AI-Powered Penetration Testing: Navigating the Phases of Cybersecurity

Kiera Conway
Dakota State University
Seattle, USA
Kiera.Conway@trojans.dsu.edu

***Abstract*—The modern integration of Artificial Intelligence (AI) into Penetration Testing (PT) is rapidly becoming a transformative force. This report conducts a comprehensive review of five novel research papers, highlighting the escalating demand for AI-driven PT tools, the necessity for further research, and the motivation for innovative approaches to outpace cyber threats. Building from these insights, a short seminar course titled "Harnessing Artificial Intelligence for Penetration Testing" is designed. It aims to provide students with a comprehensive understanding of PT and the vital role of AI in enhancing efficiency and efficacy. This course goes beyond theoretical knowledge; it imparts practical skills and empowers students to navigate the complex realm of cybersecurity, combat adversaries, and actively contribute to a more secure digital environment.**

Keywords— Artificial Intelligence, Penetration Testing, Machine Learning, Reinforcement Learning, Deep Learning, Cybersecurity, Ethical Hacking, Vulnerability Assessment

INTRODUCTION

In the complex landscape of cybersecurity, the art of penetration testing (PT) has emerged as a critical and dynamic discipline. PT, often referred to as ethical hacking, is the "offensive approach" of probing and assessing computer systems, networks, and applications "to actively identify vulnerabilities and then exploit them in the same way as a genuine attacker [1]." Therefore, as modern digital tools become increasingly intelligent and interconnected, organizations face mounting challenges to safeguard their assets and data. The need for skilled professionals who can "provide organizations with actionable information about their security posture, enabling them to identify and prioritize areas of risk and improve their overall security [2]," has never been more pressing.

PT is a multifaceted endeavor that approaches cybersecurity in a series of meticulous steps. It begins with Gathering Information Phase, where data about the target system and potential vulnerabilities is meticulously compiled. This is followed by the Scanning Phase, where potential vulnerabilities are verified as exploitable. Once vulnerabilities are confirmed, the Exploitation Phase leverages these weaknesses to gain access to the target system. The following three phases, which include establishing persistence, minimizing traces of the intrusion, and documenting any results, fall into the post-exploitation category [3]. To streamline discussion in this report and course, these three phases will collectively be referred to as the 'Post-Exploitation Phase,' thus enabling a more central focus on the initial three phases.

Given the complex nature and substantial workload inherent to traditional penetration testing (PT), it necessitates significant expertise from penetration testers (pentesters) [4]. However, due to a current critical shortage of skilled experts in the field, there is a fundamental need to embrace the automation group of AI, Machine Learning (ML), and Reinforcement Learning (RL). First, the growing complexity of networks demands a more sophisticated and efficient approach to identifying vulnerabilities. Traditional manual PT methods, while valuable, are often time-consuming, resource-intensive, and may struggle to keep pace with the dynamic nature of cyber threats. Additionally, the incorporation of ML into automated PT systems has the potential to reduce "recurrent human errors" resulting from factors such as "tiredness, omission, and pressure [1]." Automated PT can significantly reduce the time and resources required for comprehensive testing, making it a practical choice for organizations aiming to bolster their cybersecurity defenses. As such, the evolving field of automated PT utilizes "advanced algorithms, machine learning, and AI to scan systems for vulnerabilities," and significantly improves multiple facets of PT [2].

This recent surge in AI capabilities, and its potential to revolutionize multiple phases of PT, hig emphasize the importance of spreading knowledge on 'Harnessing Artificial Intelligence for Penetration Testing.' From introducing the basics of AI to delving into advanced topics such as AI-driven scanning, vulnerability assessment, and post-exploitation techniques, this course offers a comprehensive exploration of not only the current AI landscape, but also its future possibilities. The primary goal is to not just keep pace with evolving threats but to leap ahead, embracing the most cutting-edge technologies in modern AI, and empowering innovative cybersecurity professionals to navigate the complex digital frontier with confidence.

LITERATURE REVIEW

A. Review 1: Leveraging Large Language Models for PT

Introduction

During the upcoming European Software Engineering Conference proceedings, researchers Andreas Happe and Jürgen Cito will present a compelling exploration of the integration of Large Language Models (LLMs) into the realm of PT. LLMs, such as ChatGPT, GPT3.5, and AutoGPT, have gained significant popularity recently due to their remarkable ability to predict missing data and generate human-like text. As a result of these pattern-recognition abilities, which are learned through extensive training, the authors recognized the potential for leveraging LLMs to identify vulnerabilities, execute custom exploits, and even act as virtual sparring partners to their human counterparts. This integration could provide guidance to not only "empower existing human security testers," but could also "counteract the lack of sufficiently educated security professionals," thereby addressing a current critical shortage of skilled experts in the field [4].

Summary

With the aim of determining to what extent security testing can be automated through LLMs, the authors framed their research question around the deployment of these models as virtual sparring partners for security professionals. To provide a structured framework for their investigation, they turned to MITRE ATT&CK, a comprehensive repository of knowledge concerning threat actors in the cybersecurity domain. Their goal was to produce a proficient sparring partner, capable of covering a diverse array of tactics, techniques, and procedures (TTP) summarized within ATT&CK [4].

To comprehensively explore their hypothesis, the authors led a series of experiments, where they conducted demonstrations with both high- and low-levels of guidance. These demonstrations vary in detail and specificity, with high-level addressing general PT aspects and low-level dealing with more detailed, practical actions. For their high-level demonstration, they employed LLMs to assist the planning phase of a PT. This involved tasking the LLM with designing the test itself, including determining TTPs and identifying potential vulnerabilities [4]. They then explored low-level guidance, during which they engaged the LLM to assist in the execution phase of the PT. As it is assumed that pentesters have completed their high-level analysis by the time they begin a low-level analysis, this stage is often in a step-by-step format and includes activities such as identifying systems, targeting specific vulnerabilities, executing custom commands and exploits, and providing information on how to escalate privileges.

Methodologies

In pursuit of answering their research question and exploring their hypothesis further, the authors aimed to demonstrate the extent and effectiveness of deploying LLMs as virtual sparring partners. In order for the LLMs to meet the authors expectation of success, the models must produce valid and "suitable tactics and corresponding techniques [4]." To test the practicality of using LLMs as sparring partners, the authors built upon the framework established in their research question with carefully designed experiments that encompassed both levels of guidance. Their approach ranged from broad and theoretical to highly specific and practical, which allowed them to assess the capacity and applicability of these models.

