A Digital Twin for the 5G Era: the SPIDER Cyber Range

Filippo Rebecchi*, Antonio Pastor†, Alberto Mozo‡, Chiara Lombardo§, Roberto Bruschi§, Ilias Aliferis¶, Roberto Doriguzzi-Corin‖, Panagiotis Gouvas**, Antonio Alvarez Romero††, Anna Angelogianni,‡‡, Ilias Politis‡‡, Christos Xenakis‡‡

*Thales SIX GTS France, Gennevilliers, France. Email: name.surname@thalesgroup.com

†Telefonica I+D, Madrid, Spain. Email: antonio.pastorperales@telefonica.com

‡Universidad Politécnica de Madrid, Madrid, Spain. Email: a.mozo@upm.es

§S2N National Laboratory, CNIT, Genoa, Italy. Emails: {chiara.lombardo, roberto.bruschi}@cnit.it

¶Unisystems Greece, Athens, Greece Email: AliferisI@unisystems.gr

‖Fondazione Bruno Kessler, Trento, Italy. Email: rdoriguzzi@fbk.eu

**Ubitech, Athens, Greece. Email: pgouvas@ubitech.eu

††Atos, Seville, Spain. Email: antonio.alvarez@atos.net

‡‡University of Piraeus, Athens, Greece. Emails: {angelogianni, politis, xenakis}@unipi.gr

***Abstract*—Service providers, 5G network operators and, more generally, vertical industries face today a dangerous shortage of highly skilled cybersecurity experts. Along with the escalation and growing sophistication of cyber-attacks, 5G networks require the training of skilled and highly competent cyber forces. To meet these requirements, the SPIDER cyber range focuses specifically on 5G, and is based on three pillars, (i) cyber security assessment, (ii) training cyber security teams to defend against complex cyber-attack scenarios, and (iii) evaluation of cyber risk. The SPIDER cyber range replicates a customized 5G network, enabling the execution of cyber-exercises that take advantage of hands-on interaction in real time, the sharing of information between participants, and the gathering of feedback from network equipment, as well as the development and adaptation of advanced operational procedures. This aims to help 5G security professionals improve their ability to collaboratively manage and predict security incidents, complex attacks, and propagated vulnerabilities. The SPIDER cyber range is validated in two relevant use case scenarios aimed at demonstrating, in a realistic, measurable, and replicable way the transformations SPIDER will bring to the cybersecurity industry.**

***Index Terms*—5G, Digital Twin, Cyber Range**

I. Introduction

5G is expected to connect many facets of our societies, including critical sectors such as transportation, utilities, healthcare, industry 4.0, and critical infrastructure. However, it is a well-known fact that many cyber security challenges, both in terms of standards, technologies, and deployment still need to be addressed [1].

5G incorporates sophisticated infrastructure that enables end-to-end programmability ranging from the backhaul to the radio. Being the result of the de-facto convergence occurred among Communication Service Providers (CSP) and Cloud Infrastructure Providers (CIP), 5G relies heavily on virtualization technology, thus allowing an agile service deployment via the so-called "slicing" concept. However, in terms of security, the widespread use of virtualization has contributed to increase radically the exposed attack vectors. These vectors can be combined by skilled attackers to manipulate some of the infrastructure and cause widespread damage [2]. It is also important to consider that the specificities of 5G are such that it is practically impossible to devise a holistic training for any single human operator.

In this paper, we illustrate the approach taken in the SPIDER H2020 project to deliver an innovative digital twin platform targeted at the training of experts in the cyber security of 5G network deployments [3]. The SPIDER cyber range allows developing cyber exercises including both self-paced challenges, such as incident-response practice and reverse engineering, and team-based exercises, such Red Team vs. Blue Team games and Capture the Flag (CTF). Trainees can learn to apply both offensive and defensive techniques specific to 5G network and infrastructure components on target environments closely matching the real-world networks.

