IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Date of publication xxxx 00, 0000, date of current version March 28, 2024.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

Design and Evaluation of Advanced Persistent Threat Scenarios for Cyber Ranges

TORE BIERWIRTH[1 2], STEFAN PFÜTZNER[1], MATTHIAS SCHOPP[1], CHRISTOPH STEININGER[1 2]
[1]Research Institute CODE, University of the Bundeswehr Munich, 85579 Neubiberg, Germany
[2]Department of Computer Science, University of the Bundeswehr Munich, 85579 Neubiberg, Germany

Corresponding author: Tore Bierwirth (tore.bierwirth@unibw.de)

The authors acknowledge the financial support by the Federal Ministry of Education and Research of Germany in the programme of "Souverän. Digital. Vernetzt.". Joint project 6G-life, project identification number: 16KISK002

ABSTRACT Both criminals and state actors are using the cyberspace to pursue their interests, including obtaining information, sabotaging networks, and disseminating disinformation. Advanced Persistent Threats (APTs) are state and non-state threat actors with high levels of expertise, target knowledge, and available financial and material resources. To effectively counter APT campaigns, it is necessary to have a deep understanding of the methods used by threat actors. Cyber Ranges provide a realistic training environment to develop and train the skills needed to respond to future attacks. However, this requires the ability to simulate APT attacks in a Cyber Range in an automated manner. This article presents an approach to implementing APT scenarios in fully virtualized Cyber Ranges. To achieve this, we extended a theoretical model to enable the formalized representation of APT attacks. Based on this model, we developed a concept for the technical implementation resulting in a framework for an automated simulation of APT attacks in Cyber Ranges. We successfully evaluated both by formalizing two different real-world APT scenarios and implementing an abstract one.

INDEX TERMS Advance Persistent Threat, Cyber Range, Diamond Model, TTPs, information security, security training.

I. INTRODUCTION

In cyberspace, the boundaries between state and non-state conflicts become blurred, both spatially and temporally. Attacks can occur simultaneously in multiple locations. They are repeatable and can be carried out with little resources and prior knowledge. Due to appropriate anonymization precautions, they are furthermore difficult to attribute. By 2020, cybercrime had already caused \$1000 billion in global damage, which is about 1% of the world's total economic output [1].

Advanced Persistent Threats (APTs) are using sophisticated methodologies and highly advanced attack tools. They use e.g. zero-day exploits to gain access to systems with publicly unknown vulnerabilities [2]–[4]. APT attacks are characterized by a large time frame of execution, often spanning several months, during which target systems are repeatedly attacked. After gaining initial access to the targeted system, the attackers try to extend their access to other systems in the network over a long period of time. To avoid detection, they take technical measures to minimize their tracks. APT measures are often only discovered several months after the initial intrusion into the compromised network. For example, in certain instances of the Shady RAT APT campaign [5], the attacks were not discovered until two years after the initial infection [6].

To effectively address the threats posed by APT campaigns, it is crucial to have a comprehensive understanding of their operations and functions. Cyber Ranges can be used to build and expand this understanding. They enable the implementation of exercise scenarios in a simulated environment in order to train and learn how to deal with attacks.

To implement APTs in Cyber Ranges, it is necessary to be able to formalize them in a technically detailed manner, such as by using unstructured reports. Existing models, such as the Cyber Kill Chain (CKC) from Lockheed Martin [7], or its extensions [8]–[11], do not enable a formalized description of APTs and cannot be used as a basis for technical implementations. The Diamond Model [12] enables the formalization of APTs. In addition, hypotheses can be used to close logical gaps in the attack process. However, the current version of the model cannot be used as a basis for implementations in Cyber Ranges due to the lack of possibility to describe technical details. The MITRE ATT&CK® framework [13], can be used for this purpose, but it does not provide a formalization option.

Taking these limitations into account, this paper makes the following contributions:

- We introduce the Tactics, Techniques, Procedures Diamond Model (T2P-DM), which extends the existing Diamond Model to integrate tactics, techniques, and procedures (TTPs) from the MITRE ATT&CK framework which can be used to implement Cyber Range scenarios.
- With T2P-DM we enable the formalized description of APT attacks while closing logical gaps.
- We developed a Framework for an Automated Simulation of APT attacks in Cyber Ranges (FASAC).
- We evaluated T2P-DM and FASAC using concrete case studies in order to verify the suitability of the model and its implementation.

The remainder of the paper is organized as follows: In Section II the necessary foundations are elaborated. Section III presents the related work. Section IV describes our concept, which comprises of a theoretical model and a technical framework. Section V evaluates the concept using selected scenarios. Finally, Section VI concludes the paper with a brief summary and outlook.

II. FUNDAMENTALS

In this section, basic terms relevant for the concept to be developed and its underlying technology are introduced.

A. ADVANCED PERSISTENT THREATS

In 2011, computer scientists at Lockheed Martin published a white paper documenting a class of adversaries that attack enterprise networks and named them Advanced Persistent Threats (APTs). These groups are highly trained and educated in information and system security, with access to vast resources, both financial and technological. [7]

Thus, both state and non-state cyber actors, that possess a high level of expertise, knowledge about their target, and ample financial and material resources are commonly referred to as APTs [14]. APTs use sophisticated methodologies and highly advanced attack tools, including zero-day exploits, to gain access to systems [2]–[4]. Those APT attacks are characterized by a large time horizon, often targeting systems over a period of months. Once initial access is gained, attackers attempt to expand their access to other systems in the network over an extended period of time. The cyber actors use technical measures to minimize their traces, making it more difficult to detect them.

B. CYBER RANGES

The term Cyber Range refers to an interactive training environment that educates and trains users in cybersecurity issues, including recognizing anomalies and attacks in networks, initiating countermeasures, and handling IT security tools [15]. Cyber Ranges are also used to test the functionality and effectiveness of IT security products, conduct business impact simulations and review IT security concepts [15].

C. CYBER RANGE SCENARIOS

A Cyber Range scenario provides users of a Cyber Range with a narrative that embeds the tasks to be performed and the goals to be achieved. It defines the domain in which the scenario is executed, such as security awareness or war games in a military environment, and offers learning concepts to train the users' skills. Gamification elements are used to increase motivation and thus learning success. Scenarios can be designed as static or dynamic. Static scenarios follow a single and fixed goal while dynamic ones evolve with each action a user takes. [16]

To design a scenario the creation of attack trees is crucial [17]. Attack trees simulate a hypothetical or real cyber attack and thus show which attack path an attacker uses to gain access to the IT systems of its victims.

III. RELATED WORK

This chapter reviews theoretical models for describing APT attacks and examines existing technical implementations that enable automated simulation of these attacks.

A. MODELS TO DESCRIBE APT ATTACKS

There are various models that illustrate the lifecycle of a cyber attack. These models divide APT attacks into phases that an attacker must pass through in order to achieve its goal.

1) The Cyber Kill Chain by Lockheed Martin

In 2010, security researchers from Lockheed Martin investigated APTs [7]. As a result, they developed the Intrusion Kill Chain, which is commonly referred to as the Cyber Kill Chain (CKC). The researchers identified seven phases that an APT must pass through to successfully execute a cyber attack: reconnaissance, weaponization, delivery, exploitation, installation, command and control (C2), and actions on objectives. The Lockheed Martin model focuses on malware, which is a main criticism of the model. It does not address alternative attack vectors, such as threats from internal attackers or the exploitation of remote access. Although the initial compromise of the target system requires only a part of the attacker's time and technical resources, the CKC describes this process in six of the seven phases. The description of the phase actions on objectives is too general and does not adequately reflect the necessary technical measures and time frame for the compromise of the target. Although the amount of time spent in compromised target systems has decreased steadily in recent years, cyber actors still remain undetected for an average of almost a month [18]. During this period, attackers gain further access to the target complex. In cyber attacks, lateral movement techniques are used in almost 70% of cases to gain access to relevant information and achieve defined goals. [19].

2) Adaptations of the Cyber Kill Chain

In recent years, there have been various proposals to adapt Lockheed Martin's CKC to the evolving threat situation and the tactical-operational approach of APTs. After analyzing

22 APT campaigns, Ussath *et al.* [8] propose three relevant phases for describing a cyber attack, consisting of initial compromise, lateral movement, and C2. They assign generalizing techniques and methods of APTs to each of these three stages without applying a high level of detail. Zhang *et al.* [9] propose a four-stage model for the automated creation of APT attack sequences based on log entries from intrusion detection systems. They extend the aforementioned three-stage model by adding an information collection phase. In a comparative study of APT campaigns, Chen *et al.* present a six-stage CKC [10]. In contrast to Lockheed Martin's CKC, they combine the steps of reconnaissance and weaponization. A report published by the IT security service provider Mandiant in 2013 [11], analyzed the activities of the cyber actor known as APT1. The authors of the study developed an attack lifecycle to describe the general approach of APTs based on techniques used by the group. The phases of initial compromise and establishing foothold summarize the first six stages of Lockheed Martin's CKC. The authors define privilege escalation as a dedicated phase in which the attacker gains higher-level access to the target system. During the internal reconnaissance phase, the attacker gathers information about the network to enable lateral movement. The maintain presence phase involves ensuring continued access to the target's most important systems over an extended period of time. This is followed by completing the mission, which is the final stage of the APT attack. Similar to the previously presented models, this phase involves the collection and exfiltration of sensitive data and information. The phases escalating privileges, internal recon, move laterally, and maintain presence are iterative steps that an APT executes multiple times. The role of the initial recon phase in this model is unclear. It is not listed in the general description of the attack lifecycle of the study and remains unmentioned in the text. It can be assumed that the reconnaissance of the target should not be depicted as a separate step.

