J. Parada, C. Alcaraz, M. Reyes and J. Lopez, "Adaptive Adversary Emulation Framework for Cyberwarfare Operations", *IEEE Communications Magazine*, 2026.
http://doi.org/10.1109/MCOM.001.2500721
NICS Lab. Publications: https://www.nics.uma.es/publications

Adaptive Adversary Emulation Framework for Cyberwarfare Operations

J. Parada[1,2], C. Alcaraz[1], M. Reyes[2], and J. Lopez[1]

[1]Computer Science Department, University of Malaga, Spain
[2]Eurecat, Centre Tecnològic de Catalunya, Barcelona, Spain

Abstract

The geopolitical landscape is becoming increasingly complex and tense, with a largely silent cyber conflict unfolding on a daily basis. Consequently, traditional defenses based on detection and post-event analysis are therefore becoming insufficient. This situation has compelled both public and private entities to adopt a more proactive approach to addressing digital threats in order to anticipate them. This is particularly important in the Critical Infrastructure (CI) sector, given its strategic geopolitical role and the severe consequences that attacks on such infrastructures may entail. In this context, this paper proposes the Operational Platform for Offensive Replication (OPFOR), a modular framework for executing Adversary Emulation (AE) based on attacker behaviors attributable to real malicious actors with specific objectives. Likewise, we introduce the concept of Adaptive Adversary Emulation (AAE), defined as the execution of an actor's behavior adapted to a different context while preserving the similarity of the actions and the Tactics, Techniques, and Procedures (TTP) employed by adversaries.

1 Introduction

Cybersecurity management in the field of Critical Infrastructures (CI) has become a vital concern for modern societies, making it a central focus in research and academic domains [1]. The reason for this evolution is due to two main factors that have aligned to promote the current situation. The first factor is the technological growth of industrial ecosystems and the Industrial Internet of Things (IIoT), with numerous interconnected devices. This has led to the development of better automated and interconnected systems, but also to the development of formerly isolated industrial infrastructures, increasing their attack surface and becoming more difficult to protect due to their complexity. This interconnection in IIoT has grown, in part driven by the rapid technological advances of recent years, including the rise of technologies and approaches. These advances have led to a significant improvement in defensive techniques and tools, but also to an enhancement of malicious tactics and the sophistication of cyber-attacks, representing a great challenge, especially for the field of IIoT and critical scenarios. Secondly, industrial infrastructures, and more specifically the so-called CIs, present a lucrative target for cybercriminal organizations. Due to their critical role in modern society, these systems have become strategic targets for adversaries seeking economic gain. Additionally, the current global climate of warlike tensions has contributed to an increase of cyber operations and cyber-espionage performed by different states against this type of infrastructure, with specialized and well-resourced attackers. These malicious activities are quietly taking place in parallel, even between countries that are not directly involved in the conflict, but have alliances or opposing interests. A notable example of this type of cyber-attack is the operation conducted by *Sandworm*. The group was credited with the deployment of a malware in a regional energy company, successfully disconnecting up to nine substations with the objective of cutting power to nearly two million people [2]. In certain instances, state-attributed cyber-attacks targeting governmental services and critical national security infrastructure have escalated to the severance of diplomatic relations between the nations involved [3]. The emergence of proactive approaches like Cyber Threat Hunting (CTH) stands out among the different responses to the need to protect CIs. Unlike traditional defensive systems that prioritize detection and reactive measures, these approaches emphasize preparedness and the proactive discovery of emerging threats, aiming to stay one step ahead of enemies. Therefore, CTH is characterized by its proactive approach, using an inductive process of continuous hypothesis generation and validation based on observation and the knowledge collection [4].