The main objectives of the SPIDER cyber range are thus:

- Ability for the trainees to conduct exercises in a realistic yet safe 5G environment. The environment is configured and instantiated *on-demand*, encompassing specific configurations for slice capabilities and its components.
- Ability to automatically infer performance tracking of trainees. Emphasis is given to the *passive tracking* of trainee activities (i.e., without human in the loop). Such inference is used primarily to *evaluate* training activities, then to identify the learning gaps that need to be covered;
- Ability to deploy exercises in which *machine learning components* (e.g., a cryptomining attack detector) can be used by Blue Teams as part of a defensive toolbox.

II. Related Work

Cyber ranges are training environments containing both physical and virtual components, capable of representing realistic scenarios for cyber training and of supporting the practical acquisition of knowledge through hands on activities [4]. Started as military-oriented tools, in recent years they have

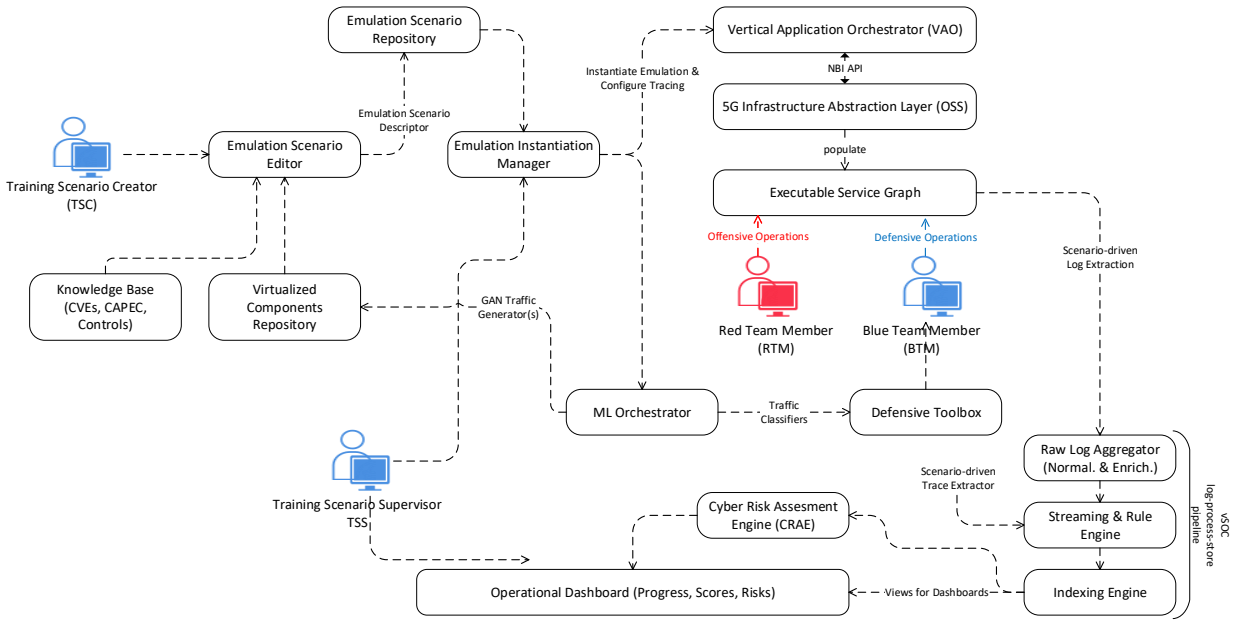

Fig. 1. Overview of SPIDER Reference Architecture.

gained ground in industrial, commercial, and academic environments, offering additional capabilities in niche applications such as Internet of Things (IoT) [5], cyber-physical systems [6], and smart grids [7]. In line with this trend, SPIDER specializes on 5G networks, which represent the backbone of future Information Communications Technology (ICT) systems. By adopting an emulative approach [8], SPIDER is able to reproduce a realistic 5G infrastructure. Cyber ranges can also be categorized by their objective: research & development (testing implementations, methods, tools, building blocks, and systems); training & education (academia, specialized security courses and cyber-security certifications); and exercises & competitions (CTFs or Cyber Defense Exercises) [9].

III. Architecture

The analysis of functional requirements resulted in the definition of 14 components that constitute the SPIDER reference architecture. Figure 1 and Table I illustrate those components.