The presented models primarily extend the original CKC to include the necessary steps an attacker must take within a target to achieve his mission objectives. The models differ primarily in the granularity of the description of an APT attack and in the names and sequences of the individual phases. Their common goal is to describe cyber attacks in a general and structured manner.

3) The MITRE ATT&CK® Framework

ATT&CK [13], which was initiated by the MITRE organization, is a catalog with tactics used by attackers, each of which is linked to the associated techniques and procedures (TTPs). It is important to note that ATT&CK is not a sequential model of a cyber attack that must be followed step-by-step. Rather, it offers a detailed knowledge base that was created through analyses of real APT attacks and is continuously updated. Tactics represent the highest level of abstraction in the ATT&CK framework. It describes the reasons behind a cyber actor's use of a specific technique in its attack. The techniques form the core of the ATT&CK framework and represent the lowest level of abstraction within the model. Each technique is described in detail within the framework, often including code examples and countermeasures. Additionally, the framework links analyses of APT attacks. It focuses on aggregating the techniques used in APT attacks. Therefore, it is suitable not only for planning and implementing concepts to defend against threats from APT attacks but also for creating Cyber Range scenarios.

4) The Unified Kill Chain

In a paper from 2017, Paul Pols proposes the Unified Kill Chain (UKC) [20] as an attempt to unify various lifecycle models. Pols examines existing approaches in a literature study and uses them to design an 18-stage model.

Compared to the ATT&CK framework, the UKC is more comprehensive and includes social engineering and pivoting as separate phases. The target manipulation phase covers a wider range of attacker activities, including denial of service or targeted file deletion. Additionally, the descriptions of each phase are tailored to provide a comprehensive explanation of the techniques used by attackers. The order of tactics listed in the ATT&CK model is based on the phases of the lifecycle of a cyber attack. However, the framework itself does not specify a defined order. The UKC defines the phases sequentially. In a white paper published by Pols, the 18 phases are assigned to three overarching intermediate objectives of an attack: initial foothold, network propagation, and actions on objectives [21]. The number and order of the phases can differ depending on the type of attack. Pols provides a tailored UKC for different attack paths in his work [20].

5) The Diamond Model

In 2013, Caltagirone *et al.* introduced the Diamond Model to describe cyber attacks [12]. The goal was to identify the fundamental components of such attacks and to develop a formal method for analyzing them. Compared to other models for explaining APT attacks, the Diamond Model offers the most comprehensive description. It combines the approaches of CKC models and provides a structure for mapping and describing cyber measures in detail for individual attack events. It assists in analyzing cyber attacks and provides a technical description as a template for simulating scenarios in a virtualized environment. The model consists of several components, which are described in the following.

Diamond Event: The Diamond Event is the central component of the model, consisting of both core and meta properties. According to Caltagirone *et al.*, any offensive cyber measure involves one or more attackers who utilize their capabilities against the victim through an infrastructure to achieve the intended goal. This results in the core elements of a Diamond Event: adversary, capability, infrastructure, and victim. Additionally, meta properties such as timestamp, phase, result, direction, methodology, and resources are introduced to further characterize the event. Fig. 1 shows the core properties of the model as nodes whose relationships to each other are

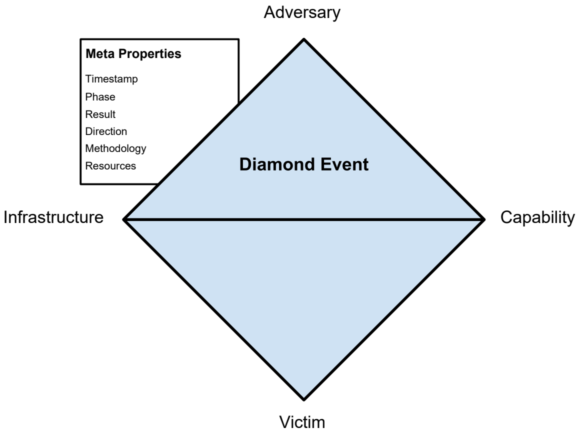

FIGURE 1. **Core and meta properties of a Diamond Event [12]**

represented by edges.

Core Properties: The authors assign core characteristics to an attack event. These are adversary, capabilities, infrastructure, and victim.

Adversaries are insiders, outsiders, individuals, groups, and organizations that attempt to compromise computer systems. According to Caltagirone *et al.*, the adversary is usually unknown at the time of discovery, so the property often remains empty.

Capabilities describe the technical tools and methods used by the adversary in an event. Capacity as a capability is recorded as a sub-property. It describes all possible weaknesses and vulnerabilities that can be exploited by the adversary's capabilities.

Infrastructure describes the physical and logical communication structures that enable the adversary to use its capabilities against the victim. After the initial penetration of a computer system or network, the adversary uses C2 structures to apply its capabilities in a controlled and persistent manner.

The victim is the target of the adversary. Vulnerabilities are exploited by the adversary's capabilities through an infrastructure against the victim. Caltagirone *et al.* propose to distinguish between a victim as a person, called victim personae, and the victim in the form of an asset, called victim asset. Typically, individuals and organizations are captured as victim personae. When analyzing an APT attack, a victim asset within a Diamond Event is to be understood as a technical feature or component.

The authors postulate that each system, and thus each victim asset, has vulnerabilities and weaknesses. Vulnerabilities and weaknesses are described as a sub-property of the core property victim, regardless of whether it is a victim personae or a victim asset.

Meta Properties: In addition to the core properties, a Diamond Event is further specified by additional meta properties. The meta properties of an attack event can be freely chosen and designed. The authors suggest six predefined properties: phase, timestamp, result, direction, methodology, and resources.

The phases are executed in sequence to achieve the desired result for the attacker. Caltagirone *et al.* state that any malicious action must consist of at least two phases to be successful. One phase is the selection of the target. The other is the deployment of a capability by the adversary against the victim. The authors formalize the phases P as tuples with n elements, where n describes the number of defined attack phases:

$$P = \langle p_1, p_2, \ldots p_n \rangle \tag{1}$$

The following applies:

- p is a phase in a series of actions by the adversary.
- $n \geq 2$, which means that each attack consists of at least two executed attack phases.
- p_1 is the first phase of an attack.
- Phase p_{n+1} is performed in the following of phase p_n.

In principle, the attack sequence models presented in Section III-A can be considered when choosing an appropriate phase division. Only one phase can be assigned to each Diamond Event. As the phases of an attack must be performed sequentially, a fine-grained division of the phases can provide valuable insights into the analysis process of a cyber measure. Comparing different APT attacks is only possible if the phase partitioning allows the variance in the collected data required for the analysis. For example, a kill chain model with the three phases initial compromise, lateral movement, and C2 can only provide three data points for phase analysis.

Timestamp documentation allows to record the start and end of a Diamond Event. This makes it possible to examine how long it takes an adversary to execute individual events. By grouping related events according to their associated phases, it is possible to determine which individual phases are the most time-consuming from the adversary's perspective. The cyber actor then does not go through the corresponding phases, or only with a delay.

The Diamond Event has an additional meta property known as the result. The authors note that it is not always possible to determine the result. The property can be classified as success, failure, or unknown. Another suggestion is to classify the property based on its impact on the three IT protection goals confidentiality, integrity and availability: confidentiality compromised, integrity compromised, and availability compromised.

A direction is a meta property of an event within an APT attack. The property takes on seven possible values: adversary-to-infrastructure, infrastructure-to-adversary, victim-to-infrastructure, infrastructure-to-victim, infrastructure-to-infrastructure, bi-directional, and unknown.

The meta property methodology allows an attack event to be categorized based on a typified view of the technical method used. Caltagirone *et al.* propose to record the methodology directly as a meta property of an event.

The meta property resource is used to capture all types of tangible and intangible resources and states that an adversary needs to successfully execute an event.

Activity Threads: Caltagirone *et al.* define activity threads as a directed graph AT ordered by phases, in which Diamond Events are represented as nodes. Their relationship to each other is described by directed edges. The authors formally define an activity thread as $AT = (V, A)$, whereby:

- AT is a finite graph.
- $|V| \geq 1$, there is at least one event in the thread.
- V is the set of all events partitioned into subsets such that they have the same adversary-victim pair. Each subset is divided into phases $P = \langle p_1, p_2, ...p_n \rangle$, where each event is assigned to a phase p.
- A is a set of directed edges. A directed edge (x, y) is defined if the adversary could successfully execute an event y only because the event x directly preceded the event y.
- Several directed edges can lead to one event.
- Only one path can exist between two events.
- Three additional descriptive values are added to each edge:
  - The analytical confidence denotes the analytical certainty with which a connection between two events exists.
  - Each edge can exist actually or hypothetically. Hypothetical connections are represented by dashed edges.
  - The label AND/OR is intended to make clear whether an edge was necessary for a subsequent event (AND) or whether the event could also have occurred through another potential path (OR).

