SECURER: User-Centric Cybersecurity Testing Framework for IoT System

Tauheed Waheed[a], Eda Marchetti[b] and Antonello Calabrò[c]
CNR-ISTI Pisa, Italy

Keywords: User-Centric, Cybersecurity, Testing, IoT Systems.

Abstract: The rapid advancement of IoT systems and their interconnected nature highlight the critical need for strong cybersecurity measures. The SECURER framework is designed to offer a user-centric, adaptable, and comprehensive approach to cybersecurity testing. It aims to strengthen the security of IoT systems by prioritizing user behavior, aligning with evolving cyber threats, utilizing existing test suites, and ensuring regulatory compliance. We seek to showcase the functionality and applications of our user-centric cybersecurity testing framework (SECURER) by outlining a practical testing methodology to counteract cyber threats targeting the rolling code technology used in the automotive industry.

1 INTRODUCTION

Cybersecurity has posed significant challenges for emerging technologies like IoT in recent years, requiring a strategic approach to protect business infrastructure and continuity. These technologies can potentially replace existing systems, products, or services, creating new industry leaders while leaving established players behind if they fail to adapt. Cybersecurity is crucial for building secure systems and protecting users' sensitive information, especially in light of recent challenges in IoT security and privacy benchmarks. Integrating connected devices into modern households has redefined how we interact with our environment (Sáez-de Cámara et al., 2023), offering convenience and efficiency but also creating potential vulnerabilities to cyber-attacks (Heiding et al., 2023). Unauthorized access to connected devices could result in theft, vandalism, or harm to occupants. The realization that cybersecurity is not just about protecting systems but also about safeguarding the trust and privacy of individuals has led to a broader approach to conceptualizing and implementing security measures where individuals should be at the center.

One of the most effective means for increasing the cybersecurity level and identifying vulnerabilities, ensuring compliance, and maintaining the integrity of systems and data is testing (Matheu-García et al., 2019). Testing activities have several peculiarities: i) help discover security weaknesses and vulnerabilities within a system, application, or network, ii) ensure the functioning and efficiency of the different IoT components and devices, iii) validate that IoT systems are effectively implemented and accessible from threats, iv) assess the regulatory and legal requirements and the compliance with standards (such as PCI-DSS[1], HIPAA[2], and GDPR[3]) protect against security issues, data breaches or unauthorized access to sensitive data, vi) contributes to faster detection, containment, and recovery from security incidents, minimizing potential damage and downtime.

To address these challenges, we conceived the conceptualization and preliminary implementation of a user-centric cybersecurity testing framework called SECURER (User-Centric Cybersecurity Testing Framework for IoT systems) applicable to various application domains. In particular, SECURER has been conceived considering the following needs:
Providing a User-Centric Testing Environment: Many traditional cybersecurity frameworks prioritize technical aspects without considering the user's perspective and behavior. On the other hand, SECURER emphasizes user-centric security testing to ensure that security solutions are technically sound and aligned

[a] https://orcid.org/0009-0006-0489-7697
[b] https://orcid.org/0000-0003-4223-8036
[c] https://orcid.org/0000-0001-5502-303X

[1] https://www.pcisecuritystandards.org/
[2] https://www.hhs.gov/hipaa/index.html
[3] https://eur-lex.europa.eu/legal

Waheed, T., Marchetti, E. and Calabrò, A.
SECURER: User-Centric Cybersecurity Testing Framework for IoT System.
DOI: 10.5220/0013062900003825

In *Proceedings of the 20th International Conference on Web Information Systems and Technologies (WEBIST 2024)*, pages 393-400
ISBN: 978-989-758-718-4; ISSN: 2184-3252

with user behaviors, preferences, and usability requirements. This approach ultimately leads to better adoption and effectiveness.

Providing a Generic Testing Environment: IoT systems are highly heterogeneous, involving various devices with different capabilities, communication protocols, and security vulnerabilities. SECURER aims to address this complexity by providing comprehensive and tailored security assessments considering IoT ecosystems' diverse nature.

Aligned with the Rapid Evolution of Cyber Threats: SECURER aims to provide a proactive cybersecurity testing framework that can identify and mitigate potential vulnerabilities before they are exploited, enhancing the overall security posture of IoT systems.