To cover this issue, this article proposes the use of Adversary Emulation (AE) within the CTH paradigm,

a technique used to emulate the behavior of malicious actors within a controlled environment. Unlike other approaches such as penetration testing, AE focuses on Tactics, Techniques, and Procedures (TTPs) and attacker behavior rather than on the intrusion methods themselves. AE provides analyzable information that can be used to test defensive systems, measure their impact on infrastructures, and generate high-quality synthetic data for the development of new defensive and attribution systems. Given the strategic relevance of the field, this article introduces Operational Platform for Offensive Replication (OPFOR), a framework for AE specifically designed to model operations targeting real CI. Beyond reproducing realistic adversarial procedures, OPFOR incorporates adaptive behavior, allowing attack strategies to dynamically adjust to the structural and operational characteristics of the emulated infrastructure. To formalize this contribution, we introduce the concept of Adaptive Adversary Emulation (AAE), in which Cyber Threat Intelligence (CTI) is integrated with an internal understanding of network topology weaknesses. This integration enables the generation of contextualized and infrastructure-aware attack paths that reflect both adversarial intent and environmental constraints.

2 Related Work

The increasing sophistication of adversaries, including state-sponsored actors and organized cybercriminal groups, has intensified the difficulty of predicting attack behavior and performing trustworthy attribution. In this context, AE provides a structured methodology based on the systematic reproduction of TTPs, enabling a behavior-centric and CTI-informed execution of adversarial activity. Such emulation typically focuses on post-infection stages of the attack lifecycle, going beyond traditional vulnerability assessments [5, 6]. Numerous efforts have explored the application of AE across different sectors and methodological approaches. These contributions are summarized in Table 1, which synthesizes the most representative studies and compares the main aspects considered in this work to categorize and formally define AAE, the main contribution of this paper.

Compared to other studies, OPFOR is designed for CIs and adopts emulation-based environments as the foundational infrastructure for experimental validation. System adaptability is the core concept of the proposed approach, referring to the adversary's capability to dynamically design and refine attack strategies based on target network weaknesses and contextual conditions. This focus is partially explored by [7, 6], where agents integrate learning mechanisms and Machine Learning (ML) techniques to enable adaptive and context-aware offensive behaviors. Another key aspect is the ingestion and operationalization of dynamic CTI data. While numerous studies rely on structured TTPs repositories, such as those integrated into platforms like MITRE Caldera, only our prior work explicitly addresses the dynamic consumption and integration of vulnerability information and CTI. This capability is also incorporated into OPFOR, reinforcing its intelligence-driven AE model. Within the adaptability and dynamism dimensions explored in prior AE studies, the novelty of this work lies in the execution of an operation inspired by *Stuxnet*, adapted to a different network topology and designed to emulate the tactics of a specific threat actor. Among the reviewed studies, the work that most closely aligns with the objectives of OPFOR is [5]. Nevertheless, its approach relies heavily on manual strategy design and lacks autonomous adaptation and dynamic reconfiguration capabilities, which limits its applicability across multiple scenarios. When contrasted with our previous work, [6], which partially inspired the present framework, several distinctions emerge. While the research introduced foundational elements that informed the design of this approach, it does not incorporate the emulation of a real operation nor the behavioral modeling of a specific threat actor. Instead, it is limited to the execution of a general and relatively simple strategy, without an explicit tactical objective. It imposes no operational constraints governing which nodes may be targeted or the order of attack stages. In contrast, the present framework structures attacks as a prioritized chain, where precondition nodes must be compromised before the final target can be reached.

3 Architecture and Infrastructure

OPFOR comprises three main modules, *i.e.* the (i) *Infrastructure Module*, (ii) *CTH Module*, (iii) and *AE Module*. The former consists of all interrelated systems, networks, resources and data that need to be protected and tested within the CI. The *CTH Module* enables planning and creating adversaries to be executed, and validating their performance and impact. Finally, the *AE Module* consists of simulation of the environment and emulation of adversaries therein. It is important to note that this is not the AAE module, since adaptation emerges from the combination of the processes performed in this module together with the infrastructure and CTI analysis conducted in the other modules. Under this architecture, the system follows a structured workflow. First, CI data is collected from the *Infrastructure Module* to replicate the environment observed. Thus, an automated adversary is modeled based on a real actor or intrusion. This *Adversarial Model* is then executed within the replication infrastructure, which represents the target environment with different levels of abstraction. Finally, the agent's behavior and performance are validated.