The authors of the Diamond Model use Lockheed Martin's CKC as a phase classification. A thread aligned with the phases of a cyber attack in this way represents the totality of all actions performed by the adversaries against a given victim. At this point, a distinction must be made between the concept of a victim as a core property of a Diamond Event and a victim in the context of an activity thread. When considering an activity thread, the term victim is typically understood as a victim personae and is intended to describe the overarching attack target of the cyber measure. The frame of reference for the core property victim is the individual Diamond Event. In addition to victim personae, technical systems and features are also assigned to the victim concept.

Diamond Events and their connections to each other are described as actual or hypothetical events. At the beginning of the analysis of a cyber attack, the derivation of hypothetical events and their integration into an activity thread can help to improve the knowledge about the course of an attack. They allow the analysis process to search for information and indications to confirm or refute the existence of such an event.

Individual activity threads can be horizontally connected. This is always the case when the adversary uses its capabilities and infrastructures against a new victim, using the results and resources of the Diamond Events from the original activity thread. Fig. 2 visualizes the activity threads for a hypothetical cyber attack. Hypothetical Diamond Events are marked as dashed nodes.

6) Summary of the described models

Section III-A1 describes the CKC, which serves as an initial procedural model for describing APTs and provides the foundation upon which other models are built. The CKC provides an initial breakdown of the phases of an APT and enables a basic understanding of its progress. Subsequent models, such as the one developed by Mandiant and discussed in Section III-A2, extend the CKC by introducing a lifecycle perspective. This allows for iterations and, equally important, a more detailed representation of the effects of lateral movement within an attack. In contrast, the MITRE ATT&CK framework described in Section III-A3 does not serve as a procedural model, but rather provides a way to explicitly map the actions performed on a system to the higher-level attack phases. The UKC model presented in Section III-A4 consolidates the various procedural models and provides a phase classification that is both universally applicable and granular. This granularity enables a comprehensive technical implementation of the APT attack steps. However, the UKC model is purely sequential and cannot incorporate repetitive processes. The Diamond Model, described in Section III-A5, increases the granularity to the level required to describe the technical implementation of virtualized infrastructures. This is critical for accurately representing APTs in such environments. The model is characterized by its mapping of actions and parallel execution of processes, as well as its handling of repetitive steps. However, it faces challenges in representing the temporal context of APTs and the required granularity of detailed technical descriptions.

In summary, there is a need to integrate the strengths of these models into a unified approach to adequately capture the evolving nature of today's APTs. This new model should emphasize the link between high-level phases and the actions they contain, as well as consider parallelism and repetition in APT operations.

B. TECHNICAL IMPLEMENTATION OF APT SCENARIOS IN VIRTUALIZED ENVIRONMENTS

There are a number of different technical platforms available for simulating cyber attacks. These are not limited to Cyber Ranges, but also include software frameworks that enable the simulation of APT attacks. Due to the large number of existing solutions, only those that can perform an automated simulation of a cyber attack are considered.

1) CRATE

In [22], Gustafsson *et al.* give a technical insight into the Cyber Range and Training Environment (CRATE) of the

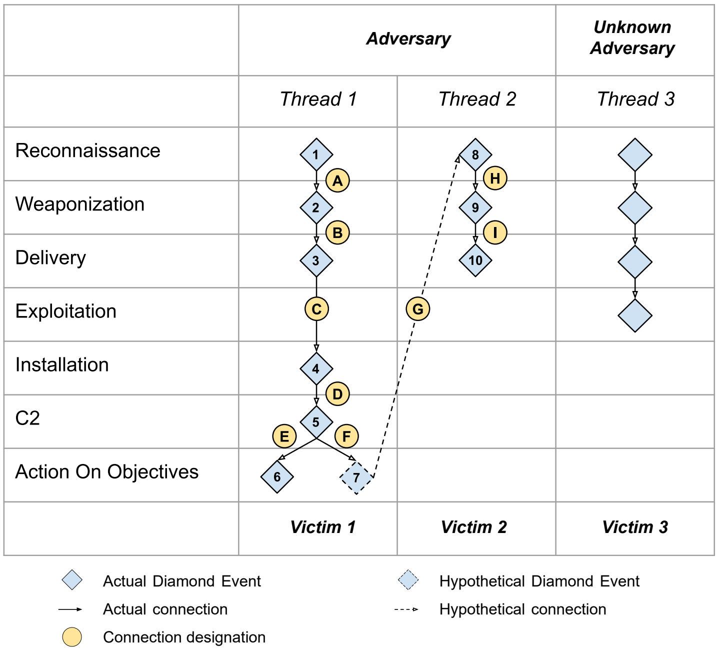

FIGURE 2. Activity threads for a hypothetical cyber attack [12]

Swedish Defence Research Institute. The architecture of CRATE consists of three main components: the virtualization, the control plane, and the event plane. The control plane contains all the components needed to set up and control the environment and the scenarios running in the Cyber Range. This includes a central API server, a VM repository, the CRATE Exercise Control (CEC) for creating and managing training scenarios, and the Scanning, Vulnerabilities, Exploit, and Detection (SVED) tool for automating attacks. With CRATE, it is possible to deploy cross-platform environments with virtual and physical hardware.

CRATE can perform automated attacks and simulate user behavior. It uses SVED, which consists of five core functionalities. The Threat Intelligence module collects vulnerabilities from various public sources. The Designer is used to create attack graphs via a graphical user interface, which the Executer executes via the Attacker/Sensor Agents in the environments. The Logger module stores all log data generated during the execution of the attack.

The source code of CRATE has not been released. Therefore, it cannot be conclusively determined whether automated scenarios are implemented in an event-driven manner or whether fixed temporal sequences of attack events are implemented.

2) Splunk Attack Range v2.0

The Splunk Attack Range in version 2.0 [23] is a framework for automatically creating and attacking virtual environments using Ansible and Terraform. The architecture of the framework does not allow centralized control of the environment. The setup of the necessary components and the launch of the attack scenarios are initiated and executed on the local systems.

The Atomic Red Team (ART) function library [24] and the open source tool PurpleSharp [25] are used to execute the attacks. ART is a collection of function calls that map the TTPs of the MITRE ATT&CK framework. It is not possible to link individual techniques to simulate a complex APT scenario.

To map more complex attack chains, the Splunk Attack Range integrates the Prelude Operator [26]. To simulate APT attacks, the Prelude Agent is installed on the system to be tested. Pre-built chains [27] can be sent to the agent, which then executes them sequentially on the test system. Since Prelude Operator is not publicly available, no definitive statement can be made about its capabilities.

3) Metta

Metta [28] follows a similar approach to ART. Virtual machines running any operating system can be started using the Vagrant API. Using a YAML syntax, actions are created based on predefined ATT&CK TTPs. Multiple actions can be combined into a single scenario. Python and Redis are used to create the virtual machines of the environment and to execute the actions and scenarios. A visual editor for developing scenarios is not available.

Metta also executes attack steps sequentially. User actions cannot be simulated. The time factor in a cyber attack can only be simulated to a limited extent. Individual actions can be supplemented with sleep commands, but complex time patterns cannot be reproduced.

4) Infection Monkey

Infection Monkey [29] is an open-source framework that can simulate a cyber attack on an existing infrastructure. Monkey agents are installed on the systems in the network, which communicate with the central Monkey server. The framework does not provide an integrated solution for creating virtual machines. The agent is available for Windows and the most common Linux distributions. The server executes predefined and customizable scenarios, and several TTPs of the MITRE ATT&CK framework are mapped. The web interface compiles and executes scenario actions sequentially. However, the web interface cannot create more complex scenarios that are time- and event-controlled, and where the result of an action can affect the subsequent action.

To determine which actions are carried out when Infection Monkey has spread on a system, Post Breach Actions are used. The network propagation module defines segmented network areas and tests them for proper separation. The provided modules can be extended with Python code. However, it is not possible to simulate user actions on the systems.

5) Summary of described technical implementations

Section III-B1 described CRATE, which demonstrates how a technical control framework can represent automated attack sequences in a Cyber Range. Physical hardware can also be included to represent certain attack sequences. Section III-B2 examines the Splunk Attack Range, which allows the rapid creation and management of a virtualized infrastructure through the use of automation tools such as Ansible and Terraform. This also highlights the importance of a centralized component for managing complex attack chains. Metta, outlined in Section III-B3, meanwhile provides an approach for a temporal mapping of MITRE ATT&CK's techniques. While the use of actions based on text editors and a standardized syntax allows for easy implementation, a more intuitive, graphical programming possibility of such sequences is missing here. Section III-B4 described Infection Monkey and their advantages of a centralized, web-based management component based on the client-server principle. The event-driven mechanism for post-breach actions is particularly relevant for the representation of APTs. However, there is still room for combining more complex temporal contexts.

The analysis of the technical implementations mentioned above indicated that a control framework is necessary for a technical representation model for APTs. Such a framework should be able to perform all technical operations and do so based on a temporal or event-driven context. Additionally, a component to create and orchestrate virtualized infrastructures is essential for representing APTs in Cyber Range scenarios. Compared to existing solutions, it is crucial to develop a method for representing complex temporal contexts and user actions.