A Training Scenario Creator (TSC) is the person in charge of creating the training scenarios and performing the corresponding system configurations. Each emulated scenarios involve one or more ethical hackers (Red Team) and/or DevOps engineers (Blue Team) that will compete against each other on a virtual playground. The *Emulation Scenario Editor* allows defining a training scenario and authoring a valid *Emulation Scenario Descriptor* via a Graphical User Interface (GUI), guaranteeing as well its structural and business validity. The resulting *Emulation Scenario Descriptor* models all aspects of a scenario, i.e., its type, the target audience (in terms of experience), the relevant attacks, the expected escalation graph, the virtual assets composing the scenario and their connections. The TSC is assisted in the creation of the scenario by a *Knowledge Base* component that persists information

TABLE I
SPIDER Architectural Components

Component	Description
Knowledge Base	Keep track of existing 5G asset types, vulnerabilities, threats, and controls
Emulation Scenario Editor	Manage the creation of a service graphs. These service graphs are deployed on top of a programmable 5G testbed
Emulation Scenario Repository	Persist the service graphs that have been created by the Emulation Scenario Editor
Virtualized Components Repository	Manage registration of virtual components. Such components may be vulnerable (with a given CVE) or even deliberately misconfigured during their instantiation (in the frame of an exercise). Each component can be used in the frame of a service graph
Emulation Instantiation Manager	Load an emulation scenario from the repository and coordinate its instantiation in the 5G resource
VAO	Handle the choreography of service graph life-cycle management. It interacts with the OSS Northbound API. It is triggered by the Emulation Instantiation Manager
OSS	Handle the configuration and management of programmable 5G infrastructure according to the needs of an emulation scenario
ML Orchestrator	Employed off-line to train specific ML models that can be used for sophisticated offensive or defensive activities
Raw Log Aggregator	Receive and aggregate raw logs of all runtime components (Virtual Machines of applications, switches, VNFs)
Streaming & Rule Engine	Offer a Complex Event Processing functionality (i.e. time window operations) that is used to issue specific events that are valuable for SPIDER analysis
Indexing Engine	Persist all logs and events, making them searchable
CRAE	Track in real-time the evolution of the cyber risk exposure related to the emulated infrastructure
Operational Dashboard	Visualize the progress of a running emulated scenario

regarding existing services, their vulnerabilities, the cyber threats that each vulnerability is bound to, and the control elements that may be applied upon. Similarly, the actual virtualized components referenced from the templates are stored into a *Virtualized Components Repository*. Once a scenario is created, it has to be persisted as a "template" in the *Emulation Scenarios Repository*, which stores the scenario descriptors.

A Training Scenario Supervisor (TSS) manages the execution of training scenarios on top of the platform. The TSS interacts with the *Emulation Instantiation Manager* in order