The *Infrastructure Module* is the only layer that may exist outside the framework system domain, mainly because of infrastructure delocalization. This layer repre-

Table 1: Main adversary emulation researches and approaches

Ref	Year	Sector	Infrastructure	Adaptive	CTI	Automatic	Operation	TTP	Actor	Chain
[7]	2020	Cyber-physical	Simulator	✓	✗	✓	✗	✗	✗	✗
[5]	2021	General	Laboratory	✗	✗	✗	✗	✓	✓	✗
[8]	2023	IoT	DT	✗	✗	✓	✗	✗	✗	✗
[9]	2023	General	Testbed	✗	✗	✓	✗	✓	✗	✗
[10]	2024	General	Hypervisor	✗	✓	✗	✗	✓	✗	✗
[6]	2025	Industrial	DT	✓	✓	✓	✗	✓	✗	✗
OPFOR	2026	CI	Emulator	✓	✓	✓	✓	✓	✓	✓

sents the set of systems, both digital and physical, that must be protected. Although this paper focuses on CIs, the layer also may include computer systems belonging relevant sectors, such as citizen data management. The main component is called the *Data Gatherer*, responsible for collecting information about the infrastructure, including software, hardware, and all elements required to recreate the environment in a virtualized layer. The generated information is mainly collected through two types of processes: (i) manual processes, where system definition information is gathered using known standards, and (ii) automated processes, where data from sensors, Programmable Logic Controllers (PLCs), and other devices provide status information or are extracted through additional monitoring devices. Finally, this information is sent to the *CTH Module*, specifically to the *Cyber Threat Hunting Hypothesis Generation Module (CTH-HGM)*, where it is analyzed to generate hypotheses about previously unknown threats.

4 Cyber Threat Hunting Module

The *CTH Module* is responsible for generating and validating hypotheses about new threats. This process follows an inductive approach in which telemetry data, vulnerabilities, and TTPs are analyzed to identify weaknesses related to emerging or previously unknown threats. The module is further divided into two submodules: *CTH-HGM*, responsible for hypothesis generation based on information collection and analysis, and *Cyber Threat Hunting Hypothesis Validation Module (CTH-HVM)*, responsible for hypothesis validation.

Within *CTH-HGM*, a series of consecutive phases are performed by CTH experts. The first phase consists of the *Collection* of information, both internal from the *Infrastructure Module* and external sources (from CTI). In the next phase, the *Analysis* of the information, CTH experts analyze CTI records according to the context of their specific industry. Beyond common CTI sources, CIs belonging to strategic sectors must also consider the geopolitical environment in which they operate. This includes assessing the potential societal impact of infrastructure failure and the relationships with interdependent businesses and governmental institutions to better understand the incentives of different adversaries, such as Advanced Persistent Threat (APT) groups. Based on this analysis, hunters can estimate which assets are most susceptible to cyber-attacks by performing risk assessments based on asset criticality, emerging threats, and geopolitical interests.

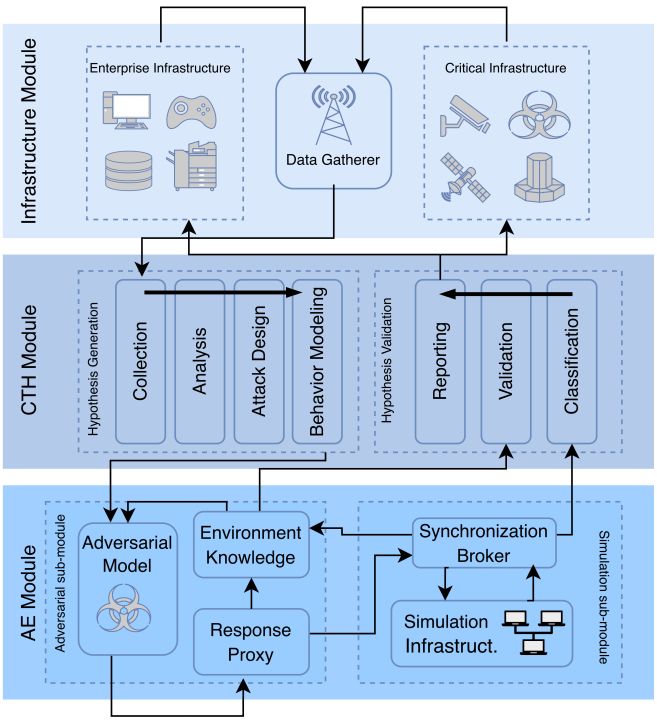

Figure 1: General Architecture of the OPFOR Framework

The two subsequent phases involve the design of experiments to be executed in the *AE Module*. These experiments include *Attack Design*, which defines the attacker's objective grounded in a real actor or strategy. The attack aims to exploit a vulnerability in the environment. The adversary must be equipped with sufficient tools and capabilities, implemented either through predefined algorithms or through Artificial Intelligence (AI) techniques that enable learning about the system's topology and attacking it. The final phase is *Behavior Modeling*. Un-