IV. CONCEPTION

Out of all the models studied, the Diamond Model offers the most comprehensive description of APT attacks. It combines approaches from CKC models and provides a structure that can be used to map actions at any level of detail into individual attack events. The Diamond Model assists in analyzing an attack while providing a technical description as a template for implementing a scenario in a virtualized environment. However, the Diamond Model has limitations in capturing TTPs and their temporal relationships. This makes it unsuitable for simulating APT attacks in fully virtualized Cyber Ranges. To address this issue, the Tactics, Techniques, Procedures Diamond Model (T2P-DM) was developed as an extension of the Diamond Model. T2P-DM is presented in Section IV-A. The extended model is used to describe APT attack steps in a way that enables the implementation as a scenario in a fully virtualized Cyber Range.

None of the considered practical implementations for the automated attack simulation of APT attacks enable the combination of virtualized implementation and event-driven execution of the attack. Dynamic reactions to external events cannot be mapped, and in some cases, the simulation of user actions is not possible. As a result, they are not able to simulate relevant TTPs realistically. The Cyber Range CRATE is the only implementation capable to control the temporal sequence of an attack in a fine-grained manner. However, it cannot implement more complex temporal and event-driven attack patterns. For this purpose, we introduce FASAC, a Framework for an Automated Simulation of APT attacks in Cyber Ranges in Section IV-B.

A. THEORETICAL MODEL

The Tactics, Techniques, Procedures Diamond Model (T2P-DM) provides a structure for describing attacks and developing attack scenarios. It creates the structural conditions for the technical Framework for an Automated Simulation of APT attacks in Cyber Ranges (FASAC). The new model extends the core and meta properties of the original Diamond Model. Integrating the TTPs from the MITRE ATT&CK framework [13] provides the T2P-DM with the necessary technical details. Additionally, they allow the representation of actual events and filling logical gaps through hypothetical events.

1) Core Properties

The model's extension modifies the notions of capabilities and victim in relation to the core properties of the attack event. The core properties that are not explicitly mentioned here are used as intended by the original Diamond Model.

Capabilities: The MITRE ATT&CK framework offers a comprehensive knowledge base of techniques used by APTs to attack their victims in real-world scenarios. These techniques are well-suited for implementing attack scenarios in Cyber Ranges with a high degree of realism. Therefore, the Diamond Model is extended to assign exactly one technique from the MITRE ATT&CK framework to each attacker capability in a Diamond Event. If the corresponding ATT&CK technique is divided into sub-techniques, the relevant sub-technique will be specified.

Victim: The Diamond Model proposes distinguishing between victim personae and victim asset to separate non-technical and technical analyses regarding the term victim. In the technical implementation of an APT attack in Cyber Ranges, a victim personae needs to be considered in very few cases. Consistently using the term victim asset instead of victim also resolves the imprecision of using the term in the context of activity threads.

2) Meta Properties

The meta properties are also adapted to be usable for the APT scenario implementation. For this purpose, the phase property is adapted and it is explained why the methodology property is not considered in the process execution. The remaining meta properties are used as defined in the original model.

Phase: The original Diamond Model, as originally formulated, uses Lockheed Martin's CKC, which is inadequate for differentiating and describing the phases of an attack. In contrast, the UKC was developed by taking into account existing models and evaluating real attacks. It provides a comprehensive and widely applicable classification of attack phases, which is finely grained and adapted to technological advancements. The UKC allows a differentiated and complete technical implementation of individual attack steps when planning APT scenarios. It groups individual phases into the higher-level intermediate objectives, including initial foothold, network propagation, and action on objectives. To use the UKC in T2P-DM, the phases P of the original model are redefined as follows:

$$P = P_{IF} \cup P_{NP} \cup P_{AO} \tag{2}$$

with

$$p_i \in P \tag{3}$$

and

$$p := p_1, \dots, p_n, \tag{4}$$

whereby:

- P is the union of the disjoint sets of the intermediate phases initial foothold P_{IF}, network propagation P_{NP}, and action on objectives P_{AO}.
- P_{IF}, P_{NP} and P_{AO} are linearly ordered.
- P_{IF}, P_{NP} and P_{AO} consist of a set of phases p.
- p is a phase in a series of actions by the attacker.
- p_1 is the first phase of an attack.
- Phase p_{n+1} is performed in the following of phase p_n.
- Each attack consists of $n \geq 2$ executed attack phases.

Shown in Fig. 3, the individual phases are mapped to the intermediate objectives initial foothold, network propagation, and action on objectives, which are defined by the UKC and added to the graphical representation of activity threads.

Methodology: Mapping the techniques and sub-techniques from the MITRE ATT&CK framework to a Diamond Event enables the representation of the appropriate level of abstraction. Thus, further abstraction of the applied technique is unnecessary. Because of this, the methodology meta property is not further considered.

B. TECHNICAL FRAMEWORK

None of the implementations discussed in Section III-B enables the creation of APT scenarios in its full complexity. Accordingly, an implementation must be capable of simulating user behavior, realizing complex attack sequences, and performing time and event-driven actions. With FASAC, a framework is presented that provides the stated functionalities and implements the APTs formalized with T2P-DM. It automates the execution of attacks in a Cyber Range by using different identified components and techniques.

1) Logic Engine

The T2P-DM describes APT scenarios as a sequence of related attack events. Diamond Events are represented as nodes and their connections as edges in an activity thread. When an attack is launched, the individual Diamond Events are processed sequentially. Taking into account an event-driven architecture, the attack steps to be simulated are created in FASAC with a web-based graphical editor and are then available in an overview as predefined modules. The graphical logic editor is based on the principle of flow-based programming [30]. As shown in Fig. 4, individual modules are linked together. It is possible to use modules multiple times.

2) Container Environment

All components of FASAC and the resources that need to be simulated for the scenarios are executed exclusively in a container environment, managed by an orchestrator. Predefined templates for the necessary containers are provided by FASAC. These templates are dynamically modified using a template language and enriched with additional variables. For realistic scenarios that require IT systems and associated services that cannot be mapped in containers, the orchestration platform supports the creation, management, and termination of virtual machines (VMs). As a result, the processes required for the operation of FASAC use the same technology stack. This significantly reduces the implementation and administration effort while increasing the reusability and maintainability of program code.

C. LOGICAL STRUCTURE

The FASAC container environment is divided into a control layer and one or more scenario layers. The control layer contains all components necessary for the framework's operation. Each APT scenario that needs to be simulated is assigned to its own scenario layer. The container orchestrator

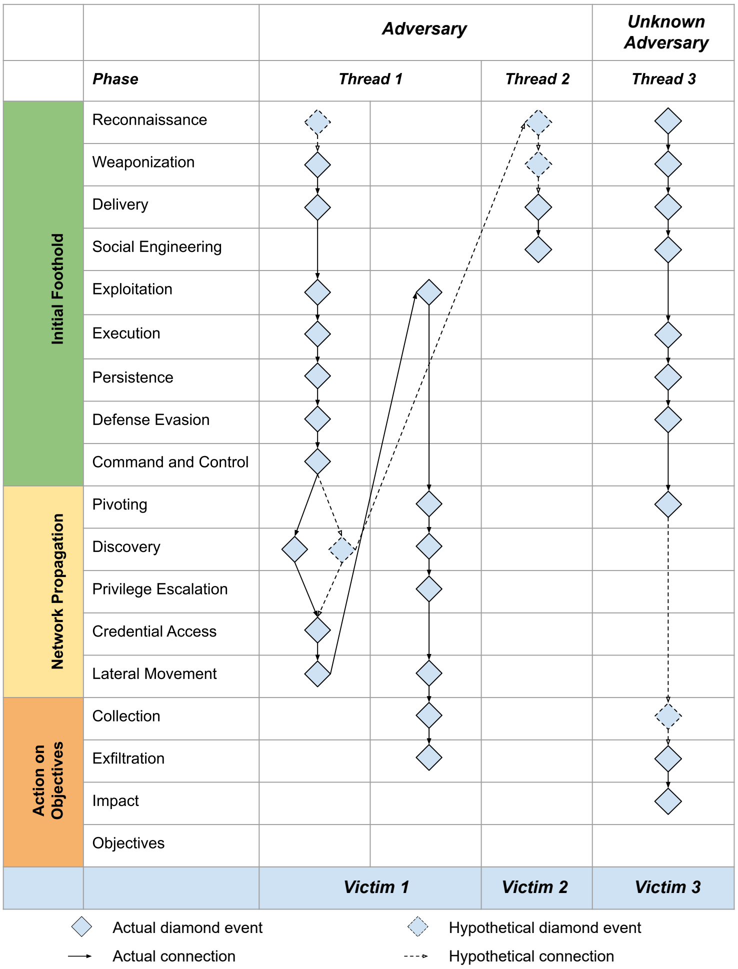

FIGURE 3. Activity Threads of the T2P-DM with the phase division and the intermediate targets of the UKC

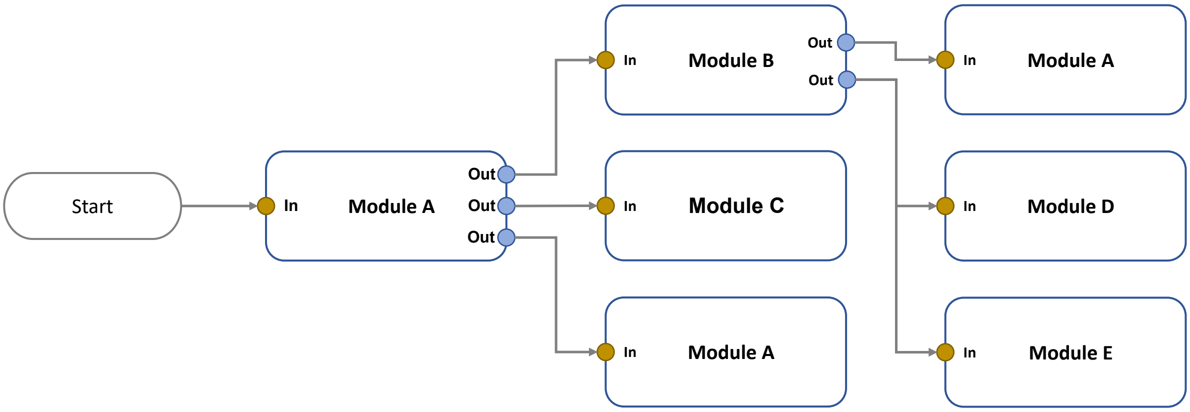

FIGURE 4. Adapted flow-based programming approach

enables logical separation between individual scenario layers, preventing undesired communication between containers of different scenarios.

