Cyber Range Training Programme Specification through Cyber Threat and Training Preparation Models

Michail Smyrlis[1,2][0000−0001−9527−516X], Konstantinos Fysarakis[1][0000−0002−6871−8102], George Spanoudakis[1,2][0000−0002−0037−2600], and George Hatzivasilis[3][0000−0002−2213−7759]

[1] Sphynx Technology Solutions AG, Zug, Switzerland
{smyrlis, fysarakis, spanoudakis}@sphynx.ch
https://www.sphynx.ch/
[2] Department of Computer Science,City, University of London,London, UK
{michail.smyrlis.2, g.e.spanoudakis}@city.ac.uk
https://www.city.ac.uk/about/schools/mathematics-computer-science-engineering/computer-science
[3] Institute of Computer Science Foundation for Research and Technology–Hellas (FORTH) Heraklion, Crete, Greece
hatzivas@ics.forth.gr
https://www.ics.forth.gr/

Abstract. In light of the ever-increasing complexity and criticality of applications supported by ICT infrastructures, Cyber Ranges emerge as a promising solution to effectively train people within organisations on cyber-security aspects, thus providing an efficient mechanism to manage the associated risks. Motivated by this, the work presented herein introduces the model-driven approach of the THREAT-ARREST project for Cyber Range training, presenting in detail the Cyber Threat Training and Preparation (CTTP) models. These models, comprising sub-models catering for different aspects of the training, are used for specifying and generating the Training Programmes. As such, the paper also provides details on implementation aspects regarding the use of these models in the context of a usable cyber range training platform and two specific training scenarios.

Keywords: Cyber Range · Cyber Security · Security Assurance · Training Programmes · CTTP Models · CTTP Programmes

1 Introduction

The increasing levels of complexity and inter-connectivity of ICT infrastructures, supporting a plethora of heterogeneous applications, have given rise to an increased number of perceived threats and cyber-attacks. Cyber criminals constantly improve their arsenal and launch impactful attacks that affected both organisations and individuals. This is exacerbated by the lack of security awareness, as users are not able to promptly identify and minimise the impact of a

cyber attacks, instead acting as enablers for the various threat actors to successfully deploy attacks [17]. According to PwC's Global Economic Crime and Fraud Survey 2020 [16], companies, on average, experienced 6 security incidents throughout the last 2 years whereas 47% of the interviewed had experienced fraud in the same period. The latter consists the 2nd highest report level of incidents in the past 20 years. According to Cybint Solutions [1], the average cost of a data breach in 2020 exceeds $150 million, while Gartner [11] estimated that the worldwide spending on cyber-security is forecasted to reach $133.7bn in 2022.

In this landscape, cyber-security training is becoming increasingly pertinent as an effective way of mitigating cyber risks. The need for not only more skilled cyber-security professionals but also well-trained individuals regardless of their security expertise is ever-increasing. Nevertheless, the cyber-security training should be implemented as a holistic approach and the gained knowledge should be validated. Part of a well-defined cyber security program, is the creation of information security awareness and training campaigns that would be able influence the adoption of an overall secure behaviour. To accomplish that, modern training strategies are not only limited to learning software and hardware skills, but also include training to understand actual cyber security threats along with resistance-training techniques [19]. However, cyber range training that does not have the capacity to fit the necessities of an organisation and to effortlessly adjust to the quickly developing scene, is deficient, and rapidly becomes obsolete [20].

Motivated by the above, this paper presents the Cyber Threat and Training Preparation (CTTP) Models and associated Training Programmes (CTTP Programmes) at the core of the model-driven Cyber Range Training approach developed under the H2020 THREAT-ARREST Project ([21],[22]). The delivery of Cyber Range Training Programmes is based on these CTTP models which define the structure and automate the development of the training programmes by determining a number of different aspects, such as: (a) the assets of a cyber-system, their relations and the threats covered by the CTTP Programme, (b) the ways these assets will be emulated and simulated, (c) the evaluation of the trainees based on their actions and level of expertise and (d) the preparedness and effectiveness level that the trainees are expected to achieve based on the targeted training programme. The benefit of having a model for every different aspect of a Training Programme, is the connection of it with the actual cyber system and its assessment allowing the trainee to interact with an actual cyber system. As of today, a model-driven approach that incorporates emulation, simulation, serious gaming and visualisation techniques aiming at preparing individuals with different types of responsibilities and level of expertise in defending high-risk cyber systems and organisations to counter-advanced, known and knew cyber-attacks does not seems to exist.