While the traditional approach to leveraging LLMs in PT requires human testers to manually initiate conversations using prompts, the authors sought to automate this process by using pre-trained Autonomous AI Agents: AutoGPT and AgentGPT. Not only do these agents increase productivity, but the incorporation of "external knowledge and automated feedback" can mitigate the occurrence of fact inventing, referred to as hallucinations [4]. Each tool can operate independently, eliminating the need for constant human intervention. This is accomplished by automatically breaking down predefined tasks into smaller, specialized subtasks through the use of "self-prompts [5]." However, despite their simulation similarities, AutoGPT is described as having more decision-making capabilities than AgentGPT, while AgentGPT offers a more user-friendly experience that welcomes a wider range of users [6]. However, as both AutoGPT and AgentGPT can successfully accomplish an assigned objective from a single directive, they are considered valid options for PT.

In the high-level experiments, the authors focused on the LLMs' potential to provide strategic guidance for both a general and specific target using both autonomous agents. For the general scenario, they provided AgentGPT with the task of "becoming a domain admin in an Active Directory," and for the specific target, they tasked AutoGPT with creating a PT plan [4]. These experiments were considered successful as both AI agents provided responses which were "realistic, and feasible, and would give a penetration tester good feedback about potential attack vectors [4]." However, it is important to note that while AutoGPTs functionality also enabled it to crawl the target's website, it declined to perform certain actions, citing ethical concerns.

In contrast, the low-level guidance experiments focused on providing step-by-step guidance, offering detailed actions such as identifying and exploiting system-specific vulnerabilities, executing custom commands and exploits, and providing insights on privilege escalation. At this stage, it was assumed that pentesters had already completed their high-level analysis, obtained some basic level of access to the system, and simply required guidance to escalate to root. Therefore, the goal of this experiment was to achieve privilege escalation and gain root access on a deliberately vulnerable Linux Virtual Machine (VM). The authors used python to set up a connection between GPT3.5 and the vulnerable VM and asked the LLM to analyze the VM's state, generate commands or actions, and potentially control or influence the VM's behavior. The script operated in an infinite loop, instructing GPT3.5 to suggest Linux shell commands, execute them over SSH on the vulnerable VM, analyze the command and its output, identify potential security vulnerabilities, and finally provide steps on how to exploit them. The results showed that GPT3.5 successfully obtained root privileges, identified and exploited security vulnerabilities, and retrieved essential system files for privilege escalation [4].

Main Findings

During the experiments, the researchers found that the LLM displayed signs of understanding causal relationships and exhibited a degree of logical thinking in its suggestions for PT tasks. These suggestions followed logical patterns, even when specific information about the target system's configuration or vulnerabilities was not provided. The authors highlighted that these suggestions, while "eerie", were primarily generated "based upon pattern-matching and not on a deeper understanding" of the subject matter [4].

The authors also found that, on a small scale, the performance of LLMs appeared unstable and inconsistent, and often produced a large variation in generated commands and identified vulnerabilities. During individual and short runs, the LLM would become too fixated and overly focused on a specific detail and lose sight of the broader picture, similar to "going down a rabbit hole [4]." While extending or combing results from multiple runs led to more consistent outcomes, LLMs were deemed less predictable and consistent compared to traditional enumeration tools like linpeas.sh in their current state.

LLMs were also found to be limited by their ethical filters, which prevent the AI from generating responses or taking actions that could engage in unethical behaviors. This was shown during the experiments when AutoGPT refused to execute additional network scans or phishing attempts. The authors found that many of these restrictions could be bypassed by running the LLM locally or by using prompt

engineering to test slight prompt variations and reduce triggering ethical filters. The simplicity of engineering prompts was shown when the authors requested "verification commands for vulnerabilities" instead of "exploits for vulnerabilities" and when they instructed the AI not to "ask questions or provide judgments [4]." While these techniques prove effective in reducing ethical denials, they also raise concerns about potential misuse. Due to the ease and accessibility of LLMs, they can be employed by both legitimate security professionals and malicious actors.

While the experiments with LLMs have showcased their potential in providing valuable PT guidance, there remains a pressing need for further refinement in their application. The findings indicate that LLMs, although proficient at pattern recognition and generating suggestions, still rely heavily on data-driven responses rather than a true comprehension of security systems. Addressing the challenges of occasional hallucinations and variability in single runs, especially when overly focused on specific aspects, is crucial to ensure their reliability. However, the urgency to incorporate AI in PT is crucial. As the field faces a critical shortage of skilled security professionals, it becomes increasingly vital that the relationship between pentesters and AI is strengthened. As the cybersecurity landscape evolves, preparing for AI-driven attacks becomes not only a necessity but also an opportunity for the industry to stay ahead in the ongoing battle against emerging threats.

Relevance to Your Course Content

This paper aligns with my course content by exploring the integration of a familiar AI model, specifically LLMs, into the field PT. As the LLMs discussed in this paper are among the most recognizable AI models, they provide an ideal starting point for introducing the central theme of my course. Their familiarity offers a comfortable and approachable introduction to 'Harnessing Artificial Intelligence (AI) for Penetration Testing.' Furthermore, this report not only engages in theoretical discussions, but also delves into the practical application of popular prompt-response techniques within PT. Through tangible examples, it illustrates how AI can enhance various facets of PT, offering both a relevant and captivating perspective to witness firsthand the transformative potential of AI from the outset.

Additionally, the paper introduces important ethical dilemmas that arise when AI is used as a tool in PT. Ethical considerations hold great significance in the cybersecurity domain, and addressing these issues early on is crucial. The report explores the effectiveness of prompt engineering, raising questions of its acceptability and ethical boundaries. It also addresses the accessibility of these powerful tools to both security professionals and malicious actors, prompting considerations about the distinctions between their respective ethical codes. Exploring and understanding these ethical complexities is a vital step to exploring PT.

B. Review 2: Gathering Information with AI and Reinforcement Learning

Introduction

The report by Ghanem and Chen focuses on the initial step of PT, known as Gathering Information; its primary focus is on how the integration of AI, particularly RL, can revolutionize this critical phase. RL has quickly become one of the most important PT advancements, resulting from the recent integration of AI and cybersecurity. This transformative approach to ML enables systems to learn through experiences from interactions with their environments. The incorporation of RL into automated PT techniques not only increases productivity, but also limits common human errors. However, existing automation systems have limitations in their scope and optimization that result in an inability to comprehensively address all potential threats while efficiently managing resources. Recognizing these challenges, Ghanem and Chen's research paper sets forth to employ ML techniques in the development of an Intelligent Automated Penetration Testing System (IAPTS) that will be "capable of imitating human PT experts in performing an intelligent and automated pen test [1]."

