MobiDojo: A Virtual Security Combat Platform for 5G Cellular Networks

Hyunwoo Lee[†], Haohuang Wen[†], Phillip Porras[‡], Vinod Yegneswaran[‡],
Ashish Gehani[‡], Prakhar Sharma[‡], Zhiqiang Lin[†]
[†]The Ohio State University, [‡]SRI

***Abstract*—The fifth-generation (5G) cellular network has advanced significantly, becoming a crucial component of modern communication. However, there are still many inherent security vulnerabilities in the 5G network standard, which advocates continuous research and development efforts. To this end, there are various open-source 5G software and public testbeds for 5G network testing and research. While those tools are valuable, users with limited expertise often struggle to deploy a 5G network and conduct sophisticated security testing with these platforms. To fill this gap, we introduce MobiDojo, the first virtual 5G security testing platform that supports one-click 5G deployment and security testing with web-based graphical user interfaces. MobiDojo is built on entirely virtual (i.e., no radio hardware required) open-source software - the OpenAirInterface's 5G stack deployed as Docker containers, making it compatible with any commodity servers. Another critical capability of MobiDojo is its attack simulation plugins that allow users to execute existing attacks or create custom Packet Capture (PCAP)-based 5G attack payloads and test them within an isolated 5G test network. We anticipate MobiDojo could drive many valuable applications, including education, Capture-the-Flag (CTF) competitions, 5G security research, defense evaluation, etc., ultimately helping to improve the transparency and security of 5G networks.**

I. Introduction

The fifth-generation (5G) cellular network technology brings numerous advantages over its predecessors, enabling groundbreaking applications that were previously unfeasible due to limitations in speed, latency, and capacity. With 5G, we can expect to see the rise of smart cities, remote surgeries, and automated factories, among other innovative use cases. The technology offers significantly higher data transmission speeds, supports a greater number of connected devices, and aims to reduce network energy consumption and latency through advanced features [1].

Security Challenges. However, 5G also introduces new security challenges that must be addressed. On one hand, as 5G operates through an open wireless channel, many exploitations through this interface have become feasible. In fact, both the 3rd Generation Partnership Project (3GPP) [2] and the security community have discovered numerous vulnerabilities and exploits, many of which even originate from the previous generations. Examples of such involve cellular service disruption [3], [4], [5], [6] and tracking end-user locations [7], [8], [9], [10], by injecting, manipulating, and sniffing malicious 5G-specific protocol signals. On the other hand, as 5G networks are moving towards a software-defined network architecture, such as the integration of virtual network functions (NFV) and edge computing devices, this collectively expands the attack surface by introducing more potential entry points for traditional cyberattacks [11]. All these emerging security threats have made securing 5G networks a more challenging task compared to previous generations. Therefore, ensuring the security and privacy of 5G networks is crucial to protect users and maintain trust in the technology.

Software-defined 5G. Fortunately, the cellular network community has made tremendous progress over the years to make 5G more transparent and accessible to general users, by pushing the concept of *Software-defined 5G*. Specifically, it decouples the traditional radio hardware from the cellular software stacks running on top. Based on this concept, many open-source cellular software projects have been released, such as the leading OpenAirInterface (OAI) [12] and srsRAN [13] projects. End-users, such as educators, testers, and researchers, could download and deploy these software stacks to instantiate a standard-compliant 5G network on commodity machines and software-defined radios [14]. Moreover, to further address the hardware resource availability on the user side and facilitate large-scale experimentation, various 5G testbeds have been deployed and are open to the public for wireless research and testing. Notable examples are Powder [15], COSMOS [16], AERPAW [17], ARA [18], and Colosseum [19]. They support a wide range of 5G testing capabilities, such as massive Multiple-Input Multiple-Output (MIMO), ultra-high bandwidth, massive Internet-of-Things (IoT), and scenario-aware network deployment and emulation. The underlying driving force of these testbeds is also the software-defined 5G architecture, as it allows users to deploy any compatible software based on their hardware platforms, and thus provides huge flexibility.

Key Motivations. While these platforms and testbeds have made 5G truly accessible, there are still many challenges for less-experienced users to deploy 5G networks. The first

Workshop on Security and Privacy of Next-Generation Networks (FutureG) 2025
24 February 2025, San Diego, CA, USA
ISBN 979-8-9919276-7-3
https://dx.doi.org/10.14722/futureg.2025.23035
www.ndss-symposium.org

challenge is *deployment* due to the highly complex 5G architecture, as a functional 5G network typically involves many network functions deployed into the core network, radio access network, and user equipment. While open-source projects such as OAI already provide tutorials, users still need to go through complicated procedures including dependency installation, software compilation, network configurations, and execution. These are conducted in a command-line environment and may encounter machine-specific issues (e.g., IP configurations). The second challenge is *visual interaction*, as these platforms are typically executed through command line interfaces and thus lack visual interaction with end-users, such as tracking network status in real time and inspecting the internal logs. The third challenge is regarding the *security testing capabilities* as none of these tools come with such functions and have to rely on external frameworks. Related tools in this area, in particular open-source ones, have been scarce and are designed for technical-savvy users [20], [21].

The aforementioned opportunities and challenges render the necessity of a platform for entry-level users to easily instantiate 5G networks and perform security testing with rich visual interactions. Therefore, we present MOBIDOJO, the first graphic-based and virtual security combat platform for 5G. MOBIDOJO offers two key capabilities including (1) a web-based graphical panel for one-click deployment, control, and monitoring of a virtual 5G test network (i.e., no radio hardware required) deployed based on the OpenAirInterface (OAI) [22] project in a containerized environment, (2) security simulation plugins that enable users to graphically execute existing attack exploits [23] or create custom packet-level payloads (in *PCAP* format) for attack experimentation in an isolated 5G network environment [20]. The software-defined nature of MOBIDOJO makes it universally compatible with almost any commodity servers for either local or cloud-based deployment. To our knowledge, MOBIDOJO is the first platform that provides both visualization and security testing capabilities through a single pane of glass. Looking ahead, we anticipate MOBIDOJO will drive many valuable applications, from education and vulnerability testing to more sophisticated tasks such as 5G Capture-The-Flag (CTF) competitions, security dataset collection, and evaluation of defensive solutions. As a result, we will release MOBIDOJO open-source at https://github.com/5GSEC/MobiDojo.