The remainder of this paper is organised as follows: Section 2 presents an overview of the Background & Related work, Section 3 describes the CTTP Models, Section 4 describes two CTTP Programmes created in the context of

the THREAT-ARREST Project and Section 5 provides the specification of one of these Training Programmes. Finally, Section 6 summarises the paper and sets future goals.

2 Background & Related Work

To the best of our knowledge, there are only a few model-driven approaches that allow the whole specification of a Cyber Range Training program. Russo et al. [18] propose a Scenario Definition Language (SDL) based on the OASIS topology and Orchestration Specification for Cloud Applications (TOSCA) used as a components specification language. SDL is similar to the CTTP Specification Language [5] used in THREAT-ARREST, that allow us to specify the different components of a cyber system. Erdogan et al. [10] introduce a training and evaluation approach based on the CORAS risk models [19] that specify cyber-risk models in order to facilitate real-time risk assessment and evaluation of trainees. Similarly, the definition of the CTTP Models will drive the training process, and align it (where possible) with operational cyber system security assurance mechanisms to ensure the relevance of training. Lastly, Braghin et al. [2] provide a model-driven engineering approach based on the creation of a subset of the CTTP model, namely the Emulation sub-model (see Table 2). The approach presented herein is based on the Security Assurance Model proposed by Somarakis et al. [20], extended to cover the needs of the Cyber Range training developed under the H2020 THREAT-ARREST project. According to Yamin et al. [23], existing model-driven cyber range approaches lack the ability to validate their models against real word scenarios. Contrariwise, the proposed approach is generic, thus it can be applied to both various domains and people with different levels of expertise and cyber-security knowledge. To demonstrate this, we have created a number of Training Programmes and applied them to three different pilots: shipping, healthcare and smart-energy [6].

3 Cyber Threat And Training Preparation Models

At the core of the model-driven approach to Cyber Range training proposed in this paper is the development of the CTTP Model. The creation of the latter consists of three main phases namely: (a) the Cyber System Analysis and creation of the Core Assurance Model, (b) the *Creation of CTTP sub-models* and (c) the *Training Programme definition*. Thus, a CTTP Model has (at a minimum) three compulsory parts namely: (a) the *Core Assurance Model*, (b) the *Training Model Generation and Delivery parameters* and (c) the *Emulation*, *Simulation*, *Gamification*, *Data Fabrication* sub-model or a combination of them. The corresponding sub-models will be analysed in the subsections that follow. This process, as visualised in Fig. 1, is in line with expected stages in cyber range programme development and execution; as in the case of the THREAT-ARREST platform [4], such platforms need to incorporate emulation, simulation, serious gaming and data fabrication capabilities to be able to adequately prepare

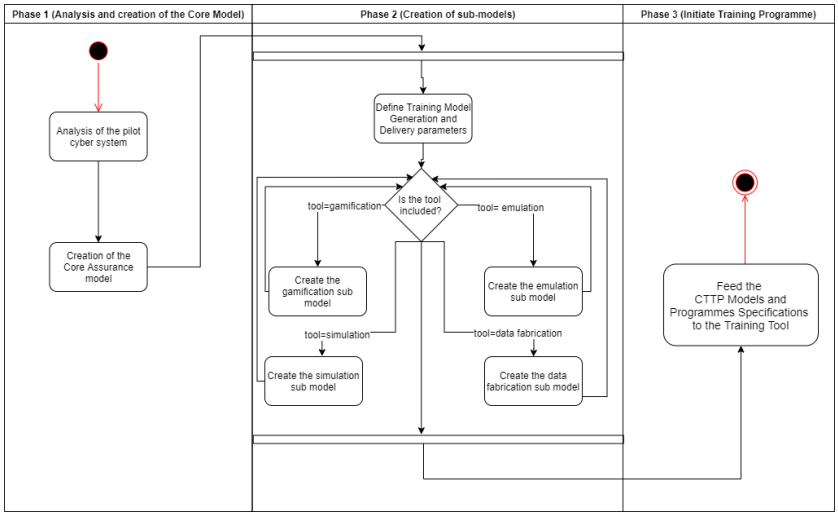

Fig. 1. Overall methodology for creating Training Programmes

stakeholders with different types of responsibility and levels of expertise in realistic and customisable scenarios tailored to the intricacies of each organisation's cyber-systems and the most pertinent cyber-attack scenarios [22].