like the previous phase, this process focuses on reproducing specific TTPs from the actor or cyberattack being emulated, prioritizing behavioral similarity over performance. In both phases, the use of ML-based techniques has been highlighted, including studies using Reinforcement Learning (RL) [7, 6]. Although research on the compatibility of AE with more recent AI approaches, such as Large Language Models (LLMs), remains limited, pentesting-based works [11] suggest that this combination may be promising.

CTH-HVM consists of three consecutive phases. The first, *Classification*, extracts information from the raw data generated by the *AE Module*. This phase may include classifying TTPs, detecting threats or anomalies in the system, or normalizing data and generating statistics. The next phase is the *Validation* of the hypothesis, which involves analyzing the classified information to verify that the generated hypothesis satisfies the initially defined requirements. In the specific case of AAE, the fundamental aspects to be validated are: (i) the agent's performance is minimally effective, i.e., it is able to breach the system at a predefined or positive ratio; and (ii) the agent performs malicious actions following behavior similar to that defined during CTH-HGM. *Reporting* is the final step, which involves generating remediation strategies and mitigation scripts derived from the observed attack patterns. For instance, the analysis may reveal nodes that facilitate lateral movement or cross-segment connectivity within the infrastructure, as discussed in Section Validation and Behavior Analysis, which reflects the agent's propagation tendencies. This even enables the identification of potential vulnerabilities and strategically critical elements that require defensive prioritization.

5 Adversary Emulation Module

The *AE Module* manages information about the environment and adversary knowledge, making decisions that are translated into actions within the model. From a cyber-warfare perspective, this module constitutes the core of the framework and gives rise to the term OPFOR, commonly understood in the military as *Opposing Force*, *i.e.*, units tasked with imitating an enemy during tactical training exercises. In this context, the objective is to reproduce adversarial behavior within a cyber environment. This module is divided into two parts: the *Adversarial Sub-module*, responsible for planning malicious actions, and the *Simulation Sub-module*, responsible for executing realistic and accurate simulations of the infrastructure. The core of the *Adversarial Sub-module* is the *Adversarial Model*, designed and deployed from CTH-HGM. This model stores the algorithms and parameters that define malicious behavior, effectively representing the adversary's decision engine. These algorithms interact with the environment, receiving the current state of the environment and producing a decision accordingly. The *Adversarial Model* receives information from the *Environment Knowledge* component, which represents a mathematical abstraction of the system's current state. This component acts as a proxy, limiting the knowledge accessible to the adversary in order to prevent unrealistic advantages. Conversely, the *Response Proxy* captures the actions generated by the adversary, applies them to the simulation layer, and blocks any illegal or inconsistent actions that the adversary may attempt to execute.

The *Simulation Sub-module* comprises components responsible for executing a virtual system that mirrors the physical environment represented in the *Infrastructure Module*. It contains two main components. The first component is the *Synchronization Broker*, which receives information from the *Adversarial Model*, and the simulation itself, and propagates updates by notifying all components of relevant events and state changes. The second component is the simulator and its supporting infrastructure, typically implemented through resource virtualization. To achieve high-fidelity simulations, different technologies may be employed, such as virtual machines or containers, as well as synchronization approaches including Digital Twin (DT) [6], LLM-based infrastructure emulators [12], or formalized probabilistic models [7].

