AI for Penetration Testing: Exploring the Intersection of Artificial Intelligence and Cybersecurity

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

***Abstract*— In response to the rising complexity and frequency of cyber threats, Penetration Testing (PT) has become imperative for safeguarding digital assets. However, traditional PT methods are limited by substantial workloads, complex networks, and a critical shortage of skilled penetration testers (pentesters) - challenges that have prompted the exploration of Artificial Intelligence (AI) integration as a potential solution. This report explores current research through a comprehensive literature review, which highlights the imperative shift toward structured methodologies to enhance operational efficiency, comprehensive assessments, and vulnerability prioritization in PT. By drawing on diverse research insights, this report identifies current challenges with maintaining data accuracy, model robustness, and environmental adaptability. It then examines the interaction of three advanced AI systems designed to address these limitations: Automated Decision-Making Systems (ADMS), Self-Improving Systems, and Human-in-the-Loop (HITL). ADMS reduces human reliance and improves decision-making efficiency by leveraging deep reinforcement learning (DRL) and Natural Language Processing (NLP) to automate vulnerability identification. Self-Improving Systems implement real-time data and adaptive algorithms to enhance model robustness and adjust to environmental changes. Meanwhile, HITL systems integrate human expertise directly into the AI learning process to deliver AI-driven tools that benefit from human insight. Then, the viability and applicability of these approaches are evaluated through empirical analysis, theoretical inquiry, and practical comparisons against established PT scenarios. The hope is that by strategically addressing the critical challenges of integrating AI into PT using various advanced systems, this report can lay a solid foundation for a transition toward more sustainable and effective PT practices.**

Keywords*— *Artificial Intelligence, Penetration Testing, Ethical Hacking, Vulnerability Assessment, Machine Learning, Reinforcement Learning, Deep Learning, Automated Decision-Making Systems, Self-Improving Systems, Human-in-the-Loop

I. INTRODUCTION

In an age defined by the relentless increase of technology, the growing digital landscape has become both a playground for innovation and a battleground for cyber threats. As organizations increasingly rely on technology to operate, communicate, and store critical data, safeguarding these assets against potential adversaries becomes paramount. This realization has given rise to Penetration Testing (PT) as a vital and proactive strategy that allows organizations to simulate cyberattacks on their systems to discover and eliminate dangerous vulnerabilities. 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]."

This offensive approach to PT emphasizes the importance of following a structured process to systematically evaluate a system's security. While the specific phases of PT may exhibit variations in terminology depending on the source, their fundamental organization remains the same: preparation, implementation, and analysis [2]. However, as the complexity and scale of networks continue to expand, traditional manual methods need help to cope with the substantial workload inherent to this process. Penetration testers (pentesters) involved in this field must maintain a continuous regimen of training and practical skill development to remain at the forefront. Unfortunately, this excessive pressure, combined with the recent shortage of pentesters, has left many in the field feeling overburdened and overtired [1].

This situation has prompted a growing trend toward integrating automation technologies, including Artificial Intelligence (AI), Machine Learning (ML), and Reinforcement Learning (RL) into PT. Automated PT can significantly reduce the time and resources required for testing, making it a crucial development in the field. While extensive research has already begun to explore this integration [1] [3], many of these approaches still require manual human intervention for vulnerability identification. However, recent innovative methods have emerged that leverage neural networks to gain a deeper understanding of the intricate and dynamic security environments within modern networks, ultimately enhancing the role of AI in PT. This evolution is pivotal for cybersecurity and AI as they converge to tackle digital threats.

II. Proposed Approach

In the rapidly advancing field of cybersecurity, particularly within the realm of penetration testing (PT), the integration of artificial intelligence (AI) holds transformative potential. This section outlines a forward-thinking approach aimed at leveraging AI technologies, such as machine learning (ML), reinforcement learning (RL), and deep reinforcement learning (DRL), to enhance PT practices. It addresses the challenges stemming from automating the PT process, including overreliance on manual human intervention and complex technical hurdles such as data accuracy, model robustness, and environmental adaptability.

The project initially delves into vital research to explore existing solutions to these challenges and provides practical comparisons for advancements in each phase of the PT process. However, recognizing the novelty and evolving nature of intelligent PT, this report takes a step further to explore the theoretical implications of the most recent foundational research. It examines the integration of cutting-edge systems such as Automated Decision-Making Systems (ADMS), Human-in-the-Loop (HITL) configurations, and Self-Improving Systems. This exploration emphasizes not only the practical application and evaluation of these advanced systems, but also the theoretical basis for how they can significantly enhance existing methods. Thus, by harnessing the capabilities of these innovative systems, automated PT can better anticipate and counteract modern cyber threats, streamline operations, and mitigate errors associated with human factors.