Core Assurance Model The Core Assurance model specifies the cyber system(s) that the Training Programme will be related to. More precisely, it determines the assets of the cyber system, their relations and their corresponding threats. An asset can be a software asset, hardware asset, physical infrastructure asset, data and person. Each asset has a number of required and optional fields. For instance, the required fields of a software asset include the *name*, *vendor* and *version*; these are also used in order to construct the Common Platform Enumeration (CPE) [3] (an industry standard that can also be used as part of the vulnerability analysis conducted from the assurance component of the THREAT-ARREST platform). Other required fields include *the type of asset* (e.g. software or physical asset) and its *kind* (service or component). Optional fields include the *value* of the asset (in monetary terms), the *date that the asset will cease to exist* (if applicable) and a brief *description*. Each asset also comes with a flag value named *status* that allows the THREAT-ARREST platform to decide whether it will be part of the overall assessment. For instance, if the status is set to *draft*, the definition of the different fields is not yet finalised, thus it will not take part in the overall assessment. Lastly, the assets defined in the Core Assurance Model are utilised in order to define the emulation and simulation parts of the CTTP Programme.

Emulation sub-Model The emulation sub-model includes the information of the emulated components that will be used as part of the Training and is intended to be being dynamically parsed by Emulation tool (e.g., virtual infrastructure management solutions based on OpenStack [15] or Kubernetes [13]). An emulation sub-model includes the: *Training Programme name*, the *status* (as described in Sect. 3), the *created and termination date* and one or more **module types**. An emulation module type includes : (a) the core information of the deployed Virtual Machine (VM), (b) its network configuration and (c) the scripts to be run at boot. The core information of the VM fields are: the *name* of the VM, the type of operating system, the specification of the connection to be used to remotely connect to the VM, the allocated size of RAM and disk, the name of the image to deploy, using as index in the OpenStack [15] repository along with the username and, in case of a Windows VM, the password, the virtual network the VM is connected to - accompanied with a fixed address, if applicable- and the id of the script module that will run at boot (if applicable). The network configuration module includes: the id that will be used as a reference point in the core information module, the Classless Inter Domain Routing (CIDR) - a method for allocating the IP address and IP routing- and the default gateway. Lastly, the script module includes the *script id* used as the reference point in the core information module- and the script to be run at boot.

Simulation sub-model The simulation sub-model includes the information of the simulated components that will be used as part of the Training phase and are intended to be dynamically parsed by simulation components (e.g., NS-3 [14]). A simulated sub-model includes: (a) the core information of the simulation environment (e.g. the *name* of the simulated tool, the *simulation template*, the *deployment mode*, the *initial execution time*, the *execution speed* etc.) and one or more simulation module types. A simulation module type includes one or more components. A component can be a *root* one, i.e. a component that has a one or more child components and is first in the component hierarchy, or a child. Each component has a number of fields such as: the *name* of the component container, the *id* of the the Training Programme it will be involved, the *type* of the component (used as a descriptor of the internal java class of the simulation tool), one or more *component attributes* (i.e. values that hold a certain state of the simulation tools during the simulation phase), a flag value that describes whether this component is root or child and the *connections* between the different components.

Gamification sub-model The Gamification sub-model includes the information needed to create a Gamification environment integrated into the training platform; e.g., in the case of THREAT-ARREST, this is provided by the Social Engineering Academy (SEA) gamification tools [7]. This sub-model consists of one or more *Game* modules. A *Game* module includes the following fields: (a) the type of the game (e.g., AWARENESS QUEST [7] or PROTECT [12], as supported in THREAT-ARREST), (b) the difficulty level, (c) the overall game

time, (d) one or more card deck id's and (e) whether this game needs any special practise.

Data Fabrication sub-model Lastly, the Data-Fabrication (DF) sub-model includes information intended to be provided to data fabrication tools, e.g. the IBM Data Fabrication Platform developed by IBM Israel [9] that is integrated into THREAT-ARREST. The DF sub-model includes: (a) the core information of the Data-Fabrication tool (such as the *name*, *status*, *created and termination date*) and (b) the declaration of the *network-attached computers*, *switches* and other relevant hardware (structured as nodes).Each hardware node is augmented with properties and "installed" software applications and services.