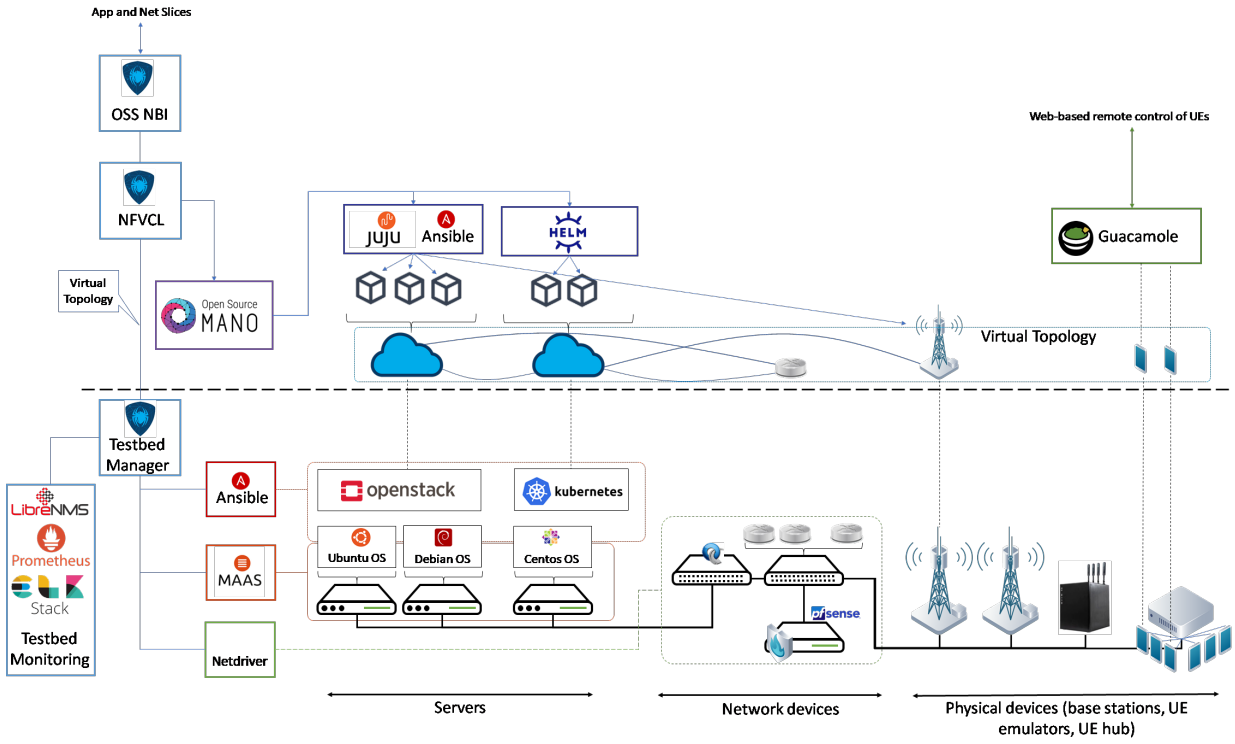

Fig. 2. MATILDA testbed integration with SPIDER.

to materialize training scenarios. This process implies the deployment and configuration of specific virtualized services on top of the 5G infrastructure. The instantiation process is not a one-shot operation, and for each scenario a specific 5G slice has to be created first. Thus, two distinct orchestration engines work together: i) *Vertical Application Orchestrator (VAO)*, which launches the negotiation of slice and deploys vertical applications; and ii) *Operations Support System (OSS)*, which creates the requested 5G slice. Upon instantiation, both Red and Blue Team members are provided with specific metadata for engaging with the instantiated scenario, materialized in the form of an *Executable Service Graph*.

During training, actions from Red/Blue Team Members are recorded through specific "tracers" installed and configured on the instantiated components via the VAO to automate performance tracking. Logged events are forwarded to a log-store-process pipeline acting as *virtual Security Operation Center (vSOC)*. Logs flow through a *Raw Log Aggregator*, to be harmonized and enriched, then through a *Streaming Engine* with scenario-specific rules that determine whether specific milestones for both the Red/Blue Teams are achieved. As a last step, logs are indexed to be efficiently queried. Refined metrics for both Red and Blue teams are visualized in dedicated performance dashboards, where the TSS is able to introspect their performance. A limited view of such dashboards is also granted to Red and Blue Team members.

Teams have the freedom to bring-their-own tools for both offensive/defensive operations. Some specific utilities are generated by a *Machine Learning (ML) Orchestrator* as an offline process. The *ML orchestrator* is used to create both defensive and offensive primitives. The defensive primitives are then persisted in a *Defensive Toolbox*.

IV. Testbeds

The proposed architecture is materialized on two different 5G testbeds, that are introduced in the following.

A. The MATILDA testbed

The SPIDER Cyber Range solution integrates with the MATILDA's orchestration framework for vertical applications and network services over 5G network sliced infrastructures [10]. Figure 2 overviews the components of such testbed. In particular, the OSS North-Bound Interface (NBI) and the NFV Convergence Layer (NFVCL), along with the Testbed Manager, are SPIDER-specific components developed to manage the deployment of the 5G slices, namely interacting with the VAO for materializing the slices, managing the lifecycle of the network services, and setting up the whole testbed before the instantiation of a training scenario, including the virtual topology with Virtual Network Functions (VNFs), Cloud-native Network Functions (CNFs), and Physical Network Functions (PNFs). Guacamole provides an interface for terminals lacking native control features, such as smartphones [11]. The physical part of the testbed include multiple virtualization servers, the networking devices, and physical devices such as gNBs, UEs and UE emulators. A testbed manager allows to manage all the physical, computing and network components available, thanks to a GUI linked to the software tools active

in the system, such as MaaS [12], LibreNMS [13], OpenStack, and Kubernetes and the monitoring tools.