Contribution. In summary, our paper makes the following contributions:

- Design of a novel and user-friendly 5G security combat platform MOBIDOJO utilizing a web-based interface and integrating open-source 5G software stacks in a containerized environment.
- Develop a visualization system in MOBIDOJO for interacting and monitoring 5G components in real-time.
- Integrate 5G security testing plugins in MOBIDOJO to enable execution of existing attacks as well as custom attack payload creation and testing in an isolated network environment.

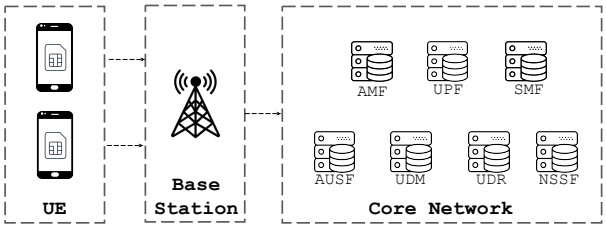

Fig. 1: Overview of 5G cellular network architecture.

II. BACKGROUND AND RELATED WORK

A. 5G Cellular Network

5G stands for the next-generation cellular network that aims for high-speed data transmission, low latency communication, and a large scale of connection. The composition of the 5G network is mainly divided into three parts: the user equipment (UE), the Radio Access Network (RAN), and the Core Network, as shown in Figure 1.

1) A UE generally refers to an end-user device that is used in mobile networks such as smartphones, tablets, and Internet of Things (IoT) devices. A UE is connected to a gNB through a wireless network and performs data transmission and communication.
2) The Radio Access Network (RAN) is also referred to as base stations or gNodeB (gNB) in a 5G context. The RAN is responsible for relaying wireless signals between the UE and the Core Network.
3) The Core Network in 5G, or the 5G core, is composed of various virtual network functions that are responsible for key functions such as data routing, authentication, and network resource management across the entire network, such as the Access and Mobility Management Function (AMF) that handles UE authentication. The 5G Core Network introduces new concepts like Network Slicing, which allows for the creation of virtual, independent networks on the same physical infrastructure, enabling the provision of customized networks tailored to meet different service requirements.

B. OpenAirInterface

OpenAirInterface (OAI) is an open-source software project designed for 5G and previous generations of cellular network technologies, widely used for experimentation and research purposes [12]. OAI has implemented the key components to instantiate 5G networks, including the UE, RAN, and the 5G core. It enables researchers and developers to build and test 5G networks in real-world environments. One of the most notable features of OAI is that it allows the implementation of 5G networks solely through software, without relying on commercial network equipment. This enables users to experiment with and validate new network functionalities, as well as analyze network architecture performance. Furthermore, OAI is compliant with the 3GPP standards, offering the advantage of constructing a network environment that adheres to actual industry standards.

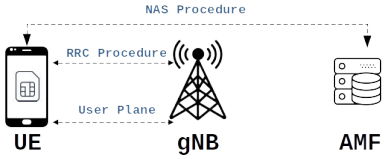

Fig. 2: Typical procedures for a 5G UE to authenticate and establish network connections.

C. 5G Security

5G networks have various inherent vulnerabilities originating from their standard design. This section discusses these vulnerabilities and the ways for adversaries to exploit them in practice. The typical procedure for a 5G UE to establish network connection is illustrated in Figure 2. During the 5G mutual authentication and key agreement process, encryption keys are distributed throughout the system as necessary. However, not all signaling messages have the same encryption requirements. For RRC (Radio Resource Control) [24] and NAS (Non-Access Stratum) [25] signaling messages, encryption is optional, and only integrity checking is mandatory [26]. Similarly, for user-plane traffic, both encryption and integrity checking are optional. This inconsistency in encryption requirements can potentially expose certain messages to unauthorized access or tampering.

Threat Model. There are different classes of adversaries who could exploit the aforementioned 5G vulnerabilities for malicious attacks, including: (1) Malicious UEs: With the availability of commercial off-the-shelf (COTS) software-defined radios (SDRs) running open-source cellular software and valid subscriber network identities (e.g., SIM cards), attackers can set up adversarial UEs. These rogue devices can be used to exploit vulnerabilities, eavesdrop on communications, or launch attacks on the network. (2) Man-in-the-Middle (MiTM) adversaries: A MiTM attacker can impersonate a legitimate base station to a victim UE and a legitimate UE to a victim base station. By exploiting messages that are not encrypted and digitally signed, a MiTM attacker can replay or modify messages in the network traffic. Tools like 5Greplay [20] and 5Ghoul [21] have been developed for researchers to reproduce and study such attacks. Similarly, practical adversaries can set up such a MiTM relay by using COTS SDRs and open-sourced software to compromise the security of 5G networks.

III. System Overview and Design

A. Overview

MobiDojo is the first graphic-based and virtual security combat platform for 5G networks. Its major implementation utilizes OpenAirInterface's pre-built docker containers to provide a visualized representation of a 5G network. The design of MobiDojo strives to achieve three major goals as summarized below.

One-click 5G Deployment. MobiDojo provides a web panel that provides common Graphical User Interface (GUI) widgets such as buttons for easy 5G network deployment. This interface provides intuitive visual interactions for users to easily deploy and manage a virtual 5G test network, without having to configure complex 5G-specific network configurations and operate command-line instructions.

Real-time Network Monitoring. The testbed visually displays the status of each 5G network component, such as, whether it is currently active or not. If a module is turned off due to an attack, the visualization promptly updates to reflect the change in status. Similarly, if a container shuts down due to internal issues, users can quickly and easily check its status. More detailed information, including logs and network traffic, can be monitored and saved locally for offline analysis.

Attack Simulation. MobiDojo has integrated attack simulation plugins that enable the reproduction and testing of various 5G-specific exploits safely within an isolated network environment. These simulations can be conducted entirely through GUI-based interactions. It also provides flexibility for users to create custom attack payloads in *pcap* format to emulate novel attack scenarios.