Training Model Generation and Delivery parameters The Training Model Generation and Delivery parameters determine the way in which the Training Programme will be structured; in the case of THREAT-ARREST, this is parsed by the Training Tool developed by ITML [8]. The parameters are descriptors that allow the trainer, trainee and the Training Tool to understand the scope of a specific Training Programme as well as the tools involved in it. To be more precise, the parameters include: (a) a brief *description* of the Training Programme, (b) the *expected goal* of it, (c) the *difficulty*, (d) the *maximum score* that the trainee can achieve and (e) the *base score* that the trainee should achieve in order to successfully complete the programme, (f) the *examined actions* that trainees are expected to take against cyber-attacks covered by the programme (e.g., preparedness, incident detection and analysis, real time incident response, and post incident response), (g) the *role(s)* that the trainee will have (e.g. system administrator, end-user etc.), (h) the *owner* of the Training Programme and (i) *educational material* that will allow the trainee to better understand the scope of the Training Programme. The parameters also describe the *Training Session Specification* which define the number of screens that will be presented to the trainee, the *order* each screen will be presented, the *difficulty* of each screen, the *duration* this screen will stay available and the *tool* (e.g. Emulation, Simulation or Gamification) involved in each of these screens. A screen is also accompanied by a *hint*, that if the trainee chooses to use, will have a negative impact to its final score. Lastly, each screen comes with one or more *expected traces*. The latter tracks the progress of the user and has three different versions: (a) the *Evaluation Report* where the trainee is being assessed by answering questions regarding the deployed defence mechanisms, the potential threats etc., (b) the *Event Captors* which monitor if the correct configuration steps have been executed and (c) the *Gamification Report* which checks the total score as reported by the Gamification tool, the number of lost lives and the remaining time.

4 CTTP Programme Scenario Definition

The creation of a CTTP Programme is based on a CTTP Model with the purpose of specifying training scenarios, focusing on particular threats, cyber system

components and assessment tools that are pertinent for the specific targeted environment (e.g., vertical domain or specific application), as defined in the model. This drives the execution of simulation, emulation and serious gaming processes, to realise within the cyber range the scenario environment and the steps implementing it.

As of today 13 such CTTP Training Programmes have been defined in the context of THREAT-ARREST, covering the domains of shipping, smart energy and healthcare. In this section, two of them will be presented: (i) the "Response & Mitigation" Programme of the smart-energy pilot in the context of smart home-/IoT environments, and; (ii) the "Navigation combo attack (phishing email and GPS spoofing)" in the context of Smart Shipping applications. Each programme includes a brief description, the progression steps and the Programme modelling.

4.1 Smart Home/IoT - Threat Response & Mitigation Training Programme

This Training Programme aims to train end-users with no security knowledge (as is typically the case for IoT/Smart Home consumers) on how to response to an abnormal behaviour and take immediate actions in order to mitigate the risk. The Programme involves the Emulation, Simulation and Gamification tool and is modelled based on Lightsources' cyber system (see Fig. 2).

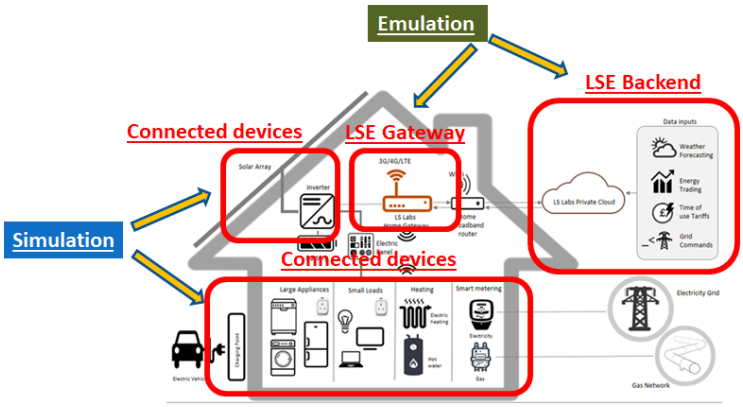

Fig. 2. Smart Energy pilot architecture and Virtual Lab deployment

Description In this scenario, the trainee (user) is the owner of a smart plug, and the web based Lightsource application allows users to monitor its power consumption and/or its on/off behaviour. It also provides alerts of the system if an abnormal behaviour is detected. An intruder has gained access to the smart plug and executed a malicious application which stopped the smart plug from reporting its power consumption and turned a switch on and off at random time points. The user is notified by an alert, through the web application, that an abnormal behaviour was detected, and is asked to read the Lightsource guideline

provided during the setup phase, in order to bring the device back to its expected behaviour.

Progression Steps The individual steps comprising the scenario are as follows:

1. The trainer sets up the gateway and provides the log files and the database schema that contains the end users' credentials (in an encrypted form) and the IP of the smart plug. He/she also sets up the private cloud that provides the alerts to the web-based application of the trainee.
2. The trainee is informed about the security concerns surrounding smart devices and, upon installation of the edge device, receives an incident response and abnormal behaviour guideline.
3. The trainee receives an alert to its web-based application letting him/her know that the smart plug stopped reporting the power consumption and that the device connected to it reports abnormal on/off patterns. The trainee opens the web-based application to check if the alert was correct.
4. The trainee reads the guideline and, as instructed in the first step, resets the smart plug to its factory settings by pressing its button for 10 seconds. Then, he/she checks the graphs presented in the web application, but he observes that the abnormal behaviour is still there (i.e. no power consumption is presented).
5. The trainee then moves to the second step of the guideline and resets the device itself.
6. Finally, the trainee checks the graphs, and observers that both the smart plug started reporting its power consumption and the connected device was not reporting abnormal behaviour.