Leveraging Existing Test Suite: SECURER can utilize pre-built test suites instead of starting from scratch. It can integrate, modify, and enhance them with additional test cases to efficiently evaluate system security. This approach saves time and resources, enabling a more targeted testing process.

Improving Regulatory and Compliance: SECURER offers a structured approach to testing and validating IoT security, helping organizations meet regulatory demands and build trust with users and stakeholders.

Leveraging the Knowledge About Cybersecurity: SECURER process offers a comprehensive approach to understanding cybersecurity risks and testing efficiency. It incorporates technical safeguards, awareness, education, and an ethical framework for handling and protecting data.

In this paper, Section 2 presents the motivations, Section 3 discusses current state-of-the-art, and Section 4 presents the architecture and behavioral model for SECURER. Its preliminary implementation and a showcase example are presented in Section5 and Section 6, respectively. Section 7 concludes the paper.

2 SECURER MOTIVATION

The current motivations inspire SECURER:

Protection Against Vulnerabilities: Therefore, cybersecurity testing is crucial for carefully identifying and addressing these vulnerabilities before they are exploited, ensuring the security and integrity of IoT systems and devices.

Maintaining User Trust: Any security breach or failure within the IoT ecosystem can significantly lose public trust (Huang et al., 2023) in these technological advancements. It is vital to enforce robust cybersecurity measures to uphold the safety and dependability of these technologies and maintain the trust of users and stakeholders.

Regulatory Compliance: This creates complex challenges for businesses seeking to navigate the landscape of regulations. Cybersecurity testing helps in regulation compliance, avoiding potential reputational damage, and legal penalties for noncompliance.

Enabling Innovation: Through the assurance of robust cybersecurity measures, businesses, and individuals can feel more confident in their innovation endeavors, accelerating innovation. It is primarily due to the reduction of concerns related to intellectual property theft and other potential cyber threats, which often hinder the free exchange of ideas and progress in technological advancement.

Business Continuity: In a cyberattack, the impact can be far-reaching, disrupting services, causing irrecoverable financial losses, and tarnishing the company's reputation. Comprehensive cybersecurity testing is crucial to establish robust defense mechanisms. By conducting extensive testing, businesses can achieve proactive cyber-secure systems, ensuring uninterrupted operations even when confronted with cyber threats.

Protecting User Data: Given the vast amount of data these devices handle, cybersecurity testing is paramount to identify and mitigate potential vulnerabilities, protect against unauthorized access or data breaches, and uphold user information's integrity, privacy, and security. Conducting these mentioned IoT devices thorough and regular cybersecurity testing helps ensure that IoT devices have the potential to safeguard sensitive data and maintain user trust in the ecosystem of connected devices. Furthermore, we have explored the current solutions and frameworks to understand the recent utilization of cybersecurity testing to achieve more secure architectures.

Leveraging Human-Centric Testing: Integrating user behavior into cybersecurity strategies can enhance testing effectiveness, making efforts more resilient against cyber attacks. A comprehensive testing framework is necessary due to the evolving complexity of cyber threats. Prioritizing the user aspect fosters a comprehensive security culture, enhancing awareness and vigilance against potential threats.

3 RELATED WORK

Without pretending to be exhaustive, the section positions the SECURER proposal inside the state of the art and overviews the currently available testing framework and ongoing cybersecurity strategy.

Multi-Access Edge Computing (MEC) (Pietrantuono et al., 2023) is a computing framework designed to bring computation and data storage closer to the user, providing low-latency access to applications and efficiently utilizing network resources. MEC facilitates seamless integration of emerging Internet of Things (IoT) applications, enhancing scalability, reliability, and real-time responsiveness, particularly in addressing cybersecurity challenges and enhancing IoT application resilience against cyber attacks.

A framework for identifying botnets has been proposed, which involves Mirai botnet (Ali et al., 2024) and additional red-teaming tools that perform denial-of-service (DoS) attacks, scanning activities, and targeting of Internet of Things (IoT) protocols. The multifunctional testbed (Sáez-de Cámara et al., 2023) serves as a cyber range for simulating cyber attacks, testing and validating security solutions, and gathering network and application data to generate comprehensive datasets for analysis and research purposes.