6 Experimental Design

In 2010, the discovery of a piece of malware, known as *Stuxnet*, represented a paradigm shift in security for CI. This malware was responsible for infecting industrial systems focused on uranium enrichment. The attack vector consisted of three phases. The first one involved infecting *Microsoft Windows* systems, on which it spread like a worm. Then it looked for specific industrial PLC software responsible for operating part of the industrial environment, including centrifuges. Finally, it compromised the system with espionage skills or by causing the centrifuges to fast-spin. Although the true actor of the attack remains unknown, the high degree of sophistication and complexity of the malware points to an actor with considerable technical capabilities and substantial resources [13]. The design of our experiment is based on an industrial environment with similar characteristics, attempting to replicate similar geopolitical actions. For this purpose, we simulate an industrial network based on nuclear plant with vulnerable components. The malware's main objective is not complete destruction but damaging nuclear power plant turbines to sabotage a country's energy system. More specifically, the adversary enters through a control panel using a Universal Serial Bus (USB) provided by an insider. They then have to breach the hubs that transport information to the control and security systems with the aim of disrupting the information and preventing real information from passing

through them. Finally, the malware must attack the systems controlling the turbines with the aim of damaging them, now without being detected.

To ensure behavior similar to that of a real attacker, we model an RL-based agent whose knowledge of the environment is restricted to nodes within its reach. Through the reward function, the agent learns where to perform lateral movement and which parameters apply for each attack vector. The attack process follows two stages: (i) reconnaissance/scanning and (ii) exploitation, supported by a knowledge base of vulnerabilities, exploits, and TTPs. This reward function is determined by the following conditions, which indicate whether the attacker has achieved victory or not: (i) the adversary wins if it takes control of the turbine servers (hereinafter referred to as targets); (ii) the adversary must take control of the targets within a predetermined time limit; and (iii) the adversary loses if it takes control of the targets and there is at least one uncompromised path to an uninfected control panel. These conditions in the reinforcement policy ensure that the agent performs the attack while respecting its phases and delivering effective performance. The first condition specifies what the exact objective is, causing the agent to focus, even if it is in later phases on the turbine server. The second condition ensures minimum performance, forcing the agent to take advantage of topological characteristics, learn from the weaknesses of its design and components, and reach its goal as quickly as possible. The last condition, ensures that the behavior is identical to the design, preventing information leaks from the turbines and providing absolute control over the data received by the panels. Studies like [2] and CTI reports like [14] show that energy infrastructures are key targets for adversary groups in operational scenarios and that attacks against them have increased in recent years. For these reasons, a network topology of a nuclear power plant has been designed based on the proposal by [15], but composed of industrial components with known vulnerabilities so that the adversary is able to bypass this system with a certain probability depending on the actions taken. The essential components for the experiment are ilustrated in Figure 2 Network Infrastructure graph, where control panels, entry point, hubs, and targets are highlighted. In addition to the topology, as can be seen in the image, *Honeypot* nodes have been added that do not provide any advantage to the attacker, with the aim of hindering the algorithm. This experiment is designed as a proof-of-concept validation of the proposed AAE framework, specifically intended to demonstrate its infrastructure adaptive capabilities. In particular, the following evaluation illustrates how the agent, without prior knowledge of the environment, is able to progressively infer the structural and strategic characteristics of the CI through a reward-driven learning process.

7 Validation and Behavior Analysis

Following the OPFOR framework structure, the second part of the experiment consists in analyzing the data generated to validate that the hypothesis is correct. Specifically, the type of behavior performed matched expectations, and it achieved a considerable success rate. To do this, we have extracted from the RL algorithm the different weights assigned to each node, given each state, calculating an arithmetic mean to evaluate how good the agent considers the conquest of that node. This allows to observe in a graph the agent's priorities and the routes it follows in its propagation, as shown in Figure 2 Graph A. In this graph, we observe a tendency to conquer the central ring and some of its nodes, with the intention of isolating control panels. Graph B, in this same figure, shows the same trend, but weighting the movements made at the end of the experiment, showing how all nodes lose importance except those surrounding the target. The agent's performance is reflected in the win–loss statistics in Figure 3, where the ratio improves over time, measured by the number of executed attack vectors instead of time intervals. This makes sense, as the shorter the time, the fewer attempts the agent has to infect each node, and a series of consecutive failures may lead to defeat even if the planned moves are accurate. As time increases, failed movements become less critical, and the agent achieves ratios above 90%. Longer time budgets increase the entropy of the agent's policy, becoming more exploratory as it is no longer constrained to focus strictly on the minimum set of precondition nodes. Thanks to these two charts, ratios can be established regarding the importance of securing each node, using the heat map in Figure 2 to determine the attacker's priorities. On the other hand, it is also possible to obtain a relationship regarding the importance of the detection capability strategy, using the proportion in Figure 3 to determine how important early detection can be in that scenario.