A. Project Scope

The rapidly evolving cybersecurity landscape, with its complex and labor-intensive strategies, stands poised to significantly benefit from recent advancements in AI technology. This surge in AI's significance is crucial for cybersecurity endeavors such as PT, the linchpin of proactive cybersecurity. The integration of AI in PT has the potential to level the playing field and empower defenders to anticipate and mitigate threats effectively.

The motivation behind this project is to comprehensively explore modern threats and current PT methods by examining the integration of various AI techniques including ML, RL, and Deep Reinforcement Learning (DRL), into PT practices. This exploration is then expanded to include cutting-edge systems such as Automated Decision-Making Systems (ADMS), Human-in-the-Loop (HITL) configurations, and Self-Improving Systems, which represent the forefront of AI-driven cybersecurity solutions. These systems are integrated to address critical challenges in PT automation, enhance decision-making precision, and improve adaptability to dynamic threat environments.

At its core, this project aims to address a fundamental question: How can AI techniques be effectively harnessed throughout the PT process? This question is critical, especially with cybercriminals' increasing adoption of AI, as it demands cybersecurity professionals to adapt promptly; reliance on traditional PT methods alone may leave pentesters unequipped to combat modern threats. This project aims to benefit cybersecurity defenders, ethical hackers, security analysts, and researchers by providing insights into AI tools and techniques and suggesting ways to overcome current limitations.

Additionally, the hope is that a shift towards intelligent automation can not only reduce testing time and resources, but also mitigate many of the prevalent and "recurrent human errors" in manual PT that stem from factors such as "tiredness, omission, and pressure" [1]. This transformation in the field, as highlighted by [3], signifies the use of "advanced algorithms, machine learning, and AI to scan systems for vulnerabilities," thus offering a path toward more effective, efficient, and error-resistant cybersecurity practices.

B. Current Limitations

However, addressing the current research landscape in AI integration in PT reveals several notable limitations. Firstly, a significant gap exists in research and testing within this domain, thereby signifying an unexplored terrain ripe for investigation. While research has already begun exploring this novel integration [1] [3], many of these approaches are limited by their reliance on manual human intervention

for vulnerability identification and exploitation. While this reliance hinders the full realization of AI's potential in the PT process, transitioning towards a fully autonomous solution presents complex technical hurdles. For example, the shift to intelligent, real-time detection would require addressing data accuracy issues, model robustness, and environmental adaptability.

To ensure data accuracy, all input data must be precise and reliable. However, achieving this precision in dynamic network environments can be challenging since data is often incomplete, outdated, or biased; this is especially dangerous in PT as these inaccuracies can lead to the misinterpretation of vulnerabilities or threats and potentially result in ineffective security measures. Therefore, data accuracy is vital for ensuring model robustness. For genuinely effective threat detection, models must adapt to evolving threats and environments to perform consistently across diverse scenarios. This level of adaptability requires sophisticated algorithms that can understand and interpret incomplete or biased data while accounting for various attack vectors, software vulnerabilities, and system configurations.

The integration of AI in PT is a relatively novel field, where these challenges of ensuring accuracy, robustness, and adaptability are still being navigated and tested. Due to its infancy, research in this domain is notably limited and requires further exploration and innovation before a fully automated PT tool becomes a reality. As this report delves into the Literature Review, it will examine existing solutions to these discussed challenges. Then, by exploring current research in the field, this report can identify existing solutions and use their insights to pave the way for innovative approaches to overcome current limitations and advance the field of AI in PT.

III. Related Work

Review 1: Introduction to PT Methodology

a) Introduction

The “Penetration Testing: Practical Introduction & Tutorials" blog published by Splunk, a leading authority in cybersecurity, serves as a critical resource for understanding the intricate methodologies of

digital defense. Splunk provides valuable insights and resources across various domains, including cybersecurity, compliance, data management, IT monitoring, and overall management of IT and business operations [4]. This report introduces PT as an offensive strategy to identify system vulnerabilities by simulating cyberattacks. [5] emphasizes that the primary goal of this process is to uncover weak points, understand potential attack scenarios, and analyze the severity of vulnerabilities. By emulating the actions of a genuine attacker and reporting discovered vulnerabilities, this approach allows target systems to prioritize critical security enhancements before real threats materialize.