B. The MOUSEWORLD testbed

The second integration is with the Mouseworld testbed [14]. Its reference architecture, depicted in Figure 3, follows the Digital Twin Network (DTN) concept proposed by the network management research group (NMRG) of the IRTF [15]. The Mouseworld testbed is employed to generate realistic labelled datasets that feed the *ML Orchestrator* in order to train ML models applied to both offensive and defensive tools employed in the SPIDER cyber range exercises. Mouseworld relies on the ETSI Open Source MANO (OSM) orchestrator stack to provision both VNFs and CNFs on top of OpenStack and Kubernetes Virtualized Infrastructure Managers (VIMs). The Service Mapping Model component is in charge of the instantiation of the network twin leveraging on a set of interconnected virtualization servers, allowing the independent execution of multiple scenarios, and the generation of the associated traffic and telemetry. Typical scenarios for SPIDER include the replication of a complete 5G network, including User Equipment (UE), RAN and the 5G Core. Mouseworld integrates a commercial traffic generator, such as Keysight IXIA Breaking Point, for experimenting with network attacks under high traffic load. Moreover, an isolated network with replicated internet services is made available to carry out additional security experiments. The Data Repository consists of a collector and storage module gathering the generated traffic in different formats from packet-level to aggregated telemetry (e.g. Netflow v9). An application layer supports the transformation of collected data into valuable datasets features, for example, by extracting statistical features from typical network flows. A Tagger entity adds labels to each flow using external and logs information output generated during the execution of each experiment. Flow tagging is highly dependent on the machine learning task and must be carefully designed for each type of scenario instantiation. Finally, data science tools such as TensorFlow and Keras are employed to develop ML tools to be transferred to SPIDER cyber range.

V. Key Features

A. Emulation Scenario Definition

An instantiated emulation scenario in SPIDER consists of a configured 5G network environment including multiple assets — such as UE or UE emulators, vertical application components, VNFs and PNFs, VIM, tenant spaces, and Software Defined Network (SDN) controllers. Each of the assets involved can contain some vulnerabilities, either inherent system properties tagged according to the Common Vulnerability Enumeration (CVE) system, or "deliberate misconfigurations", that is possible to exploit using specific tactics and techniques (e.g., by following the MITRE ATT&CK framework [16]) Moreover, each emulation scenario can be associated with specific learning objectives (i.e., perform attack/defence actions). Scenarios are modeled using an Emulation Scenario

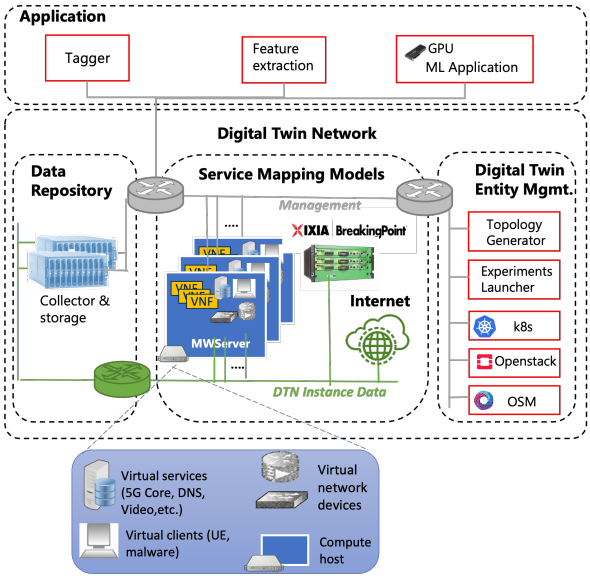

Fig. 3. Mouseworld DTN testbed.

Descriptor, which encodes all their aspects in a JavaScript Object Notation (JSON) format.

B. Management and Orchestration

As mentioned in Section III, the management and orchestration of 5G network slices is independent from the deployment of vertical applications. Therefore, the life cycle management of executable service graphs is split into two orchestration engines working together:

- VAO handles slice negotiation, as well as the deployment and decommissioning of vertical applications;
- OSS handles 5G slice creation, including the coordination of all the other building blocks in the testbed platform.