Summary

This research delves into the complexities of PT, an area that humans themselves often find challenging. The authors emphasize that blind automation, which entails complete automation without any human intervention, is impractical. This is particularly true during the initial phases of PT as the explorative nature often yields incomplete conclusions and requires continuous revisitation/changes in approach. As such, utilization of AI at this stage tends to result in uncertainty. However, the authors suggest that by using RL to automate these phases intelligently, it can more closely resemble a human expert's decision-making process.

The challenges associated with automation in PT are not new, as autonomous systems have been employed in the industry for some time. However, these current systems often require substantial hands-on guidance, extensive time and resources, and are limited to smaller networks. Especially considering "PT should be repeated and performed on a regular basis to ensure continuous security," Ghanem and Chen's work suggests that intelligent automation holds the key to significantly improving various aspects of PT [1]. These improvements would not only reduce the cost of manual, repetitive, and methodical testing but could also make PT more efficient and targeted. This streamlining and automation of repetitive tasks would reduce testing time, foster adaptability, and facilitate the exploration of innovative and unconventional techniques.

With this objective in mind, the authors advocate for the use of RL in PT, noting that RL aligns well with the "goal-directed learning and decision-making processes" required in the PT context [1]. Unlike manually created rules and configurations, RL learns through the consequences of its interactions, focusing on long-term goals rather than short-term fixes. This emphasis on RL represents a crucial step in addressing the challenges posed by PT automation and is converted into a formal computational model known as a Partially Observed Markov Decision Process (POMDP).

Methodologies

The methodologies employed in Ghanem and Chen's research revolve around the innovative application of RL within the framework of POMDP. This approach seeks to address the challenging PT scenario where an "agent cannot determine with full certainty the true state of the environment" by encompassing essential elements such as state observations, selection policies, dynamic transitions, and rewards [7]. Within this framework, an RL agent learns to make decisions based on its observations, with the goal of maximizing cumulative rewards. The strategies executed by the RL agent that returns the largest reward value are then

stored in memory for similar cases in the future, thus enabling it to autonomously tackle complex PT problems.

Ghanem and Chen tackle these challenges by integrating a combination of advanced algorithms, PERSEUS and PEGASUS, which are specifically designed for solving POMDPs. PERSEUS, a "randomized point-based value iteration" algorithm, simulates various random scenarios to obtain a set of educated guesses, which is referred to as a belief set [1]. These guesses represent possible situations or states of the environment based on the limited information available to the AI agent. This understanding is then improved gradually, as the algorithm updates its belief set after every simulation to ensure that each value either improves or at least remains constant [7].

Alternatively, the PEGASUS algorithm is a policy search method that seeks to determine optimal sequences of actions, known as policies, that maximize cumulative rewards over time. It transforms the problem into an equivalent deterministic POMDP, where each state-action pair has only one possible outcome. PEGASUS then conducts a set number of simulations, iteratively refining the policies to maximize their estimated cumulative reward value [7]. This approach is particularly effective in solving large POMDPs, making it suitable for addressing the challenges posed by PT, as it contains a "polynomial rather than exponential" time complexity, making it suitable for large-scale PT scenarios [1].

During the learning process for their proposed system, IAPTS relies on human input as experts provide knowledge to teach the system. However, over time, the system evolved, gaining the potential to develop autonomous learning modules that reduce the need for manual intervention. This evolution aligns with the various operational modes of IAPTS ranging from fully autonomous (Level 4) to learning mode (Level 1), where a human expert performs PT while the system observes and learns.

The primary goal of testing IAPTS was not only to evaluate its capabilities but also to demonstrate the suitability and effectiveness of applying RL to PT. The researchers conducted two main types of tests within controlled environments: Simple Simulation and Experience Replay. In the Simple Simulation, they set up a simulated network consisting of seven machines (M0 to M6) to mimic real-world PT scenarios. This allowed them to gain insights into how IAPTS would perform under various conditions, measure its performance metrics, assess execution times, and identify potential weaknesses. Alternatively, for the Experience Replay tests, the researchers simulated scenarios in which the same network underwent updates and upgrades. These tests aimed to evaluate how well IAPTS learned and adapted to changes in the network, further confirming its potential for automating PT processes.

Main Findings

The main findings of Ghanem and Chen's research paper provide valuable insights into the field of PT. In their Experience Replay tests, they discovered that the system successfully learned and stored knowledge from previous tests, with policies being effectively reused in most instances. This highlights the system's adaptability and capability to learn from past experiences, a crucial feature for PT automation. When compared to traditional manual methods, which rely on human expertise, and the blind automation approach, where tasks are automated but lack intelligent decision-making, IAPTS, significantly reduces the time required for testing and outperforms both approaches in terms of efficiency and effectiveness. This not only saves time and resources but also generates alternative attack strategies that humans may overlook.

The RL-generated attack policies also proved to be highly relevant and accurate, especially when targeting the most secure machine in the network. These policies were deemed plausible and realistic, mirroring how actual attackers might approach and execute an attack on the target system. Additionally, IAPTS was intentionally designed with flexibility in mind, permitting the seamless incorporation of new features and functionalities in the future. This modern design ensures IAPTS remains a versatile and evolving tool in the field of PT, through continual enhancement of its performance and capabilities.

Relevance to Your Course Content

This research paper offers a comprehensive overview of PT, including its purpose, advantages, disadvantages, and the intricate challenges associated with its pertinence to the first step in PT - Gathering Information. By emphasizing the extensive data collection and assessment required during manual execution of this phase, the authors highlight the necessity for discussions on automation in AI. Through practical simulations, the authors demonstrate how these solutions can significantly reduce human effort, enhance accuracy, improve adaptability, and expedite tasks, ultimately proving that automation can make the PT processing more efficient.

This report also introduces advanced techniques, such as RL and POMDPs, within the context of PT. RL, being a subset of AI, holds particular relevance in automating various phases of PT. It also highlights the practicality and adaptability of RL by exploring its application in partially observable environments, utilizing belief sets instead of the Q-tables addressed later in a fully observable scenario. This incorporation of RL and POMDPs in partially observable environments not only signifies the direction of future research but also illustrates that automated PT is an evolving field marked by ongoing developments. As such, this paper not only demonstrates the current achievements, but also serves as a preview of the extensive possibilities and potential advancements within the field.

Therefore, incorporating this paper into my seminar course can provide additional context to PT, especially in the initial Gathering Information phase, and explain how advanced AI-driven techniques, such as RL, are transforming the field. My goal is that it will ultimately serve as a useful resource to introduce the challenges of PT, automation as a solution, and the application of AI in enhancing cybersecurity practices.