Training Programme Modelling To realise this Training Programme, the following cyber range platform components are leveraged:

- The **Emulation tool** facilitates the following Virtual Machines:
  - The Gateway VM with a number of log files and the database schema pre-installed.
  - The VM that will involve the Simulation Tool.
  - The private cloud VM.
  - The trainee PC that includes a web browser allowing the trainee to open the Lightsource application.
- The **Simulation** and **Visualisation tool**:
  - Simulates the smart plug and a button for the device connected to it.
  - Three different phases are presented:
    - Normal Behaviour
    - Faulty/Compromised smart plug device
    - Compromised Device
- The **Gamification tool** presents a game for smart home security awareness
- The **Training Tool** includes:
  - A short course for security awareness in general
  - Lightsource' incident response guideline

4.2 Smart Shipping - Navigation combo attack (phishing email and GPS spoofing)

This Training Programme aims to train the decision-making of end-users with moderate security knowledge. The Programme involves the Emulation, Simulation and Gamification tools and is based on DANAOS' cyber system (see Fig. 3).

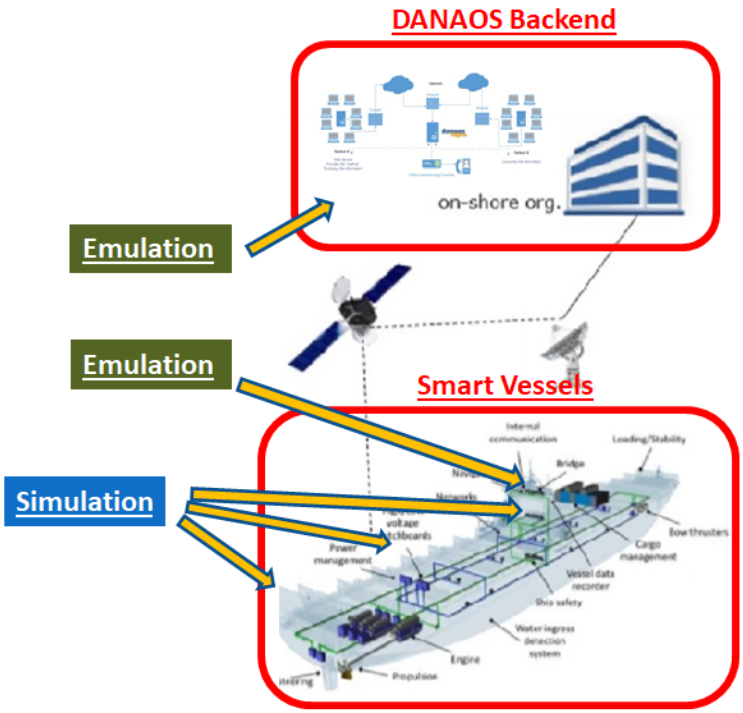

Fig. 3. Smart Shipping pilot architecture and Virtual Lab deployment

Description In this scenario, the decision-making of trainee (captain) is being tested. More specifically, it consists of two different phases. During the first phase, a set of malicious / faulty / legitimate emails is being sent to the trainee in order to mislead him/her in performing requested actions. The second phase takes place after the ship has started its journey and consists of a GPS spoofing attack, where the trainee should identify it and perform a set of actions to ensure that the ship will safely arrive to its final destination.

Progression Steps The individual steps comprising the scenario are as follows:

1. The trainee (captain) starts a journey from the port of Heraklion to the port of Piraeus (which will be designated by the back-end office via an email to the captain).