Recent studies (Heiding et al., 2023) emphasize the need for more extensive vulnerability assessments in penetration testing. Researchers conducted thorough cybersecurity tests on 22 devices across five categories in connected homes: intelligent door locks, smart cameras, smart car adapters/garages, smart appliances, and miscellaneous smart home devices.

Traditional security measures often allow communications before appropriate authentication, presenting vulnerabilities that malicious entities can exploit. The solution in (Puthal et al., 2020) aims to authenticate users and devices before communication and establish a secure channel, enhancing security and network performance in IoT and distributed Edge data center infrastructures.

Researchers have proposed DecisionTSec(Puthal et al., 2022), an innovative security approach utilizing decision trees to ensure communication security in IoT networks. The method incorporates edge data centers at the network's periphery to enhance security measures. Practical experiments on a real-time testbed have validated its effectiveness and theoretical robustness in assessing system performance and security integrity.

Through a case study, the researchers (Barhamgi et al., 2018) examine the requirements future data collection frameworks in these systems must meet to offer substantial privacy protection for users through user-centric privacy engineering strategies. The researchers (Rivadeneira et al., 2023) focus on user-centric privacy preservation models within an IoT context, highlighting unresolved issues and research opportunities. Moreover, the creation of frameworks that position the user at the heart of data management ensures they have complete oversight and decision-making power over how their data should be utilized while prioritizing safeguarding their privacy.

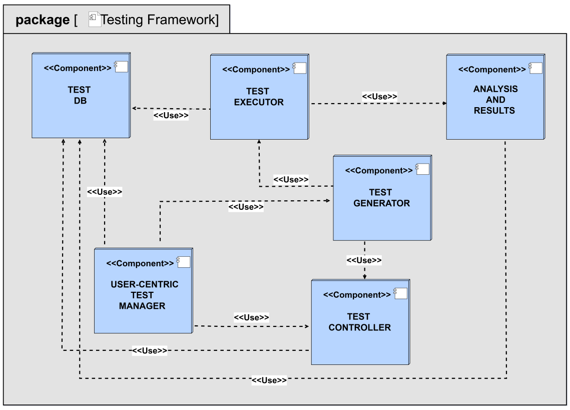

Figure 1: SECURER Architecture.

The current strategies do not prioritize user-centric security and privacy solutions or simultaneous testing. Establishing a comprehensive framework that enhances cybersecurity through user-centric testing is crucial for the responsible and effective implementation of IoT. This testing process helps protect against potential breaches and ensures that technologies are secure, reliable, and compliant with regulatory standards to maintain user trust.

We need modern solutions to assess cybersecurity from the user's perspective. Currently, there are no user-centric cybersecurity testing frameworks available. We need one, like SECURER, to identify vulnerabilities in IoT systems and protect against evolving cyber threats. Our framework aims to make strong security solutions accessible and understandable to non-expert users, increasing their adoption and proper use.

4 SECURER ARCHITECUTURE

The section describes the SECURER architecture to provide an effective framework for shifting from traditional reactive quality control methods to a more proactive and collaborative system. SECURER can help businesses optimize their operations while mitigating cybersecurity risks, enhancing their reputation, and building user trustworthiness for industrial and commercial IoT systems. SECURER targets the challenges presented in Section 2 and proposes a methodology for integrating cutting-edge technologies like IoT while addressing cybersecurity challenges and improving quality. In the next sections, the overall description of the SECURER architecture and its behavioral model will be described.

4.1 Framework Conceptualization

SECURER is a user-centric testing framework that emphasizes the shared responsibility of security. It highlights resilience and recovery strategies and consists of the components shown in Figure 1.

User-centric Test Manager (UCTM): a user-centric test manager gathers comprehensive user requirements, aiming to pinpoint specific and detailed testing objectives. It will be achieved using questionnaires designed to clarify and bridge the gap between what users expect and their testing needs. The component also manages the testing feedback and the different decisions during the testing activity execution. All the user interactions are performed through a dedicated UI/GUI.

Test Generator: it manages the selection and generation of test cases. The test cases are provided considering the specific user's requirements and needs. According to the different conditions, the component can provide testing strategies and help the user select the best one. A selection of the most proficient in identifying and effectively mitigating potential security risks will be provided among the generated test cases. Executing test cases will guarantee the seamless availability, confidentiality, and overall cybersecurity of crucial information within an IoT system.