1) Attack Simulation Procedure

The T2P-DM creates activity threads by analyzing APT attacks. These threads are then used to simulate scenarios. The Diamond Events contain all required information, documented in both the core and meta properties of an event. These properties consist of the adversary's capabilities and infrastructure, the victim asset, and the meta properties timestamp and resources.

Each attack event is defined by at least one TTP that is initiated and executed by the adversary. In a Diamond Event, TTPs are captured in the core property capabilities. FASAC ensures that any adversary action can be mapped as a TTP through its components and implemented application logic. If additional IT systems are required, the framework provides and releases them as containers or virtual machines on an event-driven basis. Time is an essential factor in APT attacks. Therefore, each component of the framework ensures easy and flexible implementation of the T2P-DM meta property timestamp at any point in the logic.

Fig. 5 presents a schematic overview of the general sequence of an automated APT attack simulation according to the T2P-DM. The diagram illustrates the processing of a single Diamond Event. Checks are performed for each event, first to determine the availability of the resources required to execute TTPs, and then, to identify whether an attacker or victim system is involved and if it already exists.

The following step verifies whether a container or VM is required to execute the TTP and if it is already available. If a container or VM is unavailable although required, it will be instantiated. The system will then check if a time offset is required. After the specified time has passed, the TTP will be executed, and the results of the execution will be verified and returned. Afterwards, the utilized resources are released, and the system checks if any additional Diamond Events need to be executed. If necessary, it will perform another run.

V. EVALUATION

To evaluate the T2P-DM, two real APT scenarios were analyzed and mapped to it. The first scenario is an attack by the APT group Emissary Panda (APT27), described in [31]. The analysis focuses on a specific attack to provide a fine-grained and chronological description of the attack events. The second scenario examines the APT group Earth Lusca [32] from a more general perspective. Here, no individual case was analyzed in detail, but rather it was examined whether the T2P-DM is capable of depicting the basic technical procedure of an actor. Based on a fictitious scenario, the evaluation of FASAC verifies the technical capabilities of the framework.

A. EMISSARY PANDA (APT27)

HvS Consulting divides the APT attack into three phases. The first phase, compromise and objectives, marks the start of the cyber measure. There, the Microsoft Exchange vulnerability ProxyLogon [33] is exploited, lateral movement in the network is conducted, and an initial collection of data is performed. In the second phase, persistence and stealth, Emissary Panda attempts to secure its access to the target network through further lateral movement. In response to being potentially detected, the APT group increases its efforts to gain access to relevant IT systems and to exfiltrate sensitive data during the third phase.

The individual actions are described, and the first attack phase is visually represented. The TTPs are also described in detail. On the basis of this information, Diamond Events are created with their core and meta properties defined. These attack events are then assembled into an Activity Thread. Fig. 6 shows the initial two phases of the attack.

Exemplary, some Diamond Events are explained in more detail below. Based on the report [31], the attacker group's first detected activity was a C2 communication from system EX01 (MITRE TTP: T1071.001). It is not known if the system was further exploited. Therefore, the previous Diamond Events must be inserted hypothetically. In Fig. 6 the C2 communication corresponds to the Diamond Event $T_1 7$. The report suggests that the attacker group used the ProxyLogon vulnerability in several cases. As the EX01 system was a Microsoft Exchange server, it is likely that it was exploited by using the aforementioned vulnerability [33] ($T_1 3$). This assumption based on TTPs used by the APT group and leads to the events $T_1 1$ (scan for vulnerable Exchange servers) and $T_1 2$ (selection or implementation of a suitable exploit). The report also indicates the execution of the HyperBro malware on Client01 ($T_1 10$, MITRE TTP: T1047). This results in the hypothetical event $T_1 4$. Another observed event is the establishment of persistence by installing a service ($T_1 16$, MITRE TTP: T1543.003), which leads to the event $T_1 5$.

B. APT EARTH LUSCA

The authors of the study provide a list of technical measures with their respective commands, but they do not associate them with a specific attack process. As a result, it is unclear which systems were targeted and to what extent the group implemented these measures. Additional information, such as timestamps and IP addresses, is only mentioned sporadically and mostly out of structural context. To process the present case study with the T2P-DM, we had to make necessary assumptions and concretizations:

- The analysis assumes a fictitious attack history based on the technical facts. Unless otherwise mentioned, the individual attack events refer to one and the same IT system.
- The Activity Thread of the fictitious cyber measure classifies attack events and their connections as actual, only if the technical characteristics described in the analysis can be assigned to this event. All other Diamond Events and connections are marked as hypothetical.
- This study categorizes individual technical procedures and assigns them to a specific attack phase. This categorization is maintained in a Diamond Event presentation, unless otherwise mentioned.

The study identifies three primary attack vectors used by Earth Lusca. The group uses spear-phishing and watering-hole attacks to gain access to end-user systems, mainly Windows-based. Additionally, Earth Lusca uses servers that are accessible via the internet as another means of penetrating victims' systems. The authors of the analysis note that, for this

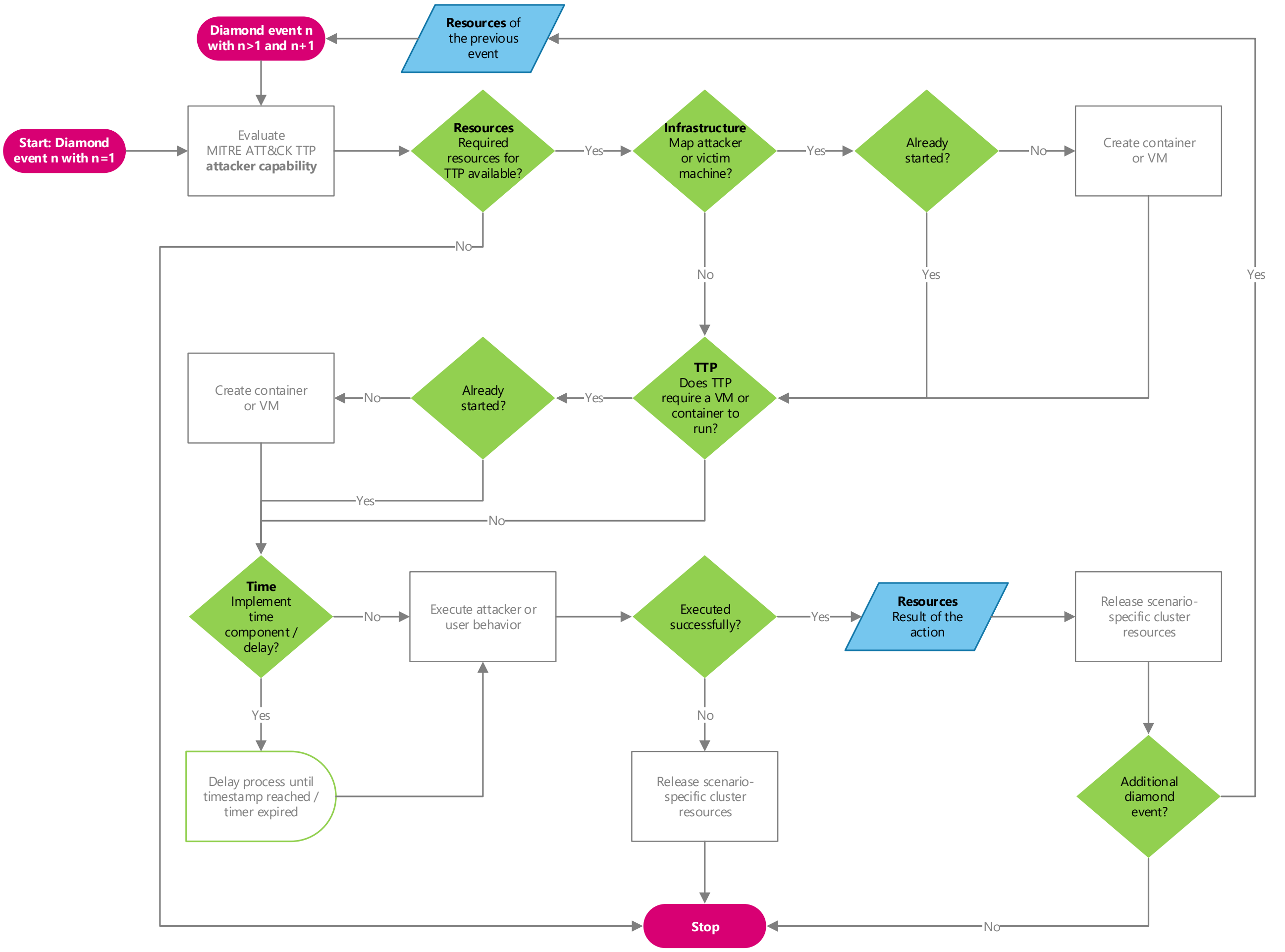
Diamond event n with n>1 and n+1
Resources of the previous event
Start: Diamond event n with n=1
Evaluate MITRE ATT&CK TTP attacker capability
Resources Required resources for TTP available?
Yes
Infrastructure Map attacker or victim machine?
Yes
Already started?
No
Create container or VM
No
No
Yes
Yes
Create container or VM
No
Already started?
Yes
TTP Does TTP require a VM or container to run?
Yes
No
Time Implement time component / delay?
No
Execute attacker or user behavior
Executed successfully?
Yes
Resources Result of the action
Release scenario-specific cluster resources
Yes
No
Delay process until timestamp reached / timer expired
Release scenario-specific cluster resources
Additional diamond event?
Stop
No

FIGURE 5. Processing of a Diamond Event with FASAC

purpose, the APT group compromises additional servers on the internet, which are then used as proxies, C2 infrastructure, and for target reconnaissance.