2. A faulty (but legitimate) email, commanding the trainee to go to Thessaloniki's port, is being sent. The email contains the details of another journey and was sent to the trainee by mistake.
   (a) The trainee identifies that this is a legitimate email.
   (b) Since the destination port was Thessaloniki, the trainee understands that this email was sent to him/her by mistake.
   (c) The trainee ignores the email and reports it back to the back-end office.
3. Then, the trainee receives a malicious (phishing) email, alerting him/her that a bad weather condition will take place, thus, he/she needs to go to another port to make a stop. The trainee needs to identify that this is a phishing email, ignore it and report it to the back-end office.
4. Lastly, the trainee receives a legitimate email with the weather forecast, denoting that the weather is good, and the destination is the Piraeus port.
5. During the trip, the trainee checks a **simulated** digital map that presents the current ship's position based on GPS data and the predetermined route (checkpoints) from Heraklion to Piraeus. The trainee realises that the ships' position on the digital map (receiving signal from a GPS receiver) is away from the designated way point and the ship is off course. The trainee should check if this is due to his/her own navigational orders or due to external factors (strong current streams) and should correct course by returning to the predetermined route or, if something is wrong, make use of the navigational monitor (digital map). The trainee proceeds with an order of actions to validate position from the GPS signal. The orders of actions are once again stem from the CTTP Model.
6. The trainee checks a magnetic compass and the marine paper map (Nautical Charts), in order to understand the actual position of the ship.
7. While checking the compass, he/she understands that it points towards a different direction to the ship course. Following, the trainee marks on the Nautical Charts, the position as depicted in the GPS (faulty coordinates). Then, the trainee is crosschecking objects (navigation aids, restrictions, bathymetry) mapped on charts with what he/she observes by looking outside the ship's bridge windows with his/her binoculars and with what he/she receives from other bridge equipment (e.g. bathymetry on the map against see depth from echo sounder). The trainee understands that the ship is navigating on different routes than those corresponding to the position given by GPS (faulty coordination).
8. Finally, the trainee understands that a GPS spoofing attack might have occurred, stops following the Digital Map Application (received signal from GPS receiver) and manually navigates the ship to its correct destination (by turning off the auto pilot).

Training Programme Modelling To realise this Training Programme, the following cyber range platform components are leveraged:

- The **Emulation Tool** facilitates the following Virtual Machines:
  - The trainee operates the VM for the captain's PC.

  - The faulty/malicious and legitimate messages are being sent by the VM that includes the trainer's mail application.
  - The Simulation and visualisation VM.
- The **Simulation Tool** contains the simulated on-deck navigation equipment, i.e. the Digital Map (GPS Receiver), the magnetic compass and the Nautical Charts
- The **Gamification tool** presents a game for social engineering.
- The **Training tool** includes a short course for social engineering.

5 CTTP Programme Model Specification

The final step towards instantiating a Training Programme is the CTTP Programme Model Specification, i.e. encoding the training scenario parameters into an instance of the CTTP model that will be used to drive the actual training. For the sake of brevity, this section will only provide an example of this process for the Response & Mitigation Training Programme defined in Sect. 4.1. A similar process can be followed for any other scenario.

As previously mentioned, this Training Programme involves the Emulation, Simulation and Data Fabrication tools. Thus, three sub-model instances (along with the Core Assurance Model and the Training Model Generation and Delivery parameters) will be presented in the subsections that follow.

5.1 Core Assurance Model

Table 1 shows a subset of the Response & Mitigation Core Assurance model, specified using the CTTP Specification Language [5]. The model specifies three different assets, two software assets and one person.

Table 1. Core Assurace Model

```
1 Person(firstName("Technician"), lastName("N/A"),
  ↪ email("tech@lightsourcelabs.com"), project("Response & Mitigation"),
  ↪ organisation("LIGHTSOURCE LAB LTD"),activeTo(2025-11-19
  ↪ 13:55),description("Technician of the
  ↪ organisation"),roles(technician)),
2 SoftwareAsset (vendor("MariaDB"), version("10.3.18"), name("MariaDB"),
  ↪ kind(Service), type(SAL), project("Response &
  ↪ Mitigation"),organisation("LIGHTSOURCE LAB
  ↪ LTD"),owner("Technician"),description("Open source database solution
  ↪ for modern, mission-critical applications."))
3 SoftwareAsset (vendor("NGINX"), version("1.17.7"), name("NGINX"),
  ↪ kind(Service), type(SAL), project("Response &
  ↪ Mitigation"),organisation("LIGHTSOURCE LAB
  ↪ LTD"),owner("Technician"),description("NGINX accelerates content and
  ↪ application delivery, improves security, facilitates availability
  ↪ and scalability for the busiest web sites on the Internet"))
```

5.2 Emulation sub-model

Table 2 shows a subset of the Response & Mitigation Emulation sub-model converted in an XML format. The XML is then converted (by the Emulation Tool) to a HEAT template and is being deployed in OpenStack [2]. More specifically, this sub-model specifies the creation of a Virtual Machine and its network configuration.