Test Executor: According to available testing libraries it finalizes the test cases for the application's domain-specific execution environment (provided by the user or retrieved through open-source proposals).

Test Controller: It provides a collection of test strategies and pre-defined test cases, if available. It helps create questionnaires for user-centric test managers and ensures test cases meet the outlined requirements. Additionally, it uses Generative AI to create test case instances and expand test suites to maintain quality standards in response to evolving cybersecurity threats.

Analysis and Results: It helps understand the impact of vulnerabilities and identify redundant paths in the source code of application domains or connected IoT devices and third-party libraries or APIs (Application Programming Interface). The analysis is primarily conducted through path coverage testing. Additionally, this component's main responsibility is to analyze the test results generated by SECURER and produce test reports for users via the UI/GUI of our user-centred test manager.

Test DB: It collects the knowledge and results generated during the testing activity. Indeed, the component provides adequate resources to other components (like GitHub) and facilities for constructing and designing questionnaires and datasets of pre-defined test cases. Furthermore, it stores and maintains user feedback and test reports, which contributes to designing a more effective questionnaire for the next iteration and improves the overall testing process.

SECURER's architecture is designed for the Internet of Vehicles (IoV), cobots, drones, and IoT. The user-focused cybersecurity approach allows for creating specific testing scenarios based on user requirements, simulating malicious behavior and system vulnerabilities while encouraging users to identify and reduce risks, ultimately increasing their knowledge of cybersecurity and the testing process. Cybersecurity testing is evolving through SECURER to address the risks of interconnected IoT systems. It aims to create an ecosystem where security is a shared responsibility, emphasizing resilience and recovery strategies. Involving users in testing helps them understand their behaviors and expectations for more secure architectures.

4.2 SECURER Behavioral Model

The goal of SECURER, as depicted through the behavioral model, is to prioritize user needs and contribute to developing secure, high-quality systems that enhance the future commercial potential of IoT while strengthening cybersecurity to foster trust among stakeholders. Therefore, this section in Figure 2 describes the SECURER behavioural model, illustrating how users can interact with SECURER to conduct cybersecurity testing by following several steps:

UTM GUI: Is the GUI of UCTM (User-centric Testing Manager)the user first must register and correctly log in to UTM (User Testing Manager) to perform cybersecurity testing and express his requirements, testing environment, and domain application. Moreover, UCTM can utilize this information to prepare an adhoc questionnaire.

Preparing Questionnaire: To prepare the questionnaire, the UCTM requests collaboration from Test DB, as it comprises predefined test cases or information that could be utilized to leverage the questionnaire.

Finalizing Questionnaire: UCTM displays the questionnaire to the users to let them fill it out. User and provides more concrete details useful for the specification of the test cases during the test case generation. For instance, the user specifies which components in the IoV (Internet of Vehicles) system should be tested against replay attacks.

Extracting User Specifications: The user completes the questionnaire and sends it back to UCTM; at this point, internal interaction tends to start within SECURER, and then UCTM extracts user specifications from the questionnaire results. Further, the UCTM requests the Test Controller to prepare a set of test cases according to SECURER requirements provided by the user in the previous step.

Generating Test-Cases: As per the request from UCTM, the Test Controller starts exploring a suitable test suite and test strategy satisfying the request received. In the case of predefined test suite availability, the Test Controller provides the features for refining or updating it according to the test specifications received. In this case, AI generative support could also be used.

Refining Test-Cases for Execution: UCTM may request the Test Generator to leverage the test cases as predefined test cases do not fully satisfy the cybersecurity needs or assess the desired cybersecurity quality level. The predefined and newly generated test cases will be selected and possibly combined to be executed by the Test Manager.

Generating Test Report: Analysis and Results component collects test results and sends back the user detailed statistical analysis. Moreover, it generates suggestions and recommendations to mitigate vulnerabilities revealed during test execution. Test results and suggestions are displayed to the user through UTM GUI or UI (User Interface), which could provide feedback for further improvement and optimization of the SECURER testing process. The following section will discuss the preliminary implementation of the SECURER components and their potential integration with state-of-the-art tools to perform cybersecurity testing and detect vulnerabilities through the user as a tester.