C. Review 3: Enhancing the Scanning Phase with GyoiThon

Introduction

The research paper "Penetration Testing Procedure using Machine Learning" focuses on the second phase of PT - the scanning phase, with a particular focus on assessing the effectiveness of GyoiThon. GyoiThon is a PT tool integrated with ML capabilities, specifically leveraging the Naïve Bayes algorithm, that primarily focuses on automating data acquisition from target URLs [8]. This integration represents a significant advancement within the field of cybersecurity, as

it not only enhances the speed and efficiency of vulnerability detection, but also introduces the potential for more precise identification of security weaknesses. By leveraging other PT tools to enhance its capabilities, GyoiThon extends its utility beyond traditional methods. It automates the process of gathering data from target URLs, thus streamlining the scanning phase while reducing the time and effort required by pentesters.

Summary

In this study, the researchers set out to address the fundamental research question: 'How effective is the GyoiThon tool in detecting vulnerabilities [8]?' The hypothesis guiding this exploration speculates that PT tools integrating ML algorithms will exhibit greater effectiveness in searching for and identifying vulnerabilities compared to their non-ML counterparts. To highlight this premise, the paper briefly examines common ML-based PT methods used in the field, including tools known for simulating real-world attacks, detecting vulnerabilities, and addressing security weaknesses. Their analysis provided valuable context and benchmarks for evaluating GyoiThon's performance and offers support for their hypothesis regarding the capabilities of using ML in PT.

This exploratory study places a particular emphasis on comparing GyoiThon's default mode with its ML mode, executing each of them within controlled environments. Through these experiments, the researchers explore the capabilities of GyoiThon and showcase its ability to enhance PT. By exploring the effectiveness of GyoiThon, the authors assess its efficiency in detecting known vulnerabilities, identifying software components, discovering configuration weaknesses, highlighting authentication issues, and pinpointing general web application vulnerabilities [8]. These capabilities emphasize GyoiThon's pivotal role in the scanning phase of PT, highlighting the demand for advanced tools and techniques to navigate the complex landscape of cybersecurity.

Methodologies

The methodology employed in this report is particularly significant as it delves into a novel area of interest within cybersecurity. It's worth noting that this comprehensive study of GyoiThon represents a unique endeavor, as the only prior study into the capabilities of the tool was conducted by its developer. As such, the researchers had the distinct advantage of operating within a flexible framework that lacks predefined steps, which enabled them to create new procedures to address their research question [8].

As for execution, the researchers established an isolated testing environment using the Kali Linux operating system within VirtualBox. Within this controlled environment, GyoiThon was employed to detect vulnerabilities related to data exchange; it analyzed both unencrypted HTTP traffic on Port 80 and encrypted HTTP traffic on Port 443. The target websites were hosted on a server provided by OWASP and accessed via a locally hosted environment.

Since the researchers' hypothesis centered on comparing PT tools with and without ML algorithms, their analysis was limited to GyoiThon's Default Mode and ML Mode. The Default Mode encompassed various steps, including parsing HTTP responses, identifying product/version information, assessing vulnerabilities using Common Vulnerabilities and Exposures (CVE) numbers, examining HTML and JavaScript comments, analyzing debug messages, and assessing login pages [9]. In contrast, the ML Mode incorporated all the steps from the Default Mode, but additionally utilized the Naïve Bayes algorithm for product/version identification [9]. This setup enabled researchers to directly evaluate the effectiveness of the Naïve Bayes algorithm in the realm of PT, aligning with their hypothesis.

Main Findings

The analysis of the PT procedure conducted using GyoiThon revealed several significant insights. First, it was observed that Port 80, commonly associated with unencrypted HTTP data, exhibited a higher number of vulnerabilities for both the Default and ML modes. This finding aligns with expectations, as Port 80's lack of encryption renders it less secure compared to HTTPS (Port 443). This absence of encryption causes Port 80 to be more susceptible to vulnerabilities and potential attacks, as was reflected in the test results. However, the variation in the number of vulnerabilities detected between these ports decreased with the use of ML mode; by identifying three additional vulnerabilities in Port 80, ML mode reduced the disparity in vulnerability frequency from six to only three [8]. Not only does this outcome highlight the potential of GyoiThon, but it also supports the hypothesis that integrating machine learning into PT tools enhances their effectiveness in identifying vulnerabilities.

While these initial results demonstrate success, it is essential to note that GyoiThon relies on external sources, such as the National Vulnerability Database (NVD), to gather information about vulnerabilities. This reliance is a limitation of the tool's capabilities as it may be unable to identify vulnerabilities that have not yet been documented in the NVD. This potential blind spot highlights the importance of staying updated with emerging threats and identifies an aspect requiring improvement. As such, while GyoiThon showcases promise as a valuable PT tool, the researchers explain that future testing against real websites and a comprehensive assessment of all nine modes is necessary to obtain a more comprehensive understanding of its capabilities [8]. These findings contribute to the ongoing development of AI-driven PT tools and emphasize the need for continuous refinement to stay ahead of evolving cyber threats.

Relevance to Your Course Content

This report extensively explores the application of ML in PT, using GyoiThon as a focal point, and aligns with the central theme of my course. With our aligning goals of providing a thorough understanding of AI techniques for cybersecurity, this report serves as a valuable reference for my course by delving into the practical application of a specific ML tool. This significance is particularly true for the second phase of PT, scanning, which is the primary focus of my second module and the explicit function of GyoiThon, which effectively identifies vulnerabilities by scanning web pages.

Through a comparative analysis between default PT methods and those augmented with AI, this article showcases the effectiveness of AI-driven approaches through direct evidence. This novel and practical study not only highlights the superiority of ML-enhanced techniques but also emphasizes the potential transformative power of AI within the cybersecurity domain. Through empirical evidence, this report encourages further exploration into the integration of AI into the field of PT and invites active engagement for modern AI security solutions.

Arguably most importantly, the article delves into the discussion of common vulnerabilities found in web applications and the various tools used to detect them. This practical understanding of vulnerabilities and the tools and techniques available for their detection and mitigation is essential for effectively navigating the complex digital domain.

D. Review 4: Exploitation in PT with RL

Introduction

In the paper titled "Vulnerability Exploitation Using Reinforcement Learning," the authors leverage modern PT techniques, specifically ML and RL, to automate one of the most critical phases in cybersecurity: exploitation. By prioritizing actions that maximize rewards, RL underscores the importance of developing tools that not only identify vulnerabilities but also utilize ML to efficiently exploit them. The authors focus goes beyond automation and emphasizes the need for further evolution in PT to address the complex field of cyber security.