8 Conclusion

This article presents a framework for the development of what is, to the best of our knowledge, the first introduction to AAE, building upon a previous experimental study and a review of the literature on AE. We highlight the relevance of AE in the current global context and distinguish it from related techniques such as pentesting. The framework emphasizes several key aspects: the geopolitical landscape, the use of CTI data from real attackers, and the main role of malicious behavior in attack emulation. To further contextualize this approach within targeted and critical domains, the needs of critical infrastructures have been analyzed. Due to their economic value and strategic importance, these environments are more likely to be targeted by orga-

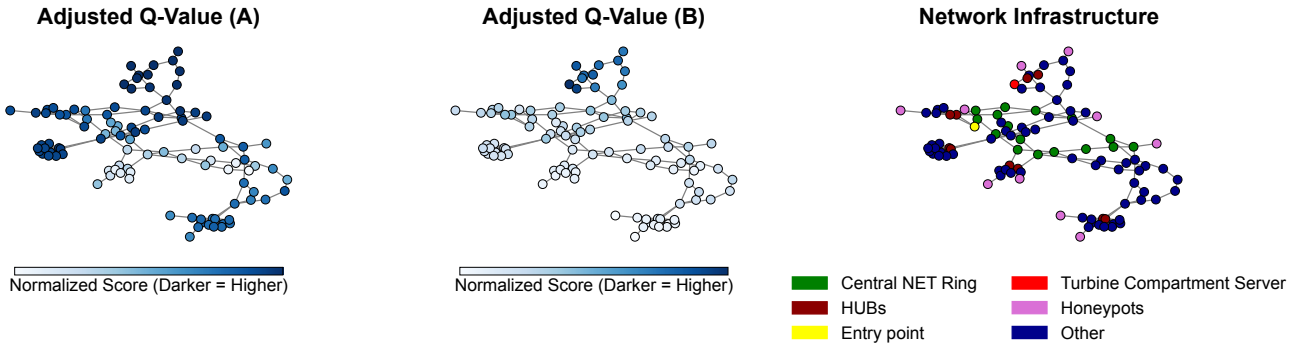

Figure 2: RL-algorithm Weights and Agent Propagation

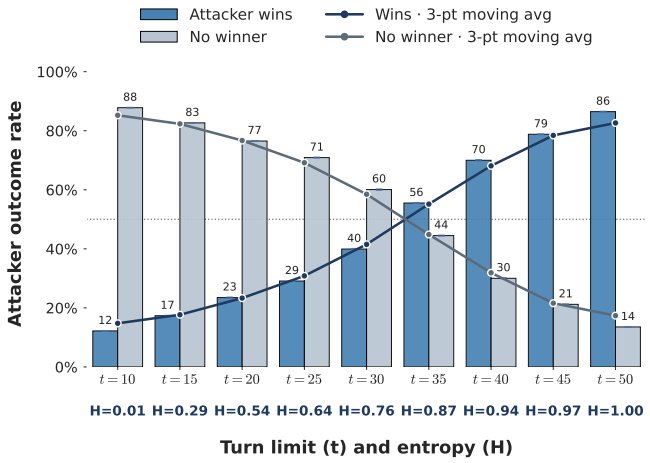

Figure 3: Victory and Defeat Ratio

nized and state-sponsored actors. Finally, to provide operational meaning to AE, it is integrated as a process within CTH, with the objective of identifying cyber threats and attacks that have not yet occurred or remain unknown. This process defines the overall structure of the framework, as described in Section Architecture and Infrastructure, resulting in an innovative methodology focused on hunting undiscovered threats by applying known TTPs to new contexts.

9 Biography Section

Javier Parada is a PhD student in Eurecat and University of Malaga, focused on Cyber Threat Hunting.

Cristina Alcaraz is an Associate Professor in the Department of Computer Science at the University of Malaga, focused on security of IIoT and digital twins.