B. System Design

MobiDojo's visual layout is shown in Figure 3. At a high level, MobiDojo is composed of five main functional components described below in detail.

1) 5G Network Visualization: This panel visualizes the overall architecture of the deployed 5G network and the status of each component. It provides buttons to monitor and refresh the component status. For instance, when a user clicks the "Check Status" button, the system automatically updates each module of the UE, Base Station, and Core Network, displaying the status of each Docker container with green and red colors. Additionally, the system performs a connectivity check from the UE to an external network (e.g., google.com) by conducting a ping test. The result of this test is displayed in a pop-up alert window, allowing the user to quickly see whether the connection was successful. Through these status checks and data connection checks, users can confirm whether the 5G network has been deployed successfully. Furthermore, such visualization also facilitates the understanding of actual 5G network architectures for users who are new to this area.

2) Network Control Panel: The network control panel manages the status of each 5G module and contains essential implemented instructions through interacting with the docker containers. It utilizes 8 buttons to control the on/off states of 5G modules. These buttons allow users to easily deploy and shut down their 5G networks without the need to enter complex, separate Docker commands through the command-line terminal. Users can initiate live traffic capture by simply clicking the "Traffic Capture Core Network" button, which runs *TShark*, a Command-Line Interface (CLI) version of Wireshark. In this separate network setup through a Docker-Compose yaml file, the UE and gNB are configured to automatically capture traffic within the Docker container. We also have implemented a button that allows users to automatically

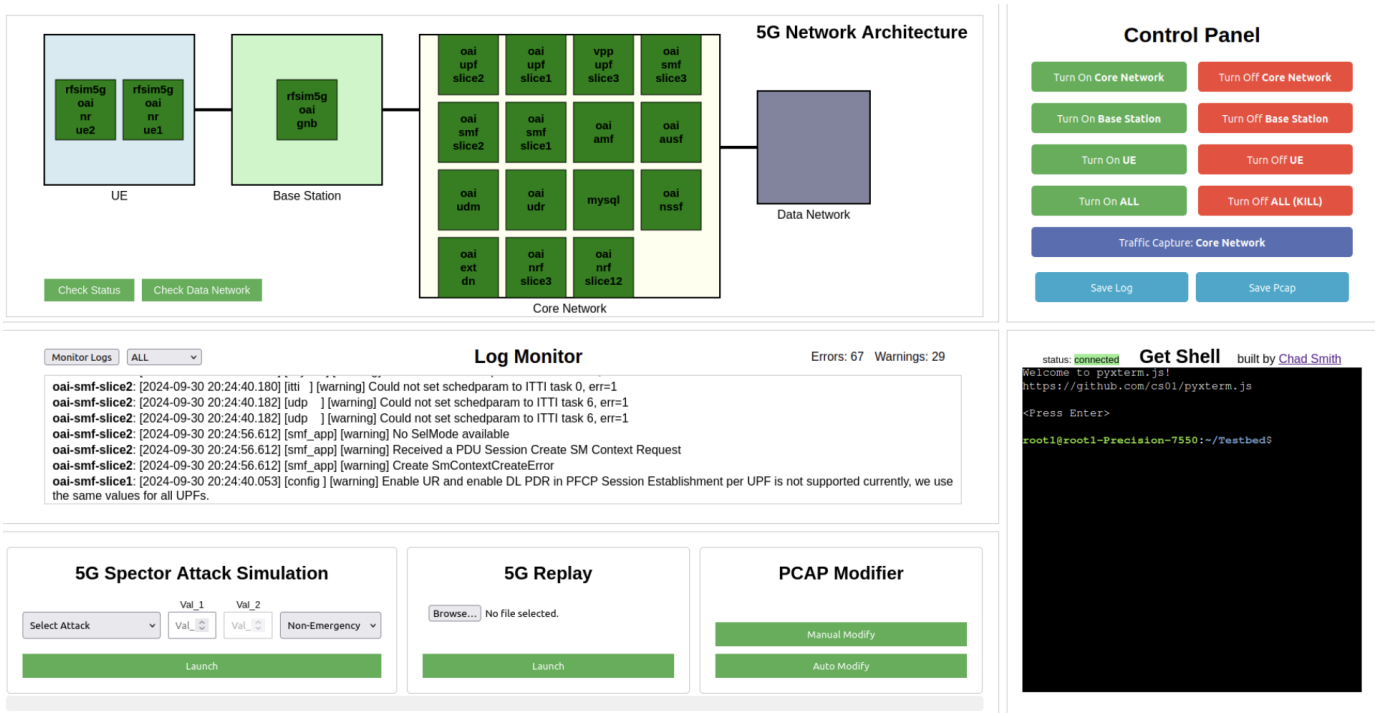

Fig. 3: The visual layout of MOBIDOJO with its five major components to control, interact, and monitor the 5G network as well as perform security exploit simulation.

download the *pcap* file of each UE and gNB and save all of the Docker logs to local storage.

3) Log Monitor: The log monitor parses and visualizes the logs generated by all 5G modules which are deployed as Docker containers. It counts the number of errors and warnings in the logs and displays them on the screen, providing users with capabilities to quickly check and diagnose network issues. Users can filter a specific container's log from a dropdown menu (the default option is ALL). By monitoring these log errors and warning messages, users can detect underlying anomalies in the 5G network.

4) Attack Simulation Plugins: There have been limited available tools and attack simulation frameworks to reproduce 5G-specific attacks. Fortunately, we still discover that some prior works have released their exploit implementation for attack reproduction and they are compatible with our implementation (e.g., OAI) [23], [20]. To this end, MOBIDOJO has integrated these attack tools as simulation plugins and provides two modes: (1) Simulation of pre-configured attacks and (2) Simulation of custom attacks through self-defined *pcap* payloads.