The intelligent agent created in this report prioritizes adaptability, ensuring it can be trained on a wide array of vulnerabilities and operating systems. This approach offers a tailored and intelligent approach to exploitation that challenges traditional methods, which often involve resource-intensive, brute-force techniques that are time and resource intensive [2]. To accelerate the PT process and ensure a more targeted and efficient approach to identifying and exploiting vulnerabilities, this agent leverages Metasploit, a well-known PT tool with a wide range of payloads for various purposes.

What further sets this approach apart is the agent's ability to archive successful exploits as states alongside corresponding payloads with high success rate. The agent then intelligently leverages this payload repository, known as a Q-table, to execute exploitation with precision – a milestone that demonstrates the potential of RL to leverage an award system and continuously refine and enhance exploitation strategies using AI. This report provides a look into the future of PT, where customization, adaptability, and intelligence combine to not only identify vulnerabilities but to masterfully exploit them.

Summary

In this report, the authors utilize ML to create an RL agent that makes decisions by interacting with a fully observable environment. The primary focus of this RL agent lies in the exploitation phase, the third and crucial step in PT. Through an extensive training process, the agent interacts with a simulated environment, dynamically adapting its exploitation strategies by analyzing various factors, including the environment configuration. This adaptive approach is made possible by representing the environment as states, each defined by a unique combination of operating system and vulnerability [2]. These states are then linked to payloads that have demonstrated a high likelihood of success and are stored in a Q-Table. Due to the variability in payload effectiveness based on these states, the authors reward successful attempts, which they define as "the establishment of a reverse shell session following payload execution [2]." Therefore, even in instances where the payload is not successful, the RL agent adjusts its decision-making process based on the rewards it receives; it then learns to prioritize actions that result in positive rewards.

Once the RL agent is trained, it is deployed in a real-world scenario where it encounters target systems with specific operating systems and vulnerabilities. Metasploit serves as a valuable resource as the RL agent selects and utilizes payloads based on its learned strategies, facilitating effective delivery of exploits to compromised target systems. The extensive payload options offered by Metasploit enhance the agent's versatility during the exploitation process. This integration contributes to the authors primary goal of creating a versatile "general agent that is capable of exploiting any/general task and making the appropriate decision [2]."

This combination of ML, RL, and established PT tools represents a significant advancement in the merging of AI and cybersecurity. Through the incorporation of RL algorithms and their integration with established tools like Metasploit, this report demonstrates an evolution of PT. This innovative approach showcases the potential of AI-driven agents to optimize and streamline exploitation tasks, ultimately benefiting cybersecurity professionals in identifying and addressing vulnerabilities in a more efficient and effective manner.

Methodologies

The methodologies employed in this study consist of two important phases, the Training Phase and the Exploitation Phase. During the Training Phase, an intelligent agent is developed through the application of RL techniques, using a guess-and-reward system. This phase involves the agent navigating a simulated environment, in which it uses an "epsilon greedy strategy" to make informed decisions by balancing exploration (delivering a randomly selected payload) and exploitation (selecting a specific payload that will yield the highest expected reward based on its learning so far). The agent then receives rewards based on the success or failure of a particular payload, from which it builds a valuable repository of previous exploits and their results. The training phase is then repeated for a certain number of iterations, with a gradual decrease of exploration.

To motivate its decision-making, a point-based reward system is employed that offers substantial rewards for success and imposes penalties for failures. These rewards are maximized by leveraging the Q-learning algorithm, to "determine the best series of actions to take based on the agent's current state [10]." This approach often results in the agent executing calculated and cautious actions to minimize risks [2].

The learning phase honed its exploitation skills across seven trials, during which the agent spent an average of 2.5 hours executing 500 attempts to exploit vulnerabilities. During this phase, the agent's primary focus was on continuous learning and strategy refinement. It actively experimented with different actions, assessing their success or failure, and served to provide insight into valuable tuning parameters from controlling the importance of new versus old information, long-term versus short-term rewards, to exploration vs exploitation [2]. An assessment of the agent's performance is then calculated to determine how effective it is at establishing a reverse shell. This computationally intensive process positively reflected the agent's ability to actively learn and adapt its exploitation techniques by making informed decisions.

In the exploitation phase, the RL agent took advantage of its learned strategies, drawing insights from its repository to

effectively select payloads from the Metasploit framework. To simulate real-world scenarios, it was deployed on multiple vulnerable machines with a "remote code execution" [2] vulnerability found in Apache CouchDB, specifically Version 3.1.0. The agent's primary objective was to establish a reverse shell, which was achieved with remarkable efficiency by leveraging payloads with the "highest rank in the Q-Table [2]." Impressively, it accomplished this goal in an average of just 8.26 seconds across the tested systems. This performance indicates that the training phase prepared the agent well and proved its ability to effectively execute learned strategies against real-world systems.

Main Findings

The study's main findings highlight the remarkable effectiveness of the RL agent in automating exploitation tasks, particularly within the realm of PT. As the RL agent gains experience through training, it exhibits a gradual shift from exploration to exploitation, becoming more discerning in its actions. For example, while it initially explores new actions to gather information, over time it prioritizes actions it has deemed effective for achieving its goals. This transition, combined with the selection of optimal parameters, consistently resulted in an average success rate of 83.64% and an average exploit time of 8.26 seconds [2]. These notable statistics highlight the potential of the RL approach to significantly reduce the time and resources required for PT, presenting a novel and cost-effective solution to the challenges of vulnerability exploitation.

In contrast to traditional exhaustive testing methods, which often follow rigid approaches, the RL agent's adaptability and capacity for fine-tuning its strategies prove beneficial. By focusing on maximizing overall rewards and balancing learning and randomness, the RL approach proves more efficient and effective in verifying exploitable vulnerabilities. In summary, the main findings of this study emphasize the RL agent's aptitude for automating exploitation tasks, its proficiency in achieving PT objectives, and its potential to revolutionize vulnerability assessment practices.

Relevance to Your Course Content

This study explores modern techniques in cybersecurity, highlighting the innovative use of RL algorithms for vulnerability exploitation and emphasizing the field's dynamic nature. The authors take a comprehensive approach as they explore not only the capabilities of RL but also its adaptability. Notably, they explore versatile fine-tuning options, such as learning rate and exploration rate, and provide insights into the impacts of these methods. The study also examines RLs application in fully observable environments, utilizing Metasploit and Q-tables instead of the previously mentioned POMDP and belief sets in partially observable scenarios. This multifaceted exploration demonstrates how RL techniques can be adapted and leveraged effectively across different cybersecurity scenarios, aligning seamlessly with our course's goal of understanding AI techniques in cybersecurity.