Mario Reyes is Scientific Director of Eurecat's IT&OT Security Technology Unit, focused on application of AI techniques and technologies in security management.

Javier Lopez is Full Professor in the Department of Computer Science at the University of Malaga, focused on network security and security protocols.

10 Acknowledgments

The first author is a fellow of Eurecat's "Vicente López" PhD grant program. Also, the work was partially supported by the project AIAS funded by the European Commission (EC) under Grant 101131292 (HORIZON-MSCA-2022-SE-01), and SYNAPSE funded by the EC under Grant 101120853 (HORIZON-CL3-2022-CS-01)

References

[1] Cristina Alcaraz and Javier Lopez. Digital twin: A comprehensive survey of security threats. *IEEE Communications Surveys & Tutorials*, 24(3):1475–1503, 2022.

[2] Andy Greenberg. Russia's sandworm hackers attempted a third blackout in ukraine, 2022. Accessed July 18, 2025.

[3] Al Jazeera Staff. Albania cuts diplomatic ties with iran over cyberattack, 2022. Accessed: 2025-07-22.

[4] Xiaokui Shu, Frederico Araujo, Douglas L Schales, Marc Ph Stoecklin, Jiyong Jang, Heqing Huang, and Josyula R Rao. Threat intelligence computing. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 1883–1898, 2018.

[5] Abdul Basit Ajmal, Munam Ali Shah, Carsten Maple, Muhammad Nabeel Asghar, and Saif Ul Islam. Offensive security: Towards proactive threat hunting via adversary emulation. *IEEE Access*, 9:126023–126033, 2021.

[6] Javier Parada, Cristina Alcaraz, Javier Lopez, Juan Caubet, and Rodrigo Román. Digital twin for

adaptive adversary emulation in iiot control networks. In Vincent Nicomette, Abdelmalek Benzekri, Nora Boulahia-Cuppens, and Jaideep Vaidya, editors, *Computer Security – ESORICS 2025*, pages 423–442, Cham, 2026. Springer Nature Switzerland.

[7] Arnab Bhattacharya, Thiagarajan Ramachandran, Sandeep Banik, Chase P Dowling, and Shaunak D Bopardikar. Automated adversary emulation for cyber-physical systems via reinforcement learning. In *2020 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 1–6. IEEE, 2020.

[8] Ewout Willem van der Wal and Mohammed El-Hajj. Securing networks of iot devices with digital twins and automated adversary emulation. In *2022 26th International Computer Science and Engineering Conference (ICSEC)*, pages 241–246. IEEE, 2022.

[9] Saeid Sheikhi and Panos Kostakos. Cyber threat hunting using unsupervised federated learning and adversary emulation. In *2023 IEEE International Conference on Cyber Security and Resilience (CSR)*, pages 315–320. IEEE, 2023.

[10] Vittorio Orbinato, Marco Carlo Feliciano, Domenico Cotroneo, and Roberto Natella. Laccolith: Hypervisor-based adversary emulation with anti-detection. *IEEE Transactions on Dependable and Secure Computing*, 2024.

[11] Isamu Isozaki, Manil Shrestha, Rick Console, and Edward Kim. Towards automated penetration testing: Introducing llm benchmark, analysis, and improvements. In *Adjunct Proced. of the 33rd ACM Conf. on User Modeling, Adaptation and Personalization*, pages 404–419, 2025.

[12] Anıl Sezgin and Aytuğ Boyacı. Decoypot: A large language model-driven web api honeypot for realistic attacker engagement. *Computers & Security*, 154:104458, 2025.

[13] David Kushner. The real story of stuxnet. *IEEE Spectrum*, 50(3):48–53, 2013.

[14] Trustwave SpiderLabs. Ransomware attacks against the energy and utilities sector up 80 percent, January 2025. Accessed: 2025-10-24.

[15] Mikhail Evgenievich Byvaikov, Elena Filippovna Zharko, Nadyr Enverovich Mengazetdinov, Aleksei Grigor'evich Poletykin, Iveri Varlamovich Prangishvili, and Vitalii Georgievich Promyslov. Experience from design and application of the top-level system of the process control system of nuclear power-plant. *Automation and Remote Control*, 67(5):735–747, 2006.