5 PRELIMINARY IMPLEMENTATION

The architecture and the behavioral model described in the previous section have been preliminary implemented into a prototype solution. In its realization, the primary aim waweb.skys to demonstrate the feasibility of the SECURER proposal, even at the expense of its effectiveness and overall performance. Indeed, open-source solutions and tools have been selected for all SECURER components shown in Figure 1, with minimal integration and some manual interactions. The *Test Generator, Test Executor, Test Controller*, and *Test DB* have been realized by employing available (open-source) tools and facilities. In particular, frameworks such as the OWASP Top Ten[4], MITRE ATT&CK[5], and CVE (Common Vulnerabilities and Exposures)[6] databases have been taken into consideration.

A pre-selection of needs and requirements was manually collected to create the questionnaire for cybersecurity testing using OLLAMA (Large Language Models)[7], while MySQL or Mongo DB[8], has been used for *Test DB* realization.

A preliminary version of UI/GUI of UCTM has been realized through React [9].

The core part of our preliminary implementation was the test generator. For its realization, the available (open-source) test generation tools and facilities have been explored, and GitHub resources and test suites are already available and integrated into the implementation. The possibility using language learning models (LLMs) integrated via OPEN API[10] was also considered.

For the test case execution, available testing libraries, extensions, and frameworks like Pytest[11], JUnit[12], and NUnit[13], have been considered. Finally, a *Test DB* has been included to collect details about the executed test cases and any identified vulnerabilities that have been classified according to their severity, and potential recommendations for mitigation should also be included in the report.

In the remaining part of this section, more technical details about the tool used for implementing the SECURER components and a discussion about its compliance and optimization are provided. In certain cases, some specific components have been realized through stubs or some manual interaction. Additionally, for feasibility reasons, the current SECURER implementation has been limited to two IoT domains: drone and cobot systems. In particular, we used:

Metasploit[14]: Metasploit is a powerful tool for simplifying hacking. It allows users to select exploits, choose payloads, configure options, and execute attacks against target systems. Metasploit has been used for realizing our Test Executor in Figure 1.

Aircrack-ng[15]: It is a suite of tools for assessing WiFi network security. It is utilized to test the security of wireless communications used by IoT.