Table 2. Emulation sub-model

```
<?xml version="1.0" encoding="Unicode" standalone="yes"?>
<Scenario name="UC1-LSE">
    <CustomVM name="broker_UC1" os="linux">
        <connectionmode port="22" connectiontype="ssh"/>
        <ram val="4096"/>
        <vcpus val="2"/>
        <disk val="40"/>
        <image name="LSE-broker" val="UC1-broker-v7" username="debian"/>
        <Network idref="home_network" fixedip="192.168.33.20"/>
    </CustomVM>
    <Networks>
        <Network id="home_network">
            <gateway name="gateway-home_network" val="192.168.33.1"/>
            <cidr name="cidr-home_network" val="192.168.33.0/24"/>
            <is_external val="false"/>
        </Network>
    </Networks>
</Scenario>
```

5.3 Simulation sub-model

Table 3 shows a subset of the Response & Mitigation Simulation sub-model in a JSON format. This subset includes two simulation components. The Smart Home root component (see root = true) and the SmartPlug child component (see root = false). The latter includes an attribute that checks if the initial value of the smart plug is set to "WORKING" and specifies that this value can change throughout the simulation phase.

5.4 Gamification sub-model

Table 4 shows the Gamification sub model, specified using the CTTP Specification Language and converted in a JSON format. More specifically, this sub-model includes the PROTECT game with difficulty level of 2, total game time of 12 minutes and the SmartHome card deck.

Table 3. Simulation sub-model

```
...,
"SimulationComponents": [
{
    "name": "SmartHome",
    "simulatedComponent": "SmartHome",
    "type": "jasima.core.Simulation.SimComponentContainerBase",
    "root": true,
    "componentContainers": [
    {
        "simpleComponents": [
            {
                "name": "SmartPlug",
                "internalID": "SmartPlug",
                "type": "smarthome.SmartPlug",
                "root": false,
                "attributes": [
                    {
                        "name": "plugState",
                        "initialValue": "WORKING",
                        "type": "smarthome.SmartPlugStateEnum",
                        "canChange": true
                    },
                    ...
                ]
            },
...
]
```

Table 4. Gamification sub-model

```
"games": [
    {
        "gameType": {
            "gameTypeID": 1,
            "game": "Protect"
        },
        "protects": [
            {
                "difficultyLevel": 2,
                "gameTime": 12,
                "cardDeckID": "cd_smarthome",
                "specialPractice": false
            }
        ]
    },
    ...
]
```

5.5 Training Model Generation and Delivery parameters

Lastly,as mentioned in Section 3, the Training Model Generation and Delivery parameters are parsed by the Training Tool in order to instantiate the Training Programme. Table 5 shows a subset of the Response & Mitigation Training Model Generation and Delivery parameters in a JSON format. This subset presents the Training Programme goal, the maximum score the trainee can achieve, the base score he/she needs in order to succeed, its difficulty and the educational materials (see *Bibliography* field) that will be made available to the trainee.

Table 5. Training Model Generation and Delivery parameters

```
[
    {
      ...,
        "scenarioGoal": {
            "description": "This scenario trains an end user with no
            ↪  security knowledge on how to response to an abnormal
            ↪  behaviour and take immediate actions in order to
            ↪  mitigate the risk. The scenario is implemented in an
            ↪  Emulation, Simulation and Gamification tool.",
            "maxScore": 10,
            "successScore": 5
        },
        "difficulty": 7,
        "bibliographies": [
            {
                "name": "Incident Response & Abnormal behaviour",
                "text": "Lightsource's incident response guideline"
            },
            ...
        ],
...
]
```

6 Conclusions & Future work

This paper presented the THREAT-ARREST's Cyber Threat and Training Preparation (CTTP) Models and the process followed for the specification of the associated Training Programmes, which are at the core of the platform's model-driven cyber range training approach. While this model-driven approach requires some effort and introduces a level of complexity to create and parse the CTTP Models, it also enables the use of an evidence-based approach to cyber range training, and the provision of programmes that are mapped to the actual cyber system and the results of its security assessment, thus targetting the most

pertinent threats in the context of the training. Moreover, the creation of CTTP models facilitates deployment of several variations of the Training Programmes (e.g., to cater for trainees with different levels of expertise) and a thorough evaluation of both the trainee and the programme itself.

As a next step, efforts will focus on specifying all the Training Programmes defined within [6], as well as on identifying new ones, based on the analysed results of the actual pilot cyber systems, provided by the Security Assurance Tool integrated within the THREAT-ARREST platform. This tool will also be used as part of the adaptation of the CTTP Models and Programmes in the piloting environments based on updates to the threat landscape. In this context, an analysis will be carried out on the impact that changes in the CTTP Programme can have, while also checking the completeness and consistency of the entire specification of CTTP Models and Programmes in the context of these changes.