The previous study on Emissary Panda already focused on attacks on Windows systems and evaluated its implementation with T2P-DM. This case study provided more detailed information than the more general consideration of Earth Lusca. Therefore, we have avoided a renewed discussion of the execution of an attack against Windows systems. Instead, we have focused on Earth Lusca's actions regarding the initial exploitation of vulnerabilities that are available on publicly accessible servers.

To derive initial Diamond Events such as T_13 (MITRE TTP: T1190), we used confirmed TTPs and events from the Earth Lusca report, similar to the formalization of Emissary Panda with T2P-DM. An initial compromise was achieved by exploiting a vulnerability in Oracle GlassFish v4.1 [34] (T_13). We filled logical gaps with hypothetical Diamond Events. This is exemplarily described using:

- T_11: Search for vulnerable Oracle GlassFish servers with version 4.1 via Shodan (MITRE TTP: T1593.002)
- T_12: Search for a publicly available exploit for Oracle GlassFish v4.1 (MITRE TTP: T1588.005)

This consideration made it possible to show how the T2P-DM represents and links the connection between two different but interdependent cyber measures as separate activity threads. Fig. 7 shows the Activity Thread graphically.

C. EVALUATION OF THE TACTICS, TECHNIQUES, PROCEDURES DIAMOND MODEL

By analyzing two APT attacks using T2P-DM, it was possible to achieve a clearer structure for the attack history in both use cases and to provide more detailed information on the actions taken. By consistently describing the core and meta properties, as well as the results of the attack event, the information of the scenarios can be organized in a structured manner and their informative value can be enhanced. Furthermore, the adaptations made to the Diamond Model simplify its practical

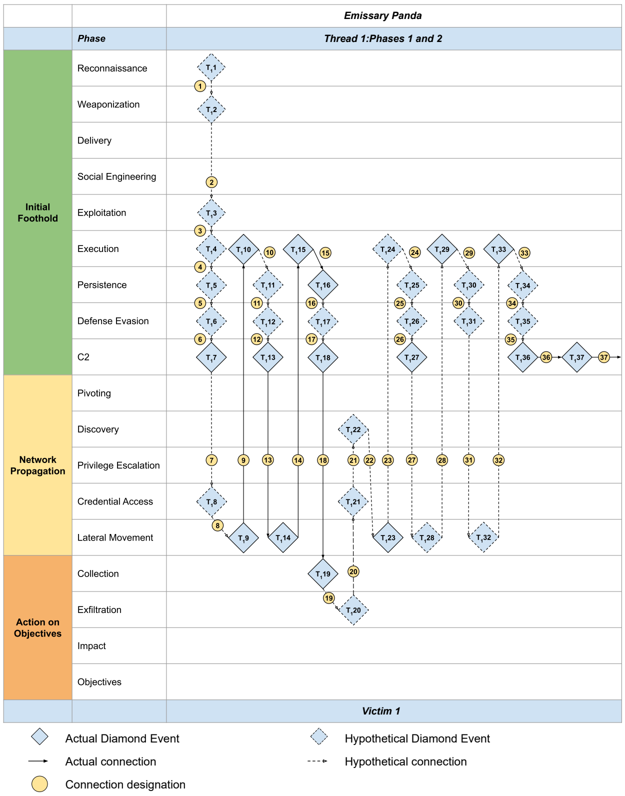

FIGURE 6. **Activity thread of the first two attack phases of Emmissary Panda**

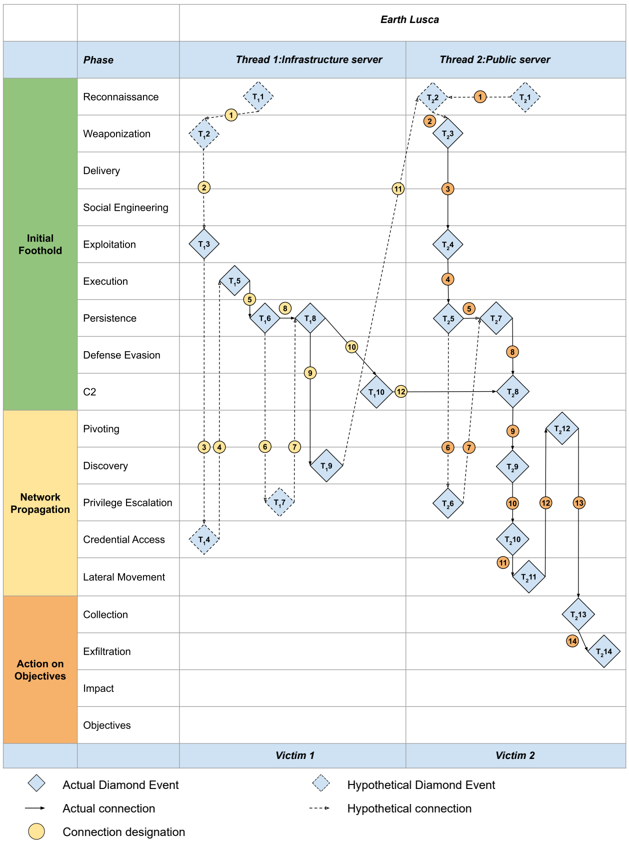

FIGURE 7. **Activity thread of Earth Lusca**

TABLE 1. **Comparison of Different Approaches**

Approach	Represents Attack Sequence	Enables Formalization of Attack Sequence	Enables Closing Logical Gaps	Enables Technical Implementation
Lockheed Martin [7]	✓			
Ussath *et al.* [8]	✓			
Zhang *et al.* [9]	✓			
Chen *et al.* [10]	✓			
Mandiant [11]	✓			
MITRE [13]	✓			✓
Pols [20]	✓			
Pols [21]	✓			
Caltagirone *et al.* [12]	✓	✓	✓	
T2P-DM	✓	✓	✓	✓

✓= Attribute of Approach

usage. By extending the core properties, the TTPs of the MITRE ATT&CK framework described in the scenarios can be directly integrated into Diamond Events. The UKC phase classification selected for the T2P-DM allows a more detailed classification of attack events than the CKC originally used. Hypothetical Diamond Events can be used to fill in missing information and to close gaps in the description of an attack. The use of activity threads enables the connection of successive Diamond Events in a directional graph, allowing for their assignment to specific attack phases and creating a clear temporal sequence of attack events.

Table 1 compares the approaches described in Section III-A. The T2P-DM has all the necessary attributes to enable a technical implementation.

D. EVALUATION OF FASAC

When evaluating FASAC, it is necessary to verify the technical capabilities of the framework. For this purpose, FASAC was implemented in the Cyber Range of the Research Institute CODE. The FASAC environment is based on a container cluster consisting of two VMs with 40 CPUs each, 256 GB RAM and 2 TB hard disk space. Furthermore, a DNS server is available for name resolution in the local network. Another VM is provided for orchestrating the container environment. Kubernetes was used as the container environment, with Rancher [35] and Harvester [36] being used for infrastructure administration, as shown in Fig. 8. The logic engine was implemented using the graphical logic editor Node-RED [37] in combination with Ansible [38] and the administration software AWX [39]. The FASAC code repository is available on