Pre-configured Attacks. This plugin is developed based on a prior work 5G-SPECTOR [23] where its repository has released 7 different types of 5G attacks of the layer three protocols, as listed in Table I. This attack suite involves a wide range of exploits from denial-of-service (DoS) [3], privacy leakage [7], [8], and network security downgrade [4], with tunable attack parameters to control the variance of attacks in each category. As a result, we developed a visual panel to allow users to select from these pre-configured attacks and execute them with self-defined parameters. For example, the BTS resource depletion attack [3] suite provides three different parameters and MOBIDOJO provides the corresponding GUI widgets to execute the attack. Specifically, *val_1* and *val_2* set the attack variance and the delay parameter in the millisecond level for each DoS session, and the dropdown menu configures whether the attack is launched in an emergency mode. The integration of the attack suite is also deployed seamlessly through modular Docker containers, as we compiled the tool as a standalone UE container so that it could be deployed into our OAI-based 5G network.

Self-defined Attacks. To facilitate users to create custom attack payloads and replay them into the network for attack experimentation, MOBIDOJO utilizes and integrates an open-sourced attack tool called *5Greplay* [20]. While open-source tools like Tcpreplay are also alternatives to address the challenge of replaying malicious traffic patterns on Intrusion Detection Systems (IDSs), their primary focus is on modifying attributes and fields related to IP, TCP, and UDP protocols. Similarly, other packet manipulation solutions, such as Scapy, are not specifically designed for 5G networks. To bridge this gap, 5Greplay is designed to facilitate fuzz testing of 5G network interfaces [20]. The main objective of 5Greplay is to simplify the testing process of 5G virtual network functions and IDSs. It achieves this by allowing users to forward network packets from one network interface card (NIC) to another, with the option to modify the packets if needed.

Attack	Adversary Type	Layer	Message Exploited	Attack Parameters	
BTS Resource Depletion	UE	RRC	ConnectionRequest	Attack Level (1 - 999)	Delay
Blind DoS	UE	RRC	ConnectionRequest	Attack Level (1 - 999)	TMSI
Downlink DoS	MiTM	NAS	AttachReject	Attack Level (1 - 999)	-
Downlink IMSI Extractor	MiTM	NAS	IdentityRequest	Attack Mode (1 - 4)	-
Uplink DoS	MiTM	NAS	AttachRequest	Attack Level (1 - 999)	-
Uplink IMSI Extractor	MiTM	NAS	AttachRequest	Attack Level (1 - 999)	-
Null Cipher	MiTM	RRC	SecurityModeFailure	Attack Mode (1 - 2)	-

TABLE I: List of MOBIDOJO's supported 5G simulated attacks implanted from the 5G-Spector attack suite [23].

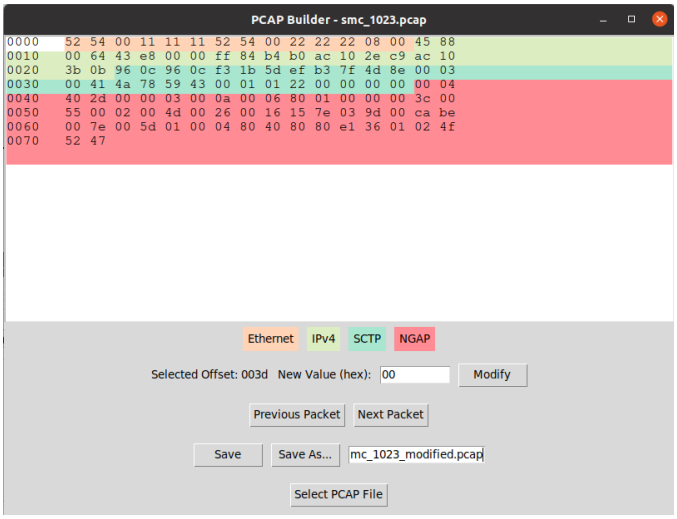

Fig. 4: Interface of the Manual PCAP Modifier.

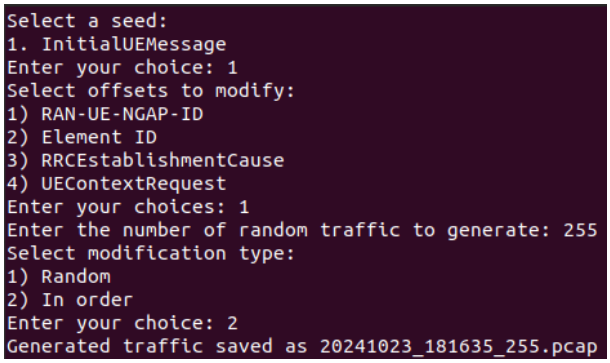

Fig. 5: Interface of the Auto PCAP Modifier.

PCAP Modifier. To complement 5Greplay, we developed a separate PCAP Modifier to allow the creation of custom exploitation payloads. We choose PCAP as the standard 5G payload format as it has been widely used for network traffic analysis and is compatible with numerous tools (e.g., the Wireshark traffic dissector). The PCAP Modifier consists of two distinct components: Manual Modifier and Auto Modifier. As shown in Figure 4, The Manual Modifier primarily focuses on the Next Generation Application Protocol (NGAP) and provides users with an intuitive interface to modify specific offsets within the packet. To enhance user convenience, the Ethernet, IPv4, Stream Control Transmission Protocol (SCTP), and NGAP sections are differentiated by color coding. The Manual Modifier allows users to easily modify packets by selecting a specific offset and entering a new hexadecimal value. The program then updates the packet with the user-provided value at the chosen offset. The accompanying figure illustrates how an NGAP traffic packet can be modified using this straightforward process.

On the other hand, the Auto Modifier leverages the comprehensive documentation [27] provided by 3GPP to define the internal structure of 5G packets. This preliminary component functions as a command-line interface (Figure 5) through prompt-based user interactions. By utilizing these well-defined mechanisms, users can effortlessly generate a wide range of packets. As a demonstration, we implemented the Auto Modifier functionality using the InitialUEMessage, which is one of the NGAP messages. According to the 3GPP NGAP document, the Message Type is defined by the Procedure-Code. The ProcedureCode ranges from 0 to 80, with the InitialUEMessage being assigned the value 15. In the section defining the InitialUEMessage, the document specifies both Mandatory and Optional fields that should be included in the message or not. The Mandatory fields for the InitialUEMessage are Message Type, RAN UE NGAP ID, NAS-PDU, User Location Information, and RRC Establishment Cause. These information elements are further defined in detail by the ProtocolIE-ID, which ranges from 0 to 438. A single message is created through a combination of these information elements. By leveraging this packet generation combination, we developed the Auto PCAP Modifier. This tool enables us to generate packets that are difficult to produce in a real environment, allowing for vulnerability and fuzzing tests to be conducted on the network. The Auto PCAP Modifier facilitates the creation of test scenarios that might otherwise be challenging to replicate, enhancing the robustness of network testing procedures.