This split allows managing and orchestrating vertical applications without the need to know telco-specific information on the underlying infrastructure. The OSS NBI is the main interface point fostering the interworking between the VAO which governs the UE, application, and edge computing domains, and managing the chains of components that define specific application graph, and the orchestration of 5G network slices mainly realized by the NFV Orchestrator (NFVO). In the first step of the orchestration, the VAO requests the OSS for a 5G slice matching the application-specific requirements. This is done by handing over a slice intent, which is codified in JSON according to a pre-defined meta-model; the OSS first interacts with the testbed components (in particular with the NFVO) ensuring the proper setup of the involved slices via the NFVCL, then exposes the JSON-based description of the materialized slice via its NBI Application Programming Interfaces (APIs). Once a proper 5G slice is set up, the VAO can access the vertical application tenant spaces in the VIMs to deploy vertical applications using a specific proxy component, while the NFVO can access the Network Function Virtualization (NFV) tenant spaces for the lifecycle management of the involved VNFs and PNFs.

Once the executable service graph deployed, the VAO supports performing Day 2 operations as the training progresses. This include specific configuration of the asset to be realized at runtime, upon the realization of a certain condition, and the installation and configuration of tracers to infer the actions from the red/blue team members the dynamic.

C. Tracing & Progress Quantification

Aside from the information related to the scenario deployment, the *Emulation Scenario Descriptor* also encapsulates specific tracing and performance extraction requirements, which are jointly handled by the VAO and the vSOC components. Log extraction agents are installed and configured on each components involved in the scenario, and interface with data shippers such as Filebeat, eBPF, and Prometheus exporters. This enables to passively introspect the end-to-end infrastructure, and to elaborate data following a log-process-store pipeline to extract training performance. The log-process-store pipeline that constitutes the vSOC is separated into three distinct sub-components that allow tracing user interactions with the platform and providing visibility on their actions.

The *Raw Log Aggregator* collects logs from the log extraction agents installed on the components for normalization, enrichment, and in order to store them as meta-data for further usage in investigations and reports. Event sources include servers, switches, routers, storage arrays, operating systems, and firewalls. The VAO is responsible to initiate and configure the Log Extraction Agent as part of day 2 operations. In order to process incoming normalized events, a fast and scalable processing mechanism is required (*Streaming & Rule engine*). A stack of different technologies is employed to facilitate those processing requirements. The core off-the-shelf components that play a significant role in the processing pipeline are Kafka, Logstash, and the Complex Event Processing (CEP) engine. Kafka has been selected among other queue systems for its exceptional scaling capabilities and its ability to perform CEP queries. CEP capabilities are provided by dedicated components such as Kafka SQL, Esper, Siddhi. Finally, the *Indexing Engine* includes a high available transactional relational database, a linear scalable document-oriented storage, and an object store (artifactory). For the sake of reference implementation Elasticsearch is used, being it open source, highly scalable and extremely documented. The Indexing engine allows the definition of multiple index-templates, indicatively including raw-logs and performance-indicators. The Indexing Engine is crucial since it is the one-stop-shop for operational dashboards. All analytic-oriented queries are performed on-top of the Indexing Engine.

D. Cyber-risk assessment

The Cyber Risk Assessment Engine (CRAE) component can be used during the emulation exercises to track in real-time the evolution of the cyber risk exposure related to the emulated infrastructure. This component is pre-deployed in the SPIDER platform and can be connected on demand to the dynamically deployed infrastructure that will be used during the emulation exercises. Once an exercise starts, the CRAE benefits from the information collected from the emulated infrastructure for monitoring purposes to update the evolution of the cyber risk exposure. In this way, if an attack is taking place the calculated cyber risk will go up as it is properly detected and identified, while the cyber risk will go down as mitigation measures are put in place. Hence, the CRAE allows to analyse the evolution of the exercise and to determine if attackers or defenders are dominating the game.