[4] https://owasp.org/www-project-top-ten/

[5] https://attack.mitre.org/

[6] https://cve.mitre.org/

[7] https://github.com/ollama/ollama

[8] https://www.mongodb.com/

[9] https://react.dev/learn/describing-the-ui

[10] https://openapi.it/

[11] https://docs.pytest.org/en/8.2.x/

[12] https://junit.org/junit5/

[13] https://nunit.org/

[14] https://www.metasploit.org/

[15] https://www.aircrack-ng.org/

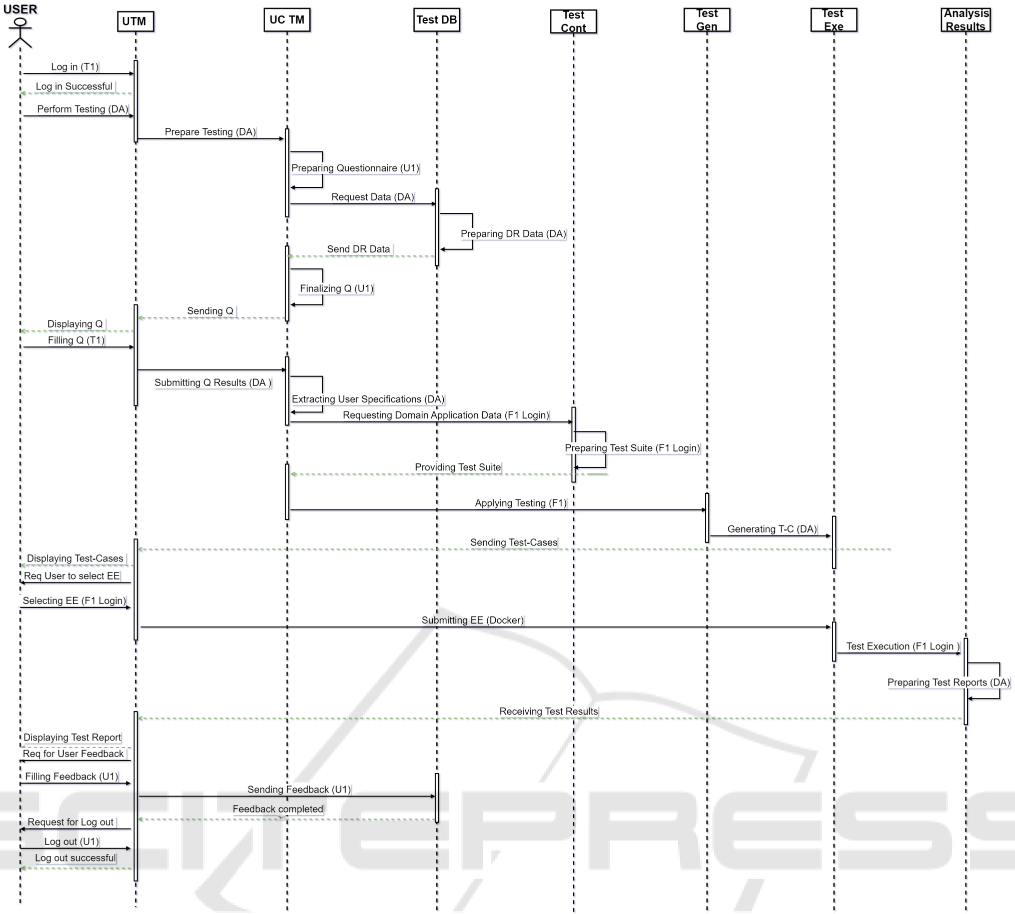

Figure 2: Sequence Diagram SECURER. Note: **DA**: Domain Application; **U1**: User-1;**Q**: Questionnaire; **DR**: Domain Requirements; **T1**: Testing-1/Tester-1; **F1**: Feature 1 (Feature to be tested against cyber-attacks and vulnerabilities);**T-C**: Test-cases (Generating test-cases for the feature to be tested by the user as per their domain requirements).

Aircrack-ng has been used to realize our Test Executor in Figure 1.

OWASP ZAP (Zed Attack Proxy) [16]**:** It is open-source web application security scanner. It can be used to test web interfaces that might be present in cobot systems.

OWASP ZA Phas has been used to realize our Test Executor in Figure 1.

Wireshark:[17] This network protocol analyzer captures real-time data on a network.

Wireshark has been used to recognize the information for our Test Controller extracted for the Test Generator in Figure1.

CONCERN:[18] It's an open-source, customizable Complex Event Processing Monitor Infrastructure for monitoring test case execution and vulnerability detection. It has been evaluated in various contexts and application domains (such as (Barsocchi et al., 2018; Calabrò et al., 2021; Barsocchi et al., 2021)). This component has been used to realize our Test Executor in Figure 1.

The SECURER prototype complies with international standards and regulations for security and industrial safety. It aims to enhance IoT security and develop reliable components for more productive and cyber-secure industrial operations.

6 SHOWCASING SECURER IN IoV

To showcase the SECURER prototype in a realistic environment, we focused on cyber-attacks targeting the Internet of Vehicles (IoV). In particular, we considered the rolling code technology used in some automotive security keys. This system generates a new code for each operation, ensuring that the same code

[16] https://www.zaproxy.org/

[17] https://www.wireshark.org/

[18] https://github.com/ISTI-LABSEDC/Concern

is never used twice. The purpose is to test it against vulnerabilities and cyber-attacks such as replay attacks and Man-in-the-middle. As with any other domains, automotive manufacturers and the broader industry should prioritize implementing robust cybersecurity testing strategies to identify and address vulnerabilities in IoV software because attacks may vary significantly based on the attackers' skill sets and the specific areas they target (Hsu et al., 2023).

To experiment, we populated the *Test DB* component of Figure 1 with the required information about the most prevalent threats and attacks of the IoV rolling code system. We also look for a preselection of test cases focused on specific attacks that could cover different roles and scenarios. The different data have been classified into various specific categories, each posing unique threats and challenges. The section explores prevalent threats to the IoV and IoT ecosystem. Cyber-attackers can assume various roles, exhibiting passive, aggressive, and malicious behaviors, locally and internationally. Access to the IoV network could disrupt the entire system. Cyber-attacks on IoV are classified into specific categories, each posing unique threats and challenges. In response to the unobtainable commercial IoV system, our team replicated a replacement for the security-key mechanism in our laboratory. It was carried out with readily accessible open-source components. The aim was to demonstrate how the SECURER operates and its potential to resolve cybersecurity issues and vulnerabilities in various application domains such as cobots and drones.