5) Shell Terminal: While MOBIDOJO encapsulates 5G network operations into various GUI components, intermediate users may also need more fine-grained control over the deployed network for advanced operations. To provide this user-friendly environment, a separate terminal space has been integrated into MOBIDOJO to allow users to work on a single screen. To achieve this, we utilized a tool called *pyxterm* [28], which enables users to run a terminal emulator directly within their web browser. By clicking the "Get Shell" button, users can access the terminal on a large screen in a new tab, maximizing their ease of use and efficiency while interacting with the testbed.

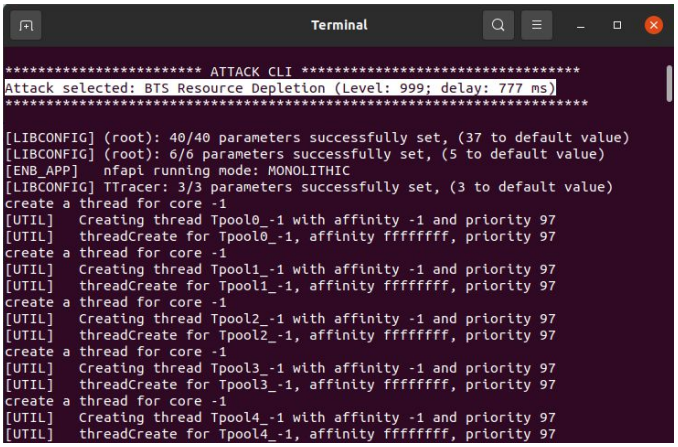

Fig. 6: Execution logs of the BTS Resource Depletion Attack in a separated terminal.

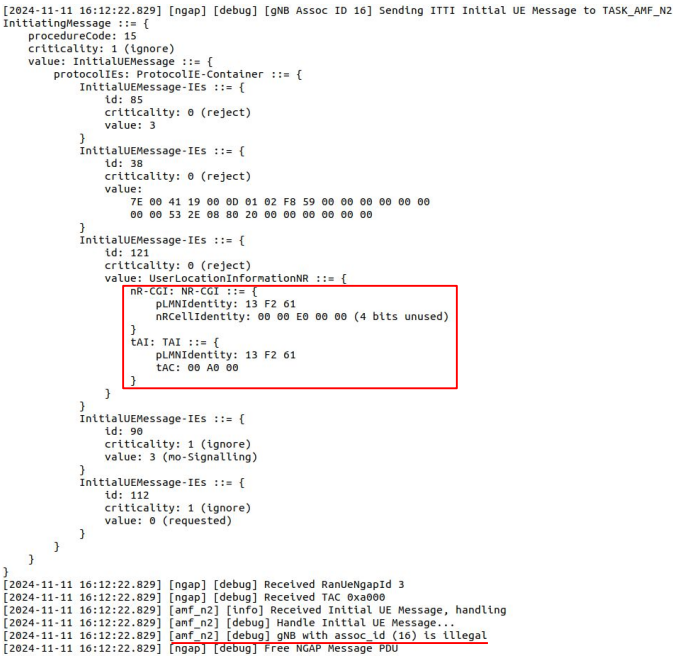

Fig. 7: Part of the AMF log where the InitialUEMessage was modified using PCAP Modifier and the packet was replayed to the 5G network with 5Greplay.

C. Experiment Examples

1) BTS Resource Depletion Attack and Log Analysis: In this example, we show how an attack is executed on MOBIDOJO and how to conduct a further detailed analysis of the attack. When a selected attack is initiated (in this case, BTS Resource Attack), a new pop-up terminal of Figure 6 appears. In this figure, the selected attack mode and input values are displayed as input. As a result of this attack, the gNB shuts down due to the DoS UE sessions, the two previously connected UEs are also disconnected. Next, by pressing the "Check Status" button, a green color fills the box if the modules are in a normal state, while red indicates an abnormal or disconnected state, allowing users to intuitively check the status of the modules.

After the attack is completed, users can download and analyze the gNB logs offline. The log monitor provides a log-saving button for this purpose. Upon reviewing the gNB logs, an "assertion fail" message is displayed, and the system shuts down due to its inability to process the RRCSetupRequest. Additionally, users may use the traffic capturing feature to save a copy of the attack session traffic locally to facilitate detailed analysis of the 5G packets.

2) Use of 5Greplay and Manual PCAP Modifier: In this example, we demonstrate how to use MOBIDOJO to create and test a novel 5G attack with 5greplay and a manual PCAP Modifier on MOBIDOJO. First, we select one NGAP message, in this case, InitialUEMessage was selected. After modifying some information through the manual pcap modifier, the PCAP file is replayed to the core network, especially AMF. With this InitialUEMessage, we modified the Public Land Mobile Network (PLMN) information. PLMN is identified by a globally unique PLMN code, which consists of a Mobile Country Code (MCC) and Mobile Network Code (MNC). As such, this is created through a combination of MCC and MNC. To modify it, we need to know which offset of the traffic is locating the information. For this, we can use a tool such as WireShark to locate the information. The next step is to manually change the value at the exact offset with the manual PCAP Modifier. Next, it is saved as a new PCAP file and replay it to the core network by using 5Greplay.

As shown in the logs in Figure 7, the modified PCAP traffic was correctly replayed to the AMF and was also confirmed by captured traffic and saved log. Errors or vulnerabilities were not identified in this example as we mentioned earlier. The highlighted log entries indicate the PLMN field of the message has been modified and sent to the AMF. Afterward, the underlined log shows an illegal gNB association ID was found and the AMF terminated this procedure because of the anomalous message. From this example, we show that MOBIDOJO could be used to conduct experimentation on malicious attack payloads or abnormal 5G protocol messages, in order to assess the 5G network's robustness against these potential threats in practice.