E. Machine Learning Orchestration

The Machine Learning Orchestrator is part of the Digital Twin Entity Management (Figure 3) in the Mouseworld testbed. The Topology Generator employs predefined templates, interacting with OSM (e.g., ETSI NFV SOL-005 interface) to provision an *executable service graph*. An Experiment Launcher is in charge of day-2 configurations and to trigger the emulation functions for dataset generation. This is based either on ProxyCharms or dedicated Ansible scripts. The Experiment Launcher uses a configuration file, to define the statistical distribution of traffic, the number of intervals in which the experiment is divided, its duration, and the type of emulated services. Currently, two SPIDER-specific scenarios have been defined: crypto mining malware detection and DNS attack.

F. Synthetic Attack Generation with GANs

One of the most innovative aspects of the SPIDER cyber range is its capability to automate offensive tactics by generating synthetic traffic traces. In this regard, a significant effort has been devoted to the study of the application of the recently appeared Generative Adversarial Networks (GANs) to generate synthetic flow-based network traffic to mimic both attacks and normal traffic [17]. In contrast to other approaches using GAN as a data augmentation solution, synthetic data generated in SPIDER can fully replace real data (both attacks and normal traffic). With this approach, any machine and deep learning models (MDL) trained with synthetic data will obtain similar performance to models trained with real data when both are tested and deployed in real-time scenarios.

The advantages of this innovative solution to generate synthetic traffic are the following: (i) SPIDER GAN solution obtains synthetic flow-based traffic capable of fully replacing real data, and therefore, this solution can be applied in scenarios where data privacy have to be guaranteed; (ii) the synthetic network traffic generation can be seamlessly integrated into a cyber range platform such as SPIDER in both Blue team based (defensive) exercises and Red team based (penetration test) exercises; and (iii) as data generated by the GAN can be shared (exported and imported) without incurring in any privacy violation, or without exposing sensitive security policies and network configuration, we envision the creation of an ecosystem composed by multiple federated cyber ranges that exchange data and GAN models for the generation of synthetic traffic.

On top of synthetically generated data, SPIDER can also leverage on publicly available pre-recorded traces that combine

benign traffic and a range of different attack types [18], as well as on open-source network traffic generators and sniffers (e.g., `hping3` [19], `mousezahn` [20], `tcpdump` [21]) to generate basic volumetric DDoS attacks such as SYN Flood. The generated traffic traces can be employed both to train the ML algorithms involved in the cyber exercises (e.g., ML-based Intrusion Detection Systems (IDSs)), and to simulate actual attacks by re-injecting the traffic into the network using `tcpreplay` [22].

VI. Use Cases & Validation Activities

SPIDER demonstrates and validates both the effectiveness of the platform as a whole, and the efficacy of the proposed learning scenarios through two distinct use cases; the first is dedicated to the testing of the 5G-ready applications and services, while the second is dedicated to the 5G cybersecurity training of experts. The reasoning behind this separation is the scope. While the first use case aims at providing a 5G cyber range with advanced capabilities (i.e., ML) for testing, evaluating performance and assessing the security of new technologies, the second use case aims at elevating the skills of cybersecurity experts in the emerging 5G landscape. Both the MATILDA and the MOUSEWORLD testbeds are employed, in order to offer a holistic experience.

To validate the use cases, SPIDER has defined a methodology which includes specific steps: i) monitor of the technical Key Performance Indicator (KPI) of the system's performance and efficiency of developed modules, ii) monitor of the effectiveness of the educational and learning module and ii) monitor of the perceived Quality of Experience (QoE) metrics.

The *technical KPIs* are further categorised in: i) general KPIs, ii) 5G KPIs, iii) cybersecurity KPIs, and iv) ML KPIs. The *educational KPIs*, are closely monitored by the scoring system proposed by SPIDER, as well as the progress of the Red or Blue team member. Lastly for the *QoE metrics* there are four sub-groups, classified as follows. The first is about technology acceptability metrics and is largely based on the work in [8], [23]. The second one refers perceived trust, which is critical when assessing cyber ranges. The third refers to system usability, as measured by interaction inspection, which is an indicator of consumer acceptance. The last one refers to the system's ability to handle user error and misuse.

VII. Conclusion

The paper has presented the architectural approach of the SPIDER cyber range, explaining the focus on 5G positioning, and including a detailed description of its main components. The cyber range platform takes full advantage of advanced network orchestration, log-processing data pipeline, cyber risk assessment frameworks, and applies advanced ML techniques in support of its hands-on learning objectives.