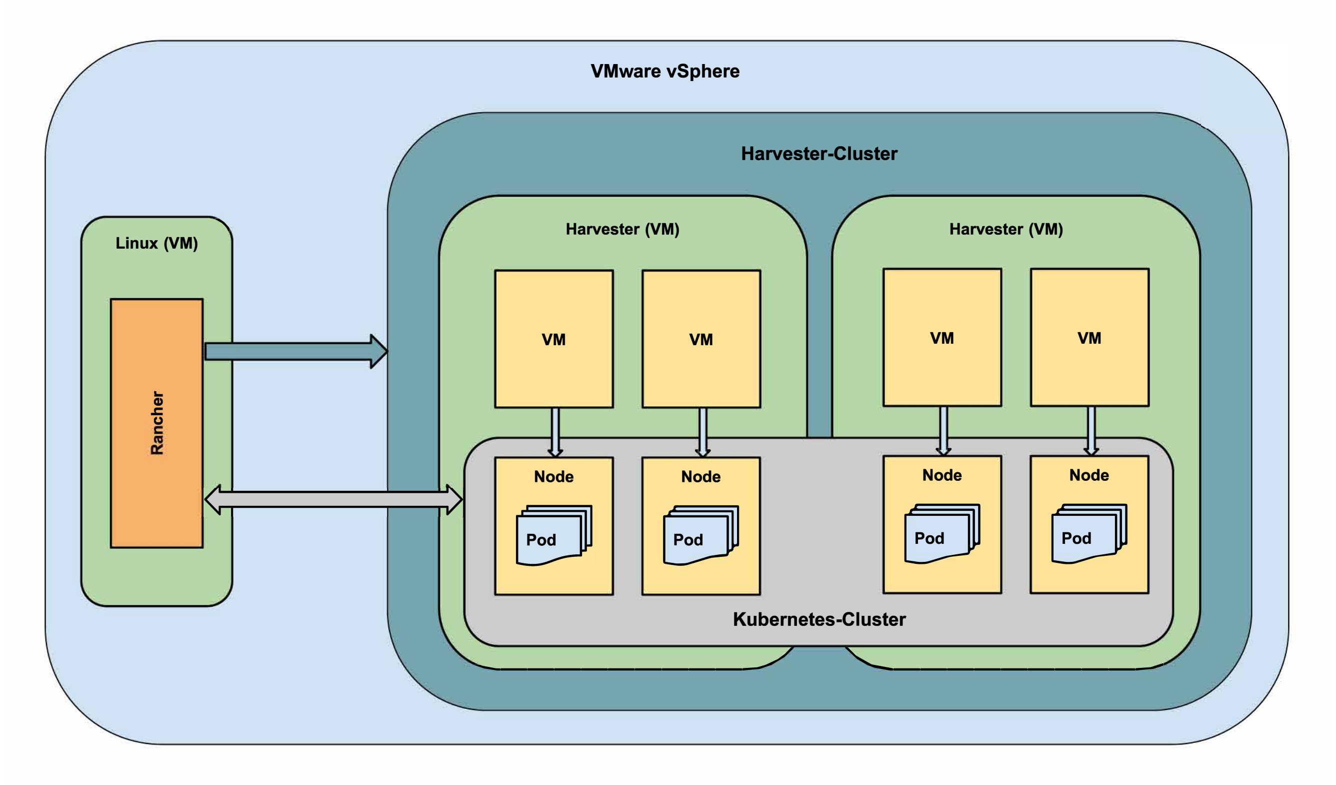

FIGURE 8. **Infrastructure stack**

GitHub [40].

The framework was evaluated using a fictitious scenario that demonstrates its capabilities and reflects common aspects of APT attacks. The scenario is described in the following:

The APT group APT999 is interested in the research work of the pharmaceutical company HealthForever AG. Through a successful cyber campaign against a globally operating logistics company, the group was able to capture thousands of contact details. As a result, APT999 also obtains the mail address of HealthForever AG's dispatcher. In order to obtain further information about the internal company and personnel structure, the dispatcher is to be attacked with a prepared spear phishing email. The group uses social media information to create a profile of the dispatcher in order to generate an email tailored to him. In the email, APT999 poses as the manufacturer of Detrack logistics software. A Meterpreter payload is used as the Remote Access Trojan. For camouflage purposes, it is placed as a free version of Detrack in the attachment of the mail. To ensure that the appearance of a valid business communication is maintained, the mail is sent during typical office hours. The company's dispatcher opens the mail's attachment on the same day, whereupon the Meterpreter starts executing on the victim system and connects to the attacker infrastructure of the APT999 group. To remain as inconspicuous as possible, the group does not interact with the attacked system until a day later. Using the pentesting framework Metasploit® [41], the attackers collect contact and delivery addresses of HealthForever AG. On the same day, the company's IT security department registers the unusual data traffic from the dispatcher's computer and disconnects the infected system from the network.

The scenario was analyzed according to the T2P-DM specifications, which were organized into individual Diamond Events. The analysis was then visualized in an Activity Thread, as shown in Fig. 9. The scenario was transferred to the technical environment using the modules developed for FASAC.

The available modules made programming skills unnecessary by allowing to integrate them into the workspace through a drag and drop functionality. The configuration was carried out via the input masks provided for the individual modules, which were then connected to each other. Depending on their functionality, modules received incoming data, processed it, and provided data for subsequent modules. Events can be executed in parallel, time-controlled or only after the fulfillment of predefined conditions. The use of FASAC enabled the mapping of a fictional scenario to a Cyber Range training in a fully virtualized environment. Table 2 provides a comparison between FASAC and the technical implementations described in Section III-B, demonstrating that FASAC is capable of implementing APTs in fully virtualized Cyber Ranges.

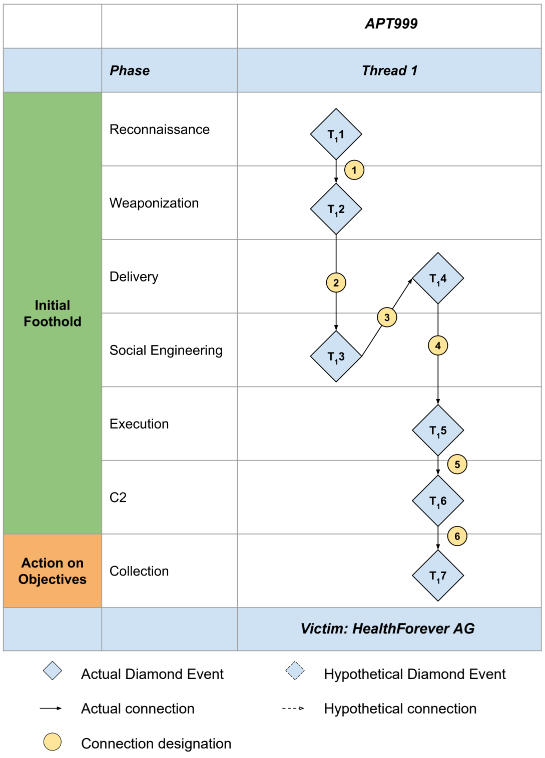

FIGURE 9. Activity thread of APT999

TABLE 2. Comparison of Different Implementations

Implementation	Complex Attack Sequence	Time-driven Actions	Event-driven Actions	User Simula-tion	Supports TTPs
CRATE [22]	✓	?	?	✓	(✓)
Splunk Attack Range [23]	?	?	?	?	✓
Metta [28]	(✓)	(✓)			✓
Infection Monkey [29]	(✓)				✓
FASAC	✓	✓	✓	✓	✓

✓= Enables Implementation of Functionality

(✓) = Partially Enables Implementation of Functionality

? = No Information Available

E. ETHICAL CONSIDERATION

Because T2P-DM and FASAC enable the detection, analysis, and replication of APTs, ethical considerations should not be completely neglected.

The proposed model and framework can reduce barriers to the replication and implementation of APTs. However, it is still necessary to have a technical understanding of adversary actions. Our model and framework are designed to investigate the attack flow of an APT, not to ease automated attack execution like the Metasploit® framework [41].

Implementing APTs in Cyber Ranges can enhance defensive measures against sophisticated attacks. Detailed analysis of attacks can be conducted and mitigation strategies can be evaluated without risking operational systems. This approach also enables teams to be trained on how to respond effectively to real APT attacks.

VI. CONCLUSION

To enhance protection against cyber criminals and their attacks, it is necessary to create a corresponding understanding and to provide adequate training opportunities. The presented model can be used to capture and analyze complex attacks in a structured manner. At the same time it serves as a foundation for replicating an APT attack using the FASAC framework.

With FASAC, a technical framework for the automated simulation of APT attacks was presented. Compared to other software solutions, it is completely based on a container environment and thus offers greater flexibility. The modular and event-driven approach allows realistic mapping of adversary and victim actions. Complex APT scenarios can be created using the integrated logic editor, without requiring programming knowledge. Similarly, time-consuming script parameterization is unnecessary as all required settings can be made in the configuration mask of the respective module.

While FASAC provides a basis for developing APT scenarios in Cyber Ranges, there are still open challenges that need further research and integration. An important characteristic of APTs is their temporal aspect, which typically spans several month or longer. For example, FASAC can simulate time periods using sleep functionalities. However, this approach is not practicable for long time periods. Therefore, a key question is how FASAC can simulate attacks that extend over a long period of time while minimizing simulation time. Another open question concerns the value of Cyber Range training to improve the resilience of participating organizations across multiple sessions. To enable FASAC to measure progress, a defined process is necessary to enable a comparison of target performance comparison before and after a training. For this purpose, appropriate measurement capabilities need to be implemented in FASAC.

Due to the modular implementation of the framework, the described enhancements can be easily integrated into FASAC as soon as they are developed. A contribution to the development of FASAC is always welcome [40].

REFERENCES

[1] B. Ballard, "Cybercrime apparently cost the world over $1 trillion in 2020," 2021. [Online]. Available: https://www.techradar.com/news/cybercrime-cost-the-world-over-dollar1-trillion-in-2020

[2] K. Zetter, *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*, first edition ed. New York: Crown Publishers, 2014.