D. Implementation Details

MOBIDOJO is built using Flask, a lightweight and flexible web framework for Python. This framework is particularly popular for developing small to medium-sized applications, but it can also be scaled for larger projects. The testbed leverages Jinja2, a template engine that allows for the embedding of dynamic content within HTML pages. As we have mentioned, MOBIDOJO utilizes OAI Docker Containers for 5G deployment, which are orchestrated using Docker Compose. Through a predefined YAML file, two UE instances are assigned to two different network slices using a network slicing technique. Network slicing enables service providers to create customized, isolated slices within the same physical network

infrastructure. This approach ensures that each slice operates independently, with its own dedicated resources, policies, and security configurations. The isolation provided by network slicing allows service providers to meet the diverse regulatory or security standards required by various industries without compromising the integrity of other slices. Users can easily set up MOBIDOJO by executing a straightforward bash script to install the software dependencies.

IV. USE CASES

A. Deployment Strategies

1) Local Machines: Deploying MOBIDOJO on a local machine offers several advantages. Users have complete control over the environment, allowing for customization, integration with existing systems, and real-time debugging. This level of control is particularly beneficial for researchers who require specific configurations or need to test novel security mechanisms. Local deployment reduces dependency on external resources, making it suitable for scenarios where stable network access is not guaranteed, such as demonstrations or workshops. Furthermore, this enables the integration of Software-Defined Radio (SDR) equipment directly with the testbed. SDRs [14] allow researchers to experiment with 5G networks in a more realistic radio frequency (RF) environment. By integrating SDRs, users can test the impact of physical layer vulnerabilities, evaluate the effectiveness of security measures against RF-based attacks, and develop novel physical layer security solutions.

However, local deployment may be limited by the hardware resources available on the user's machine. Running MOBIDOJO can be resource-intensive, especially when simulating large-scale networks or handling multiple concurrent users. Scalability can also be a challenge with local deployment, as the performance may be constrained by the capabilities of a local machine.

2) Cloud Servers: Deploying MOBIDOJO on a cloud platform offers benefits in terms of scalability, accessibility, and collaboration. Cloud servers provide virtually unlimited resources, allowing users to scale MOBIDOJO based on their requirements. This is particularly advantageous when simulating large-scale 5G networks, handling high volumes of traffic, or accommodating a large number of concurrent users. Cloud platforms enable elastic scaling, allowing users to dynamically adjust resources based on demand. This also improves accessibility by allowing remote access to the testbed from anywhere with an internet connection. This feature is necessary for distributed teams, collaborative research projects, or educational programs where participants are geographically dispersed. Users can easily share access, work together on experiments, and analyze results in real time, fostering a collaborative learning environment.

However, cloud deployment introduces a dependency on network connectivity and incurs costs based on resource consumption. Users need stable internet access to interact with the testbed hosted on a cloud server and must be mindful of the pricing model associated with their testing scenarios. Additionally, some organizations may have data privacy or regulatory concerns when hosting sensitive information on third-party cloud platforms.

3) Hybrid (Local Machines and Cloud Servers): Another deployment option is a hybrid approach that combines local machine and cloud server deployments. In this scenario, the core components of the testbed, such as the visualization system and attack simulation modules, can be deployed on a cloud server. This allows for centralized management, easy access for multiple users, and the ability to handle resource-intensive tasks. At the same time, users can deploy certain components, like the 5G network modules (UE, gNB, Core Network), on their local machines. This hybrid approach enables users to take advantage of local and cloud deployments. They can have control over the 5G network components on their local machines while utilizing the scalability and accessibility of the cloud for other testbed features. This deployment strategy offers flexibility and can cater to diverse user requirements and preferences.

B. Applications

1) CTF (Capture the Flag) Competitions: CTF competitions are cybersecurity events where participants compete in challenges to test and showcase their skills in various domains, such as cryptography, reverse engineering, and network security. MOBIDOJO can be an invaluable asset for CTF organizers looking to create realistic and engaging challenges focused on 5G networks. By leveraging MOBIDOJO's capabilities, organizers can design simulated 5G network environments that allow participants to hands-on experience with identifying and exploiting vulnerabilities, analyzing network traffic and logs for signs of malicious activity, and developing and deploying custom security measures to protect against specific attack vectors.

The open-source nature provides flexibility for organizers to customize the environment according to their specific challenge requirements. They can modify the network topology, configure security settings, and even attack scenarios to align with the competition's objectives. Furthermore, MOBIDOJO's modular architecture enables participants to contribute improvements and extensions, fostering a collaborative learning environment and encouraging active engagement with the system. The hands-on nature of the challenges promotes a deeper understanding of 5G security concepts and encourages participants to think critically and creatively when approaching complex problems.

2) Education: MOBIDOJO's user-friendly interface and intuitive design make it an ideal platform for educational purposes, enabling students and instructors to easily visualize, build, and experiment with 5G networks in a controlled environment. It can be seamlessly integrated into university courses or training programs, providing students with practical experience in understanding 5G network architecture, functionality, and security. For instance, students can leverage MOBIDOJO to set up different network configurations, simulate various attack scenarios, and implement detection mechanisms. By

incorporating both attack and detection features, it allows students to gain an understanding of the functionality and security aspects of 5G networks. Moreover, the open-source nature of the system encourages collaboration and knowledge sharing among students and instructors, fostering a dynamic learning environment. By incorporating MOBIDOJO into educational programs, institutions can effectively prepare the next generation of cybersecurity professionals to tackle the unique challenges of securing 5G networks.

3) Security Research: Researchers can use MOBIDOJO to simulate and analyze different 5G network configurations, study the effects of various attacks, and develop and test detection mechanisms. The system's ability to visualize network traffic and monitor security breaches provides valuable insights for advancing 5G security research. Researchers can collaborate by sharing configurations, attack scenarios, and detection mechanisms, fostering a community-driven approach to improving 5G network security. The open-source nature allows researchers to customize and extend its functionality to suit their specific research requirements, promoting innovation and advancement in the field. In particular, MOBIDOJO could be integrated with existing 5G fuzzing frameworks to discover protocol-level vulnerabilities in an automatic way [29], [30].