Additionally, since RL consistently selects the most effective actions to maximize rewards, it directly addresses a critical aspect of Penetration Testing, particularly in Step 3 - Exploitation. By prioritizing the actions that yield the highest rewards, RL showcases the importance of developing similar tools that not only identify vulnerabilities but utilize ML to efficiently exploit them. Overall, this research broadens perspectives on the possibilities within the field of cybersecurity and highlights its crucial role in staying current with the dynamic landscape of digital threats.

E. Review 5: Automating Post-Breach PT with RL

Introduction

In the report, "Automated Post-Breach Penetration Testing through Reinforcement Learning," the authors introduce the concept of using RL, a subset of ML, to automate the post-breach phases of PT. Theses phases occurs after the initial breach of a system and focus on privilege escalation, maintaining persistence, and further exploration [3]. This approach aims to automate and enhance the capabilities of an AI agent, allowing it to navigate and effectively interact with diverse network environments. The agent is trained through interactions with various networks and prioritizes the balance of exploration and exploitation. The importance of this study is emphasized as there is still a substantial lack of testing for automation in the post-exploitation phase [11]. This lack of research is particularly dangerous as current PT practices are rapidly growing in complexity and resource consumption. In an attempt to mitigate these challenges, researchers explore the application of AI techniques, specifically the Deep Reinforcement Learning (DRL) subset, Deep Q-learning (DQ). They hope that by leveraging neural networks to directly map input states to action-Q-value pairs, their agent will excel in navigating complex environments without requiring a detailed model of the environment [10].

Summary

This report highlights the current limitations in the field, pointing out that even with the use of automated tools, current PT practices remain complex and resource intensive. In response to the limitations of the popular RL algorithm, Q-learning, and driven by recent advancements in deep Q-learning (DQ) algorithms, the authors made a deliberate choice to adopt DQ as their training model. Q-learning struggles when tasked with handling intricate systems or environments as it becomes computationally expensive to maintain the Q-table [10]. To overcome these challenges, the authors explore the implementation of alternative approaches such as DRL.

DRL has emerged to address these challenges and offer more solutions for RL in larger and more intricate environments. Instead of relying on a detailed model of the environment, DQ learns directly from interactions with the environment by utilizing neural networks. These networks take the current state as input, produce estimated reward values known as Q-values for all possible actions in that state, and associate each action with a unique Q-value, where higher values indicate more favorable actions. In other words, DQ distinguishes itself from traditional Q-learning by replacing the Q-table with a neural network that directly produces recommendations for actions based on the current state [10]. The agent then employs a decision-making process that involves comparing these Q-values and selects the action linked to the largest reward. These rewards are then used to adjust and refine the Q-values in both traditional Q-learning and DQ. Just as it does in the traditional method, these Q-values become more accurate over time and enhance the agent's ability to make decisions.

Overall, while the paper primarily presents a conceptual framework, it lays the groundwork for practical applications of AI in post-exploitation cybersecurity. It introduces key concepts such as Q-value estimation, exploration-exploitation

balance, and the importance of realistic training environments. The paper's focus on future research and development suggests RL's potential to shape the future of AI-driven PT.

Methodologies

For their research, the authors propose an architecture which involves employing a Deep Q Network with TF Agents, constructed on the TensorFlow library as the fundamental framework for training the RL agent. TF Agents is a specialized, modular software that leverages the capabilities of the TensorFlow framework [12]. By building upon TensorFlow and incorporating TF Agents, they ensure a solid foundation for training the AI agent by harnessing the power and flexibility of these frameworks.

To create a realistic PT environment, the authors plan to deploy the agent in virtualized Linux and Windows servers. These environments were crafted to simulate authentic cybersecurity scenarios, effectively mirroring computer networks tailored for cybersecurity training and competitions. Emulating target data, these environments included critical elements commonly found in these networks such as password files, shadow files, and system configurations. The agent's actions are confined to a predefined list of terminal commands, with the specifics of these commands being adapted based on the agent's observations within the environment. The resulting performance will then be gauged based on its adeptness at exploration and exploitation within these environments, with scores serving as rewards critical for reinforcement [11]. The balance between exploiting actions that appear promising based on its current knowledge (Q-values) and exploring new actions to discover potentially better strategies is carefully managed to optimize the agent's learning process and overall performance.

To establish a performance baseline for their research, the authors created a Python script designed to locate files within simulated environments. This script serves as a reference point to assess the AI agent's performance, enabling a direct comparison between the agent and the script's capabilities. This python script is tested across the servers to determine its effectiveness in locating files, particularly focusing on configurations and log files. To assess the trained agent's effectiveness, a comparison will be drawn between the script's performance and that of an RL agent "created using randomized policy [11]." This comparison will ultimately provide insights into the efficiency of the trained agent's policy.

The methodologies discussed in this report offer a comprehensive plan for training and evaluating an RL agent for automated post-breach PT. This plan encompasses Deep Q-Learning, specialized architecture, simulated environments, action-reward configuration, baseline testing, and performance comparison. These methodologies are integral to the successful development and assessment of the proposed automated PT approach.

Main Findings

One of the primary findings of the report is the recognition of the applicability of RL, specifically DQ, in the domain of PT. The report emphasizes that RL offers a promising avenue for automating the post-breach phases of PT, a field where modern practices are lacking research and traditional practices are quickly becoming obsolete. While this report focuses on conceptual feasibility of this incorporation, it directly addresses the beginning steps required to train an RL agent to perform tasks in a compromised network environment. This discovery is significant as it not only validates the role of ML in enhancing cybersecurity measures but also opens doors to the development of more efficient and effective PT methodologies, especially in regard to the post-exploitation phase.

Through the analysis of automation of the post-exploitation phase, the authors set the stage for ongoing and future investigations. The report communicates the researchers' intention to implement the proposed approach, train the RL agent, and expand the model's applicability to a broader spectrum of network environments. While this report primarily focuses on conceptual feasibility, it is also practical and action-oriented, with the aim of making meaningful contributions to the cybersecurity field by advancing the automation of PT practices.

Relevance to Your Course Content

This report highlights the critical application of AI, specifically RL, in automating and optimizing the post-breach stages of PT. Notably, it offers practical examples that demonstrate how AI can be effectively deployed during this critical phase and advances uncharted territory in doing so. This novel discussion addresses the imperative need for cybersecurity professionals to remain current and adaptive in the face of evolving threats and encourages innovation.