Typically, these signals, which are meant to be a single-use code to safeguard against unauthorized access, can also be intercepted using off-the-shelf devices like Flipper Zero[19] or more advanced devices like HackRF-One[20] shown in Figure 3. Once captured, the attacker can replay this signal to unlock the vehicle. This kind of vulnerability not only puts the physical security of the car at risk but also opens up pathways for further attacks on the connected infrastructure, potentially leading to broader disruptions within the IoT ecosystem.

Wherever possible, the *Analysis and Result* component of Figure 1 has also provided some available countermeasures to mitigate potential cyber-attacks. As described in the previous section, this information can be provided to the *User-centric Test Manager* of Figure 1 as feedback and for ensuring proactive cybersecurity activity(Gupta et al., 2023).

In this experiment, all the activities concerning generating the UI activity have been performed through stub. The UI stub was in charge of managing the questionnaire and subsequent data filling, as described in section 4. In this case, the *User-centric Test Manager* of Figure 1 received from the UI stub a prefilled form in which the replay attack has been selected and considered critical. A replay attack consists of intercepting the signals of the rolling codes during the locked or unlocked operations and using the collected code to unlock the vehicle without needing the original key fob. This attack compromises the vehicle's security and poses a significant threat to the automotive ecosystem to which the car might belong.

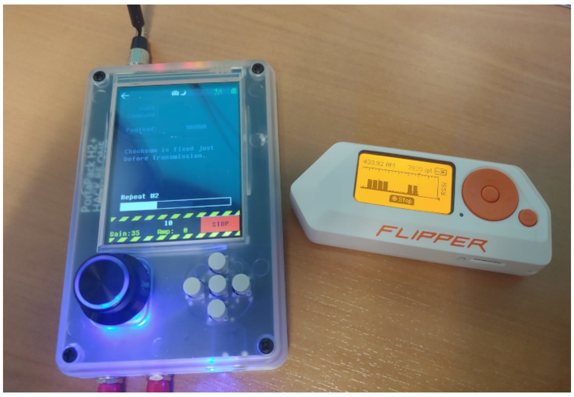

Figure 3: Two off-the-shelf devices for MiM attacks.

Thanks to the questionnaire data *User-centric Test Manager*, *Test DB* and *Test Controller* Figure 1 collaborated in identifying the most suitable test strategies and test cases mitigate the risks associated with replay attacks. In this instance, the *Test Generator* component's generative AI provided 20 distinct test cases in response to a request to simulate attack scenarios on the security of the tested rolling code systems. Through stub UI, the *Test Executor* received then the rolling code systems instance [21] to be tested and the link to the simulation environment where the test could be automatically executed.

The test case execution simulated the potential attackers, enabling testers to identify weaknesses in the received code implementation. During the execution, the *Analysis and Result* component performed Static Analysis (SAST) of the collected test results. It evaluated the security level of the tested Internet of Vehicles (IoV) applications.

In this experiment, the test case execution reveals minor failures in the analyzed rolling code. However, the *Analysis and Result*, still using generative AI, provided adequate guidelines suggesting regular security audits and stress tests on the rolling code systems to ensure they remain impervious to replay

[19] https://flipperzero.one/

[20] https://greatscottgadgets.com/hackrf/one/

[21] https://github.com/robert-mcdermott/rolling-code-auth

attacks. Moreover, the SECURER recommends the integration of encrypted signals and implementing a dynamic code verification system that can invalidate a code immediately once it is used, further enhancing the system's security. Even if very simple, the experiment proves the proposal's feasibility and lets us highlight the peculiarity, potentiality, and criticalities of the SECURER implementation.

The findings indicate that vehicles do not automatically realign the rolling codes upon receiving sequential lock/unlock prompts from various sources. This discovery, alongside the identified test scenarios and the examination of real-time replay attacks, is anticipated to refine our approach to cybersecurity, focusing on user safety and trust. This strategy of consistent user feedback and achieving optimization is outlined in our SECURER's designed architecture and operational model. However, the integration of components within our SECURER is crucial for the effectiveness of our cybersecurity testing framework.