4) Defense Evaluation: The security combat capabilities of MOBIDOJO enable the evaluation of various defenses, such as tools for intrusion and anomaly detection [23], [31], [32], [33], [34]. The evaluation is conducted within a controlled and isolated environment for preliminary experimentation before testing the defense out in a practical environment. In this context, MOBIDOJO's attack simulation plugins will enable researchers and developers to reproduce attacks in the 5G test network, and then deploy the corresponding defenses to evaluate their performance, e.g., the accuracy of detection. The visualized components help monitor the network status to efficiently inspect the outcomes of the attacks and defenses in repeated experiments.

V. DISCUSSION AND FUTURE WORK

Our approach shows the potential of a user-friendly and comprehensive 5G security platform. However, there are still limitations and areas for future improvements.

Attack Suite Improvement. First, expanding the range of supported security scenarios and attack simulations would enhance MOBIDOJO's comprehensiveness. Currently, MOBIDOJO's attack simulation mainly tackles packets of the 5G NGAP interface between the gNB and the AMF in the 5G core. This could be extended to support many other standard interfaces such as the F1 interface between gNB's network functions [35]. Specification information such as message definitions and semantics should be integrated to improve the usability of the PCAP modifier tool, allowing users to identify interesting protocol fields to generate meaningful packet payloads.

Integration of SDR and 5G Testbeds. Another area for future exploration is the integration of software-defined radio (SDR) capabilities, as many open-sourced 5G software supports a wide range of SDR models. This enables researchers to test 5G networks in realistic radio frequency (RF) environments and commercial UE devices. MOBIDOJO could also be integrated with public 5G testbeds such as Powder, COSMOS, and Colosseum [36], [16], [19] to utilize their hardware platforms as its 5G network backend. This would address the limitations of simulated-based environments and facilitate the experiments and attack simulations at lower cellular protocol stacks such as RF and physical layer.

Extensions for O-RAN. 5G and future generation cellular networks are moving towards a disaggregated and software-defined architecture called Open Radio Access Network (O-RAN) [37] where near-real-time or non-real-time network controllers (RICs) are deployed to monitor and control network behaviors with modular xApps, particularly for security applications [38], [39]. Future versions of MOBIDOJO could be extended to visualize O-RAN related components such as the RIC services and xApps, and meanwhile provide corresponding monitor and control functions to instantiate and interact with xApps. Attacks against O-RAN's standard interfaces, such as E2, could also be thoroughly studied by an extended attack simulation plugin dedicated to O-RAN.

VI. CONCLUSION

We presented MOBIDOJO, a novel virtual security combat and testing platform for 5G cellular networks. MOBIDOJO utilizes a user-friendly web-based interface and pre-configured Docker containers, enabling users to focus on understanding 5G concepts and functions without being overwhelmed by complex setups. The testbed offers easy-to-use features and supports various use cases, making it suitable for both educators and researchers. In addition, it provides a variety of functions, including real-time visualization of the status of the 5G module, simplified network configuration through intuitive graphic-based monitoring and controls, comprehensive 5G security training with attack simulations, and a virtual environment that bridges the gap between theory and practice. This development is anticipated to provide opportunities for exploring a wide range of attack and defense scenarios, as well as uncovering vulnerabilities in cellular network standards and implementations. Looking ahead, as 5G and future networks continue to be deployed worldwide, MOBIDOJO offers a valuable tool for education and research, paving the way for a better understanding and mitigation of 5G security risks.

ACKNOWLEDGEMENTS

We thank the anonymous reviewers for their constructive feedback. This research was supported in part by NSF awards ITE-2226443, ITE-2326882 and CNS-2112471. Any opinions, findings, conclusions, or recommendations expressed are those of the authors and not necessarily of the NSF.

REFERENCES

[1] J. Zhang, W. Xie, and F. Yang, "An architecture for 5g mobile network based on sdn and nfv," 2015.

[2] 3GPP, "Security architecture and procedures for 5g system," https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3169, 2024.

[3] H. Kim, J. Lee, E. Lee, and Y. Kim, "Touching the untouchables: Dynamic security analysis of the lte control plane," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 1153–1168.

[4] S. R. Hussain, M. Echeverria, I. Karim, O. Chowdhury, and E. Bertino, "5greasoner: A property-directed security and privacy analysis framework for 5g cellular network protocol," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 669–684.

[5] G. Lee, J. Lee, J. Lee, Y. Im, M. Hollingsworth, E. Wustrow, D. Grunwald, and S. Ha, "This is your president speaking: Spoofing alerts in 4g lte networks," in *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, 2019, pp. 404–416.

[6] E. Bitsikas and C. Pöpper, "You have been warned: Abusing 5g's warning and emergency systems," in *Proceedings of the 38th Annual Computer Security Applications Conference*, 2022, pp. 561–575.

[7] M. Kotuliak, S. Erni, P. Leu, M. Roeschlin, and S. Čapkun, "Ltrack: Stealthy tracking of mobile phones in lte," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1291–1306.

[8] S. Erni, M. Kotuliak, P. Leu, M. Röschlin, and S. Capkun, "Adaptover: adaptive overshadowing attacks in cellular networks," in *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, 2022, pp. 743–755.

[9] M. Chlosta, D. Rupprecht, C. Pöpper, and T. Holz, "5g suci-catchers: Still catching them all?" 2021.

[10] A. Dabrowski, N. Pianta, T. Klepp, M. Mulazzani, and E. Weippl, "Imsi-catch me if you can: Imsi-catcher-catchers," in *Proceedings of the 30th annual computer security applications Conference*, 2014, pp. 246–255.

[11] S. Fonyi, "Overview of 5g security and vulnerabilities," *The Cyber Defense Review*, vol. 5, no. 1, pp. 117–134, 2020.

[12] "oai / openairinterface5g," https://gitlab.eurecom.fr/oai/openairinterface5g, May 2024.