Additionally, this study provides insights into the advanced technologies that power AI-driven PT, notably highlighting the significance of DQ and its role in improving traditional Q-learning methods. As ML and deep learning are at the core of AI's capabilities, delving into these technological intricacies establishes a robust foundation to comprehend how AI models are constructed, trained, and effectively deployed. This comprehensive understanding is essential as it empowers individuals to make informed decisions, adapt AI tools to specific cybersecurity challenges, and innovate within the field, ultimately contributing to the ongoing evolution and effectiveness of AI-driven PT practices.

INCORPORATION OF FINDINGS INTO THE COURSE

The findings extracted from this literature review serve as a strong foundation for developing the short seminar course focused on harnessing Artificial Intelligence (AI) for Penetration Testing (PT). A key takeaway from the literature is the growing importance of AI-driven PT tools, which encompass AI-enhanced vulnerability detection, network scanning, exploit selection and execution, and post-exploitation management. As these are all crucial aspects of the PT process, the practical application of these tools, as highlighted in [9] and [2], emphasizes the importance of integrating a combination of demonstrations and practical exercises into the course.

Additionally, the literature review sheds light on the ethical dilemmas faced when employing AI in cybersecurity, as discussed in [4]. This insight alludes to the importance of discussing responsible and ethical practices within the course, ensuring that it emphasizes not only technical proficiency, but also an awareness of the ethical considerations associated with PT. Furthermore, as ethical considerations hold great significance in the cybersecurity domain, it is important to address these issues early in the course.

The research also emphasizes the importance of executing the PT steps thoroughly and completely. For example, when

tackling the first step, Gathering Information, it is important to determine whether the investigation targets a fully or partially observed environment. The literature, particularly [1] and [2], show how this knowledge can impact the exploit techniques used. These understandings will be harnessed throughout the course to encourage a deeper comprehension of the intricacies that incapsulate each PT phase, from gathering information to exploiting vulnerabilities, and will provide the basis for structuring the course modules according to the PT phases.

These discoveries, which acknowledge the growing demand for AI-driven PT tools, ethical considerations, the imperative for a comprehensive understanding of PT steps and techniques, and the need for hands-on experiences, provide valuable preliminary course ideas. Beginning with Module 1, I plan to draw information from [4] to build a foundation and introduce AI, ML, and deep learning as they pertain to PT. After building foundation, this module will focus on the essential first step of penetration testing – Gathering Information. Inspired by [1], this module will cover the limitations of existing automation data acquisition techniques and discuss modern ML techniques that challenge these, such as IAPTS.

Module 2 will follow the pattern of PT phases and focus on the second step of PT: Scanning. By building on the foundational knowledge acquired in Module 1, this section will analyze various automated data acquisition techniques. While this module will cast a wide net, there will be a particular focus on GyoiThon and similar tools, following the research from [8]. My goal is to include a practical lab that will concentrate on utilizing these automated scanning tools to detect vulnerabilities efficiently, emphasizing the AI's adaptability over manual methods.

Module 3 will delve further into the PT process by addressing the third step: Exploitation. Guided by insights from [2] this module will explore how AI and ML can be practically applied to identify and exploit vulnerabilities. It will emphasize the role of AI-driven PT tools like Metasploit to ensure a more targeted and efficient approach. To provide hands-on experience and a better grasp of these exploitation techniques, this module will incorporate ethical practical exercises.

The final module will shift its focus to the post-exploitation phases of PT by building upon the research in [11]. This module will integrate the knowledge acquired in previous modules to address PT steps 4, 5, and 6, with an emphasis on maintaining persistence, privilege escalation, and further exploration. Given the limited research in this area, this module will explore potential future trajectories for the post-exploitation phase. Drawing inspiration from [1], [2], and [11], this section will explore the role of advanced AI techniques, particularly deep learning and RL. The primary goal is to provide insights into how AI can effectively navigate and interact with compromised systems.

Structuring the course modules in this manner will provide a comprehensive understanding of AI in PT by covering the entire PT process, from information gathering to post-exploitation. This approach achieves several objectives: it introduces essential foundational knowledge, explores real-world applications, and ignites a curiosity for the evolving landscape of AI in cybersecurity.

COURSE OVERVIEW

A. Introduction

Through a curriculum structured around the PT phases, the following modules will explore the intersection of AI and PT by offering valuable insights and practical knowledge into the dynamic field of Cyber Security. With the central theme of 'Harnessing Artificial Intelligence for Penetration Testing,' topics ranging from AI-driven PT tools, ML, RL, and advanced deep learning techniques. Modules will provide an assortment of practical demonstrations and hands-on exercises to illustrate how AI enhances manual offensive strategies and highlight AI's transformative power in the world of cybersecurity.

Module 1 establishes the groundwork by introducing the fundamentals of AI in the context of PT before exploring the first PT phase: Gathering Information. Module 2 extends this knowledge into the Scanning phase, emphasizing AI's adaptability through practical hands-on labs. Module 3 ventures into the world of Exploitation, showcasing how AI and ML can identify and exploit vulnerabilities. Finally, Module 4 explores the post-exploitation phases, where the limited research landscape opens doors to innovation in the future of AI.

B. Course Table of Contents

- Module 1: Introduction to AI and ML in Penetration Testing
  - Submodule 1.1: Introduction to AI and ML in Penetration Testing
  - Submodule 1.2: Gathering Information and Reconnaissance with AI
- Module 2: Machine Learning for Vulnerability Assessment
  - Submodule 2.1: Scanning and Vulnerability Assessment with ML
  - Submodule 2.2: Exploiting and AI-Enhanced Techniques
- Module 3: Post-Exploitation AI and ML Techniques
  - Submodule 3.1: Maintaining Connection, Covering Tracks, and Reporting
  - Submodule 3.2: AI-Enhanced Post-Exploitation and Privilege Escalation
- Module 4: Deep Learning and Advanced Techniques
  - Submodule 4.1: Deep Learning and Advanced Techniques
  - Submodule 4.2: Review/Conclusion

C. Conclusion

These modules not only lay the foundational understanding of AI and its domains but also delve into its practical applications in information gathering, scanning, exploitation, and post-exploitation phases. As participants progress through this course, they will gain a deeper appreciation of the critical role AI plays in cybersecurity, all while preparing themselves for the challenges of an ever-evolving digital landscape.