Future work will address both the validation of the platform through the realization of the pilot use cases with real users, and the incorporation of more complex network scenarios.

Acknowledgment

This work is partially supported by the European Commission under the H2020 SPIDER project under grant agreement number 833685.

References

[1] I. Ahmad, S. Shahabuddin, T. Kumar, J. Okwuibe, A. Gurtov, and M. Ylianttila, "Security for 5g and beyond," *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 3682–3722, 2019.

[2] ENISA. (2020) Enisa threat landscape for 5g networks. [Online]. Available: https://www.enisa.europa.eu/publications/enisa-threat-landscape-report-for-5g-networks/@@download/fullReport

[3] SPIDER: a cybersecurity platform for virtualised 5g cyber range services. [Online]. Available: https://spider-h2020.eu/

[4] NIST. (2020) The Cyber Range: A Guide.

[5] O. Nock, J. Starkey, and C. M. Angelopoulos, "Addressing the security gap in IoT: towards an IoT cyber range," *Sensors*, vol. 20, no. 18, p. 5439, 2020.

[6] G. Kavallieratos, S. K. Katsikas, and V. Gkioulos, "Towards a cyber-physical range," in *Proceedings of the 5th on Cyber-Physical System Security Workshop*, 2019, pp. 25–34.

[7] B. Hallaq, A. Nicholson, R. Smith, L. Maglaras, H. Janicke, and K. Jones, "Cyran: a hybrid cyber range for testing security on ics/scada systems," in *Cyber Security and Threats: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2018, pp. 622–637.

[8] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, and M. A. Ferrag, "Cyber ranges and testbeds for education, training, and research," *Applied Sciences*, vol. 11, no. 4, p. 1809, 2021.

[9] 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, p. 7148, 2020.

[10] R. Bruschi, J. F. Pajo, F. Davoli, and C. Lombardo, "Managing 5G network slicing and edge computing with the MATILDA telecom layer platform," *Computer Networks*, vol. 194, p. 108090, 2021.

[11] Apache. (2020) Guacamole. [Online]. Available: https://guacamole.apache.org/

[12] Canonical Ltd. (2022) Metal-as-a-service 3.1. [Online]. Available: https://maas.io/

[13] (2021) Librenms. [Online]. Available: https://www.librenms.org/

[14] A. Pastor, A. Mozo, D. R. Lopez, J. Folgueira, and A. Kapodistria, "The Mouseworld, a security traffic analysis lab based on nfv/sdn," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, 2018, pp. 1–6.

[15] C. Zhou, H. Yang, X. Duan, D. Lopez, A. Pastor, Q. Wu, M. Boucadair, and C. Jacquenet, "Digital twin network: Concepts and reference architecture," Internet Engineering Task Force, Tech. Rep., 2021. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-zhou-nmrg-digitaltwin-network-concepts-06

[16] MITREA. MITRE ATT&CK. [Online]. Available: https://attack.mitre.org/

[17] A. Mozo, Á. González-Prieto, A. Pastor, S. Gómez-Canaval, and E. Talavera, "Synthetic flow-based cryptomining attack generation through generative adversarial networks," *Scientific Reports*, vol. 12, no. 1, pp. 1–27, 2022.

[18] Canadian Institute for Cybersecurity. (2022) Datasets. [Online]. Available: https://www.unb.ca/cic/datasets/index.html

[19] Salvatore Sanfilippo. (2006) hping. [Online]. Available: http://www.hping.org/

[20] Ulrich Weber. (2010) Mausezahn. [Online]. Available: https://github.com/uweber/mausezahn

[21] The Tcpdump Group. (2022) Tcpdump and libpcap. [Online]. Available: https://www.tcpdump.org/

[22] AppNeta Inc. (2022) Tcpreplay - Pcap editing and replaying utilities. [Online]. Available: https://github.com/appneta/tcpreplay

[23] F. D. Davis and V. Venkatesh, "Toward preprototype user acceptance testing of new information systems: implications for software project management," *IEEE Transactions on Engineering management*, vol. 51, no. 1, pp. 31–46, 2004.