[3] Kaspersky Lab, "BlackOasis APT and new targeted attacks leveraging zero-day exploit," Jul. 2017. [Online]. Available: https://securelist.com/blackoasis-apt-and-new-targeted-attacks-leveraging-zero-day-exploit/82732/

[4] S. Adair and T. Lancaster, "DriftingCloud: Zero-Day Sophos Firewall Exploitation and an Insidious Breach," Jun. 2022. [Online]. Available: https://www.volexity.com/blog/2022/06/15/driftingcloud-zero-day-sophos-firewall-exploitation-and-an-insidious-breach/

[5] D. Alperovitch *et al.*, *Revealed: operation shady RAT*. McAfee, 2011, vol. 3.

[6] D. Alperovtich, "Revealed: Operation Shady RAT," Santa Clara, California, 2011. [Online]. Available: https://icscsi.org/library/Documents/Cyber_Events/McAfee%20-%20Operation%20Shady%20RAT.pdf

[7] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," 2010. [Online]. Available: https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf

[8] M. Ussath, D. Jaeger, F. Cheng, and C. Meinel, "Advanced Persistent Threats: Behind the Scenes," in *2016 Annual Conference on Information Science and Systems (CISS)*, Institute of Electrical and Electronics Engineers, Princeton University, Ed., Princeton, NJ, 2016, pp. 181–186.

[9] R. Zhang, Y. Huo, J. Liu, F. Weng, and Z. Qian, "Constructing APT Attack Scenarios Based on Intrusion Kill Chain and Fuzzy Clustering," *Security and Communication Networks*, vol. 2017, 2017. [Online]. Available: https://downloads.hindawi.com/journals/scn/2017/7536381.pdf

[10] P. Chen, L. Desmet, and C. Huygens, "A Study on Advanced Persistent Threads," in *Communications and Multimedia Security : 15th IFIP TC 6/TC 11 International Conference, CMS 2014, Aveiro, Portugal, September 25-26, 2014 ; Proceedings / Bart De Decker ... (Eds.)*, ser. Lecture Notes in Computer Science: 8735, B. de Decker, Ed. Springer, 2014, pp. 63–72.

[11] Mandiant, "APT1: Exposing One of China's Cyber Espionage Units," 2013. [Online]. Available: https://www.mandiant.com/media/9941/download

[12] S. Caltagirone, A. Pendergast, and C. Betz, "The Diamond Model of Intrusion Analysis." [Online]. Available: https://apps.dtic.mil/sti/pdfs/ADA586960.pdf

[13] MITRE Cooperation, "Enterprise Techniques," 2023. [Online]. Available: https://attack.mitre.org/tactics/enterprise/

[14] National Institute of Standards and Technology, *Managing Information Security Risk - Organization, Mission, and Information System View*, ser. NIST Special Publication. Washington, D.C.: U.S. Department of Commerce, 2011, no. 800-39. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-39.pdf

[15] R. Kaschow, O. Hanka, M. Knüpfer, and V. Eiseler, "Cyber Range: Netzverteidigung trainieren mittels Simulation," in *D·A·CH SECURITY 2017*. München: Syssec, 15, pp. 126–137. [Online]. Available: https://www.syssec.at/de/veranstaltungen/dachsecurity2017/papers/DACH_Security_2017_Paper_13A3.pdf

[16] E. Ukwandu, M. A. B. Farah, H. Hindy, D. Brosset, D. Kavallieros, R. Atkinson, C. Tachtatzis, M. Bures, I. Andonovic, and X. Bellekens, "A review of cyber-ranges and test-beds: Current and future trends," *Sensors*, vol. 20, no. 24, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/24/7148

[17] B. Schneier, "Attack Trees: Modeling security threats," *Dr. Dobb's Journal*, no. 12, 1999. [Online]. Available: https://www.schneier.com/academic/archives/1999/12/attack_trees.html

[18] Mandiant, "M-Trends 2021: FireEye Mandiant Services | Special Report," 2021. [Online]. Available: https://www.arrow.com/ecs-media/16352/fireeye-rpt-mtrends-2021.pdf

[19] Carbon Black, "The Ominous Rise of "Island Hopping" & Counter Incident Response Continues," 2019. [Online]. Available: https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/docs/vmwcb-report-the-ominous-rise-of-island-hopping-and-counter-incident-response-continues.pdf

[20] P. Pols, "The Unified Kill Chain: Designing a Unified Kill Chain for analyzing, comparing and defending against cyber attacks," Masterarbeit, Cyber Security Academy, Den Haag, 2017. [Online]. Available: https://www.unifiedkillchain.com/assets/The-Unified-Kill-Chain-Thesis.pdf

[21] ——, "The Unified Kill Chain: Raising Resilience Against Advanced Cyber Attacks," 2022. [Online]. Available: https://www.unifiedkillchain.com/assets/The-Unified-Kill-Chain.pdf

[22] T. Gustafsson and J. Almroth, "Cyber range automation overview with a case study of CRATE," in *Secure IT Systems*, M. Asplund and S. Nadjm-Tehrani, Eds. Cham: Springer International Publishing, 2021, pp. 192–209.

[23] Splunk, "Splunk Attack Range," o. D. [Online]. Available: https://github.com/splunk/attack_range

[24] Red Canary, "Atomic Red Team," o. D. [Online]. Available: https://github.com/redcanaryco/atomic-red-team

[25] M. Velazco, "PurpleSharp," o. D. [Online]. Available: https://github.com/mvelazc0/PurpleSharp

[26] Prelude Research, "Prelude Operator," 2022. [Online]. Available: https://www.prelude.org/purpose

[27] ——, "Prelude Chains," 2022. [Online]. Available: https://chains.prelude.org

[28] Uber Technologies, "Metta," o. D. [Online]. Available: https://github.com/uber-common/metta

[29] Akamai Technologies, "Infection Monkey," 2022. [Online]. Available: https://www.akamai.com/infectionmonkey

[30] J. P. Morrison, "Flow-based Programming: Concepts," o.J. [Online]. Available: https://jpaulm.github.io/fbp/concepts.html

[31] HvS Consulting, "The APT Fallout of Vulnerabilties such as ProxyLogon, OGNL Injection and log4shell," Feb. 2022. [Online]. Available: https://www.hvs-consulting.de/public/ThreatReport-EmissaryPanda.pdf

[32] J. C. Chen, K. Lu, G. Chen, J. Horejsi, D. Lunghi, and C. Pernet, "Delving Deep: An Analysis of Earth Lusca's Operations," Jan. 2022. [Online]. Available: https://www.trendmicro.com/content/dam/trendmicro/global/en/research/22/a/earth-lusca-employs-sophisticated-infrastructure-varied-tools-and-techniques/technical-brief-delving-deep-an-analysis-of-earth-lusca-operations.pdf

[33] Mandiant, "Pst, want a shell? proxyshell exploiting microsoft exchange servers," 2021. [Online]. Available: https://www.mandiant.com/resources/blog/pst-want-shell-proxyshell-exploiting-microsoft-exchange-servers

[34] MITRE Cooperation, "CVE-2017-1000028," 2017. [Online]. Available: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-1000028

[35] Rancher, "Rancher," 2022. [Online]. Available: https://www.rancher.com

[36] Rancher Labs, "Harvester," 2022. [Online]. Available: https://harvesterhci.io

[37] Node-RED contributors, "Node-RED," 2022. [Online]. Available: https://nodered.org

[38] Red Hat, "Ansible," 2022. [Online]. Available: https://www.ansible.com

[39] AWX project contributors, "AWX," o. D. [Online]. Available: https://github.com/ansible/awx

[40] FASAC project contributors, "FASAC," 2022. [Online]. Available: https://github.com/spfuu/fasac

[41] Rapid7, Inc, "metasploit," 2006. [Online]. Available: https://www.metasploit.com/

Tore Bierwirth received the M.Sc. degree in computer science from the Carl von Ossietzky Universität Oldenburg, Germany, in 2014.

He is currently working as a Research Assistant at the Research Institute CODE, University of the Bundeswehr Munich, Germany.

His research interests addresses IT security in critical infrastructures with a focus on medical networks as well as IT security training in the context of Cyber Ranges.

Stefan Pfützner received the M.Sc. degree in cyber security from the University of the Bundeswehr Munich, Germany, in 2022.

He is currently working as an IT security consultant and advises federal and state authorities on IT and cyber security issues.

Matthias Schopp received the B.Sc. degree in business computer science from Baden-Wuerttemberg Cooperative State University and the B.A. degree in business administration from the Open University Milton Keynes in 2011 and the M.Sc. degree in Computer Science at the Ludwig-Maximilians-Universität of Munich, Germany, in 2014.

He is the laboratory supervisor of the Cyber Range at the Research Institute CODE, University of the Bundeswehr Munich, Germany with the main tasks developing new cyber security scenarios and implementing and executing Cyber Range training.

His research interest includes system security with a focus on cyber security and cyber range training, attacker methodology and cyber range architectures.

Christoph Steininger received the B.Sc. degree in business computing from the University of Passau, Germany, in 2017 and the M.Sc. degree in business computing from the University of Passau, Germany, in 2020. He also holds a M.Sc. degree in global information technology management from the University of Turku, Finland, awarded in 2020.

He is currently working as a Research Assistant and Ph.D. Student at the Research Institute CODE, University of the Bundeswehr Munich, Germany.

His research interest includes system security with a focus on cyber security and cyber range training, attacker methodology and cyber range architectures. He holds the cyber security certifications OSCP and OSWP and is involved in conducting cyber range training in the cyber range at the Research Institute CODE.

• • •