relate to his interests. Paul is a member of the ACM.

Benjamin Akhras completed his Bachelor's degree in Information and Communication Systems in 2017. Since 2013 he is working as an DevOps Enigeer and Site Reliability Engineer within the public and private sector. Between 2016 and 2021 he designed and implemented the architecture for various mission networks for the Austrian Armed Forces. Since 2021 he has been working at the AIT Center for Safety & Security as Research Engineer in the area of Cyber Range architecture.

Wolfgang Hotwagner graduated with a bachelor's degree in Information and Communication Systems in 2017. From 2006 to 2019, he worked as a Linux system administrator, gaining experience with automation and blue teaming, among other things. In his spare time, he focuses on IT security topics. Since 2019 he works as a Research Engineer at the AIT Center for Digital Safety & Security where he focuses on pentesting, Cyberrange and anomaly detection.

Stela Kucek is a freelancer situated in Vienna, Austria. Kucek earned her bachelor's degree in Business Informatics (computer science with business studies) from University of Vienna in 2018 and is currently studying for a master's degree in Computer Science at the University of Vienna. From 2018 to 2021 she was a freelancer at AIT Austrian Institute of Technology, Center for Digital Safety & Security in Vienna, Austria. Her interests lie in software development, cyber security and artificial intelligence.

Timea Pahi is a scientist in the Center for Digital Safety & Security of the AIT Austrian Institute of Technology. She is working currently as a researcher at the Austrian Institute of Technology on several research projects focusing on threat intelligence, cyber attribution and on establishing cyber range trainings. Her area of expertise includes national cyber security, the protection of critical infrastructures, building cyber range scenarios and processes for cyber situational awareness and requested tool solutions for the national partners.

Lenhard Reuter is a Research Engineer at the AIT Austrian Institute of Technology. In 2021, Mr. Reuter completed his master's degree in IT Security at FH Campus Wien, Austria. He is also a certified data protection officer by Austrian Standards plus GmbH. Since 2019, he has been a member of the CyberRange team at AIT. His special areas of research include anomaly detection in industrial control systems.

Manuel Warum is a Research Engineer at AIT Austrian Institute of Technology GmbH. He has attained a Master degree in Computer Science at the Alpe-Adria-University of Klagenfurt and has over a decade of experience in developing web applications as well as several years of experience in cyber security research.