[13] I. Gomez-Miguelez, A. Garcia-Saavedra, P. D. Sutton, P. Serrano, C. Cano, and D. J. Leith, "srslte: An open-source platform for lte evolution and experimentation," in *Proceedings of the Tenth ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation, and Characterization*, 2016, pp. 25–32.

[14] "Usrp software defined radio (sdr)," https://www.ettus.com/products/, May 2024.

[15] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, M. Hibler, D. Johnson, S. K. Kasera, E. Lewis, D. Maas *et al.*, "Powder: Platform for open wireless data-driven experimental research," in *Proceedings of the 14th International Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization*, 2020, pp. 17–24.

[16] D. Raychaudhuri, I. Seskar, G. Zussman, T. Korakis, D. Kilper, T. Chen, J. Kolodziejski, M. Sherman, Z. Kostic, X. Gu *et al.*, "Challenge: Cosmos: A city-scale programmable testbed for experimentation with advanced wireless," in *Proceedings of the 26th annual international conference on mobile computing and networking*, 2020, pp. 1–13.

[17] V. Marojevic, I. Guvenc, R. Dutta, M. L. Sichitiu, and B. A. Floyd, "Advanced wireless for unmanned aerial systems: 5g standardization, research challenges, and aerpaw architecture," *IEEE Vehicular Technology Magazine*, vol. 15, no. 2, pp. 22–30, 2020.

[18] H. Zhang, Y. Guan, A. Kamal, D. Qiao, M. Zheng, A. Arora, O. Boyraz, B. Cox, T. Daniels, M. Darr *et al.*, "Ara: A wireless living lab vision for smart and connected rural communities," in *Proceedings of the 15th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization*, 2022, pp. 9–16.

[19] L. Bonati, P. Johari, M. Polese, S. D'Oro, S. Mohanti, M. Tehrani-Moayyed, D. Villa, S. Shrivastava, C. Tassie, K. Yoder *et al.*, "Colosseum: Large-scale wireless experimentation through hardware-in-the-loop network emulation," in *2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2021, pp. 105–113.

[20] Z. Salazar, H. N. Nguyen, W. Mallouli, A. R. Cavalli, and E. Montes de Oca, "5greplay: A 5g network traffic fuzzer-application to attack injection," in *Proceedings of the 16th International Conference on Availability, Reliability and Security*, 2021, pp. 1–8.

[21] asset group, "5ghoul-5g-nr-attacks," https://github.com/asset-group/5ghoul-5g-nr-attacks.

[22] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "Openairinterface: A flexible platform for 5g research," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 33–38, 2014.

[23] H. Wen, P. Porras, V. Yegneswaran, A. Gehani, and Z. Lin, "5g-spector: An o-ran compliant layer-3 cellular attack detection service," in *Proceedings of the 31st Annual Network and Distributed System Security Symposium (NDSS'24)*, San Diego, CA, February 2024.

[24] 3GPP, "Radio resource control (rrc)," http://www.3gpp.org/DynaReport/38331.htm, May 2024.

[25] ——, "Non-access-stratum (nas) protocol for evolved packet system (eps)," http://www.3gpp.org/DynaReport/24301.htm, May 2024.

[26] F. B. Wala and M. Kiran, "5g network security practices: An overview and survey," *arXiv preprint arXiv:2401.14350*, 2024.

[27] 3GPP, "Ng-ran; ng application protocol (ngap)," http://www.3gpp.org/DynaReport/38413.htm, May 2024.

[28] cs01, "pyxtermjs," https://github.com/cs01/pyxtermjs.

[29] M. E. Garbelini, Z. Shang, S. Chattopadhyay, S. Sun, and E. Kurniawan, "Towards automated fuzzing of 4g/5g protocol implementations over the air," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 86–92.

[30] N. Bennett, W. Zhu, B. Simon, R. Kennedy, W. Enck, P. Traynor, and K. R. Butler, "Ransacked: A domain-informed approach for fuzzing lte and 5g ran-core interfaces," in *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 2024, pp. 2027–2041.

[31] M. Echeverria, Z. Ahmed, B. Wang, M. Fareed Arif, S. Rafiul Hussain, and O. Chowdhury, "Phoenix: Device-centric cellular network protocol monitoring using runtime verification," *arXiv e-prints*, pp. arXiv–2101, 2021.

[32] Z. Tan, J. Zhao, B. Ding, and S. Lu, "Celldam: User-space, rootless detection and mitigation for 5g data plane," in *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, 2023, pp. 1601–1619.

[33] J.-H. Huang, S.-M. Cheng, R. Kaliski, and C.-F. Hung, "Developing xapps for rogue base station detection in sdr-enabled o-ran," in *IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2023, pp. 1–6.

[34] A. Scalingi, S. D'Oro, F. Restuccia, T. Melodia, D. Giustiniano *et al.*, "Det-ran: Data-driven cross-layer real-time attack detection in 5g open rans," in *IEEE International Conference on Computer Communications*, 2024, pp. 1–10.

[35] 3GPP, "Ng-ran f1 application protocol (f1ap)," http://www.3gpp.org/DynaReport/38473.htm, May 2024.

[36] J. Breen, A. Buffmire, J. Duerig, K. Dutt, E. Eide, M. Hibler, D. Johnson, S. K. Kasera, E. Lewis, D. Maas *et al.*, "Powder: Platform for open wireless data-driven experimental research," in *Proceedings of the 14th International Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization*, 2020, pp. 17–24.

[37] "O-ran alliance," https://www.o-ran.org/, May 2024.

[38] H. Wen, P. Porras, V. Yegneswaran, and Z. Lin, "A fine-grained telemetry stream for security services in 5g open radio access networks," in *Proceedings of the 1st International Workshop on Emerging Topics in Wireless*, 2022, pp. 18–23.

[39] H. Wen, P. Sharma, V. Yegneswaran, P. Porras, A. Gehani, and Z. Lin, "6g-xsec: Explainable edge security for emerging openran architectures," in *Proceedings of the 23rd ACM Workshop on Hot Topics in Networks*, 2024, pp. 77–85.