Acknowledgements This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 786890 (THREAT-ARREST).

References

[1] *15 Alarming Cyber Security Facts and Stats*. 2019. URL: https://www.cybintsolutions.com/cyber-security-facts-stats/.

[2] Chiara Braghin et al. "A Model Driven Approach for Cyber Security Scenarios Deployment". In: *Computer Security*. Springer, 2019, pp. 107–122.

[3] *Common Platform Enumeration (CPE)*. 2020. URL: https://csrc.nist.gov/projects/security-content-automation-protocol/specifications/cpe/.

[4] *D1.3: THREAT-ARREST platform's initial reference architecture*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D1.3%5C%20-%5C%20THREAT-ARREST%5C%20platform%5C%E2%5C%80%5C%99s%5C%20initial%5C%20reference%5C%20architecture.pdf.

[5] *D3.1: CTTP Models and Programmes Specification Language*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D3.1%5C%20-%5C%20CTTP%5C%20Models%5C%20and%5C%20Programmes%5C%20Specification%5C%20Language.pdf.

[6] *D3.3: Reference CTTP Models and Programmes Specifications*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D3.3%5C%20-%5C%20Reference%5C%20CTTP%5C%20Models%5C%20and%5C%20Programmes%5C%20Specifications%5C%20v1.pdf.

[7] *D4.2: THREAT-ARREST serious games v1*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D4.2%5C%20-%5C%20THREAT-ARREST%5C%20serious%5C%20games%5C%20v1.pdf.

[8] *D4.3: Training and Visualisation tools IO mechanisms v1*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D4.3%5C%20-%5C%20Training%5C%20and%5C%20Visualisation%5C%20tools%5C%20IO%5C%20mechanisms%5C%20v1.pdf.

[9] *D5.1: Real event logs statistical profiling module and synthetic event log generator v1*. 2020. URL: https://www.threat-arrest.eu/html/PublicDeliverables/D5.1%5C%20-%5C%20Real%5C%20event%5C%20logs%5C%20statistical%5C%20profiling%5C%20module%5C%20and%5C%20synthetic%5C%20event%5C%20log%5C%20generator%5C%20v1.pdf.

[10] Gencer Erdogan et al. "An Approach to Train and Evaluate the Cybersecurity Skills of Participants in Cyber Ranges based on Cyber-Risk Models". In: June 2020.

[11] *Gartner Forecasts Worldwide Information Security Spending to Exceed $124 Billion in 2019*. 2020. URL: https://www.gartner.com/en/newsroom/press-releases/2018-08-15-gartner-forecasts-worldwide-information-security-spending-to-exceed-124-billion-in-2019.

[12] Ludger Goeke et al. "PROTECT–An Easy Configurable Serious Game to Train Employees Against Social Engineering Attacks". In: *Computer Security*. Springer, 2019, pp. 156–171.

[13] *Kubernetes*. 2020. URL: https://kubernetes.io/.

[14] *NS-3 Network Simulator*. 2020. URL: https://www.nsnam.org/.

[15] *OpenStack*. 2020. URL: https://www.openstack.org/.

[16] "PwC's global economic crime and fraud survey 2018". In: (2020). URL: https://www.pwc.com/gx/en/forensics/gecs-2020/pdf/global-economic-crime-and-fraud-survey-2020.pdf.

[17] Konstantinos Rantos, Konstantinos Fysarakis, and Charalampos Manifavas. "How effective is your security awareness program? An evaluation methodology". In: *Information Security Journal: A Global Perspective* 21.6 (2012), pp. 328–345.

[18] Enrico Russo, Gabriele Costa, and Alessandro Armando. "Scenario design and validation for next generation cyber ranges". In: *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*. IEEE. 2018, pp. 1–4.

[19] Peter Schaab, Kristian Beckers, and Sebastian Pape. "Social engineering defence mechanisms and counteracting training strategies". In: *Information & Computer Security* (2017).

[20] Iason Somarakis et al. "Model-Driven Cyber Range Training: A Cyber Security Assurance Perspective". In: *Computer Security*. Springer, 2019, pp. 172–184.

[21] Othonas Soultatos et al. "The THREAT-ARREST Cyber-Security Training Platform". In: *Computer Security*. Springer, 2019, pp. 199–214.

[22] *THREAT-ARREST*. 2019. URL: https://www.threat-arrest.eu/.

[23] Muhammad Mudassar Yamin, Basel Katt, and Vasileios Gkioulos. "Cyber ranges and security testbeds: Scenarios, functions, tools and architecture". In: *Computers & Security* 88 (2020), p. 101636.