D. References and Further Reading

- PT Overview: [13], [14]

- Q learning and Deep Q Learning: [15], [16], [17]
- Tools and Walkthroughs: [18], [19] [20]

Key Components of the Newly Developed Course

The short seminar course on Harnessing Artificial Intelligence (AI) for Penetration Testing (PT) is designed with several key components to ensure a comprehensive learning experience. The primary learning objectives of the course are to equip students with a deep understanding of AI's role in PT and to develop practical skills in utilizing various popular AI-driven tools. This course is designed for undergraduate-level students, early-career professionals, and other digital enthusiasts seeking to strengthen their cybersecurity skills. It is customized for individuals with a foundational understanding of cybersecurity but may have limited experience in the field of AI.

Assessment strategies within the course involve a combination of practical exercises to develop specific skills and demonstrate knowledge, reflective questions to incite further discussion, and CTFs to test a wide range of knowledge and improve problem-solving skills. The culmination of the course will be a final seminar project where students will exercise their creativity and apply the knowledge gained throughout the course to envision and design a theoretical PT tool that leverages AI to advance current methods. In this project, students will not be required to develop a functional tool but rather explore innovative ideas and propose a comprehensive design for an AI-driven PT tool. The goal of this project is to encourage students to think critically and apply AI concepts to real-world cybersecurity challenges and develop presentation skills.

The supplementary materials for this course will include research papers, online technical blogs, and access to discussed PT tools. Additionally, step-by-step walkthroughs for all course demonstrations and practical exercises will be included, ensuring students can apply the knowledge effectively. To further enrich the learning journey, links to relevant online Capture The Flag (CTF) challenges, including platforms like TryHackMe, will be provided. The linked CTFs will directly pertain to the course content, allowing students to practice application in real-world scenarios. As the course progresses, additional materials may be incorporated to ensure that students receive the most comprehensive, current, and beneficial resources available.

Conclusion

In conclusion, this short seminar course, titled "Harnessing Artificial Intelligence for Penetration Testing," is designed to provide a comprehensive exploration of the integration of AI into the field of PT. Drawing upon the insights collected from the literature review, this course aims to equip students with both the theoretical and practical knowledge necessary to navigate the complex and evolving landscape of cybersecurity. Key insights from the literature review have shaped the course content, from the structure of the course modules to an emphasis on the importance of AI-driven PT tools, ethical standards, comprehensive understanding, and hands-on experience in PT techniques. These insights highlight the critical role of AI in bolstering cybersecurity defenses, reducing human errors, and addressing the critical shortage of cybersecurity personnel. By offering a tailored curriculum structured around the PT phases, this course prepares students to not only understand the current AI landscape but also envision its future possibilities.

The potential impact of this course focuses on circulating knowledge regarding the critical need for additional research on AI-driven PT tools during each phase of PT. As students delve into the practical exercises, engage in thoughtful discussions, and undertake the final seminar project, they not only enhance their technical expertise but also develop critical thinking and problem-solving skills. This course paves the way for innovative approaches to cybersecurity, bridging the gap between AI and PT, and empowering individuals to navigate the evolving threat landscape. Ultimately, it serves as a catalyst for a more secure digital world, where AI is not just wielded by criminal hackers but harnessed by cybersecurity professionals who stay one step ahead, outsmarting and countering these threats with cutting-edge techniques to ensure the safety and resilience of our digital future.

References

[1] M. C. Ghanem and T. M. Chen, "Reinforcement Learning for Intelligent Penetration Testing," in *Second World Conference on Smart Trends in Systems, Security and Sustainability*, London, 2018.

[2] A. AlMajali, L. Al-Abed, R. Mutleq, Z. Samamah, A. A. Shhadeh, B. J. Mohd and K. M. Ahmad Yousef, "Vulnerability Exploitation Using Reinforcement Learning," in *Jordan International Joint Conference on Electrical Engineering and Information Technology*, Amman, 2023.

[3] S. Watts , *Penetration Testing: Practical Introduction & Tutorials,* 2022.

[4] A. Happe and J. Cito, "Getting pwn'd by AI: Penetration Testing with Large Language Models," in *European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, San Francisco, 2023.

[5] M. Pogla, *Auto-GPT vs ChatGPT: How Do They Differ and Everything You Need To Know,* 2023.

[6] G. Dheda, *Auto-GPT vs AgentGPT: Understanding the Differences,* 2023.

[7] M. T. Spaan and N. Vlassis, *Perseus: Randomized Point-based Value Iteration for POMDPs,* vol. 24, 2005, p. 26.

[8] R. S. Jagamogan, S. A. Ismail, N. H. Hassan and H. Aba, "Penetration Testing Procedure using Machine Learning," in *International Conference on Smart Sensors and Application (ICSSA)*, Kuala Lumpur, 2022.

[9] gyoisamurai, *GyoiThon: Next generation Penetration Test Tool,* 2021.

[10] Q. T. Luu, *Q-Learning vs. Deep Q-Learning vs. Deep Q-Network,* 2023.

[11] S. Chaudhary, A. O'Brien and S. Xu, "Automated Post-Breach Penetration Testing through Reinforcement Learning," in *Conference on Communications and Network Security (CNS)*, Avignon, 2020.

[12] TensorFlow, *Introduction to TensorFlow.*

[13] F. Abu-Dabaseh and E. Alshammari , "Automated Penetration Testing: An Overview," in *Computer Science & Information Technology - Computer Science Conference Proceedings (CS & IT-CSCP)*, Amman, 2018.

[14] Y. Stefinko, A. Piskozub and R. Banakh, "Manual and automated penetration testing. Benefits and drawbacks. Modern tendency," in *Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*, Lviv, 2016.

[15] D. Pandey and P. Pandey, "Approximate Q-Learning: An Introduction," in *International Conference on Machine Learning and Computing*, Bangalore, 2010.

[16] H. Van Hasselt, A. Guez and D. Silver, "Deep Reinforcement Learning with Double Q-learning," in *AAAI Conference on Artificial Intelligence*, Phoenix, 2016.

[17] K. Tran, A. Akella, M. Standen, J. Kim, D. Bowman, T. Richer and C.-T. Lin, "Deep Hierarchical Reinforcement Agents for Automated

Penetration Testing," in *International Workshop on Adaptive Cyber Defense*, Sydney, 2021.

[18] ZION3R, *GyoiThon - A Growing Penetration Test Tool Using Machine Learning,* 2018.

[19] D. Foti, *Metasploit: Exploitation Walkthrough TryHackMe,* 2022.

[20] M. Shivanandhan, *Metasploit — A Walkthrough Of The Powerful Exploitation Framework,* 2020.