7 CONCLUSIONS AND FUTURE WORK

The SECURER framework provides a plan to protect against replay attacks on rolling code systems, crucial for safeguarding vehicles and the broader IoT ecosystem. The paper outlines the framework's structure and core functionality and a preliminary prototype to demonstrate its feasibility. Future improvements include integrating with updated cybersecurity tools to cater to a wider range of users and application domains, enhancing the process of collecting user test requirements, and exploring the possibility of integrating GitHub Lab for user feedback and test reports.

ACKNOWLEDGEMENTS

This work was partially supported by the project RESTART (PE00000001) and the project SERICS (PE00000014) under the NRRP MUR program funded by the EU - NextGenerationEU.

REFERENCES

Ali, M., Shahroz, M., Mushtaq, M. F., Alfarhood, S., Safran, M. S., and Ashraf, I. (2024). Hybrid machine learning model for efficient botnet attack detection in iot environment. *IEEE Access*, 12:40682–40699.

Barhamgi, M., Perera, C., Ghedira, C., and Benslimane, D. (2018). User-centric privacy engineering for the internet of things. *IEEE Cloud Computing*, 5(5):47–57.

Barsocchi, P., Calabrò, A., Crivello, A., Daoudagh, S., Furfari, F., Girolami, M., and Marchetti, E. (2021). COVID-19 & privacy: Enhancing of indoor localization architectures towards effective social distancing. *Array*, 9:100051.

Barsocchi, P., Calabrò, A., Ferro, E., Gennaro, C., Marchetti, E., and Vairo, C. (2018). Boosting a low-cost smart home environment with usage and access control rules. *Sensors*, 18(6):1886.

Calabrò, A., Cioroaica, E., Daoudagh, S., and Marchetti, E. (2021). BIECO runtime auditing framework. In *14th (CISIS and ICEUTE), 2021*, volume 1400 of *Advances in Intel. Systems and Computing*, pages 181–191. Springer.

Gupta, S., Maple, C., and Passerone, R. (2023). An investigation of cyber-attacks and security mechanisms for connected and autonomous vehicles. *IEEE Access*.

Heiding, F., Süren, E., Olegård, J., and Lagerström, R. (2023). Penetration testing of connected households. *Computers & Security*, 126:103067.

Hsu, C.-H., Alavi, A. H., and Dong, M. (2023). Introduction to the special section on cyber security in iov.

Huang, H.-C., Liu, I.-H., Lee, M.-H., and Li, J.-S. (2023). Anomaly detection on network traffic for the healthcare iot. *Engineering Proceedings*, 55(1).

Matheu-García, S. N., Hernández-Ramos, J. L., Skarmeta, A. F., and Baldini, G. (2019). Risk-based automated assessment and testing for the cybersecurity certification and labelling of iot devices. *Computer Standards & Interfaces*, 62:64–83.

Pietrantuono, R., Ficco, M., and Palmieri, F. (2023). Testing the resilience of mec-based iot applications against resource exhaustion attacks. *IEEE Transactions on Dependable and Secure Computing*.

Puthal, D., Wilson, S., Nanda, A., Liu, M., Swain, S., Sahoo, B. P., Yelamarthi, K., Pillai, P., El-Sayed, H., and Prasad, M. (2022). Decision tree based user-centric security solution for critical iot infrastructure. *Computers and Electrical Engineering*, 99:107754.

Puthal, D., Yang, L. T., Dustdar, S., Wen, Z., Jun, S., Moorsel, A. v., and Ranjan, R. (2020). A user-centric security solution for internet of things and edge convergence. *ACM Transactions on Cyber-Physical Systems*, 4(3):1–19.

Rivadeneira, J. E., Silva, J. S., Colomo-Palacios, R., Rodrigues, A., and Boavida, F. (2023). User-centric privacy preserving models for a new era of the internet of things. *Journal of Network and Computer Applications*, page 103695.

Sáez-de Cámara, X., Flores, J. L., Arellano, C., Urbieta, A., and Zurutuza, U. (2023). Gotham testbed: a reproducible iot testbed for security experiments and dataset generation. *IEEE Transactions on Dependable and Secure Computing*.