In addition to introducing PT as an offensive strategy, the blog delves into the intricacies of the PT process. It not only discusses how PT enhances security, but it also details the specific and complex steps and procedures involved in the PT process. These steps systematically assess and analyze system security through information gathering, scanning, exploitation, connection maintenance, eliminating intrusion evidence, and reporting. It then transitions from theoretical knowledge to practical application by providing tangible examples of PT tasks and outlining the basic functionality of various essential tools. Overall, this article delivers a comprehensive perspective on the importance of PT in cybersecurity and its practical relevance to safeguarding businesses when digital threats are both inevitable and evolving.

b) Summary

This article provides a thorough exploration of the structured stages integral to PT, to facilitate a more methodical, ethical, and comprehensive assessment of a system's security. These stages contribute to the overall effectiveness of PT by simulating real-world attack scenarios, identifying vulnerabilities, assessing their severity, and guiding the enhancement of cybersecurity measures. While other sources may present slightly different PT stage breakdowns, this blog distinguishes itself by not only submitting a comprehensive analysis of these stages but also by providing concrete examples, precise definitions, and practical insights into their importance. The stages outlined in this Splunk article include ‘Information Gathering,' ‘Scanning,' ‘Exploiting,' ‘Maintaining Access,' ‘Covering Tracks,' and ‘Reporting’ [5]. This

structured approach enables security professionals to systematically assess vulnerabilities and prioritize security improvements.

Due to the growing complexity of modern networks, Watts highlights skills critically necessary for pentesters and underscores the importance of continuous training and practice to excel in the field of PT. These skills encompass a wide range of technical and practical competencies, ranging from "knowledge of operating systems and networking," an understanding of "authentication and authorization mechanisms," to a strong foundation in programming [5]. Whether it is achieved through self-directed efforts, such as reading and exploring online resources, or through formal education, continuous skill development is crucial. Hands-on practice is emphasized as one of the most essential components, as it allows individuals to apply their knowledge and techniques in a controlled environment and ensures proficiency maintenance and adaptability.

The author reinforces this principle through a series of practical demonstrations. The first demonstration establishes a controlled test environment through VMWare, including an attacker and victim machine. This approach showcases the use of different PT tools in various phases, starting with gathering important network-related information: First by gathering domain-specific intelligence with Whois, then identifying vulnerable devices exposed on the internet through Shodan, and finally conducting a comprehensive network topology scan with Nmap. The examples in this foundational phase can be used to assist security professionals, obtain a detailed understanding of network infrastructure, and identify potential security vulnerabilities.

Once a thorough network visualization has been established, the author demonstrates a series of exploitation tools, such as password cracking with John the Ripper and web traffic interception and manipulation with Burp Suite. [5] then proceeds to highlight the use of Metasploit exploits to establish a “backdoor shell that [will enable him] to run commands on the victim system.”

c) Methodologies

This article thoroughly explores the fundamental techniques and processes behind PT and provides a structured approach for identifying and resolving security vulnerabilities. These methodologies are categorized into several stages, each playing a crucial role in the PT process. From information gathering and scanning to exploiting vulnerabilities and maintaining access, each step simulates real-world attack scenarios and helps organizations comprehend their system's weak points. By conducting PT, businesses can effectively prioritize security measures, subsequently enhancing the overall security of their digital assets.

The first stage, as defined by this article, is 'Information Gathering.' This initial phase is equivalent to conducting reconnaissance, during which a pentester collects essential data about the target system or organization. The primary goal is to gather relevant data to understand the available testing surface and potentially detect security vulnerabilities. Some common categories of information pentesters aim to collect in this stage include IP addresses, server details, subdomain identification, and specific software applications, platforms, operating systems (OS) or frameworks [5]. This phase is about building a comprehensive profile of the target to serve as the foundation for subsequent stages of the testing process.

After identifying potential entry points and vulnerabilities in the target system during the previous phase, the pentester begins to assess these points for possible weaknesses during the 'Scanning' phase. Exploration of the target system must be conducted systematically rather than haphazardly testing each potential vulnerability. Not only is a blind approach more time-consuming, but it is also significantly less effective. Therefore, to increase the chances of successful vulnerability detection, pentesters identify known vulnerabilities in their target’s framework and assess how the system responds to intrusion attempts [5]. This stage significantly improves the efficiency of the testing process as it refines the list of potential vulnerabilities and allows testers to concentrate on only the most relevant.

After vulnerabilities have been identified during the previous stages, pentesters actively exploit them during the 'Exploitation' phase. The goal is to simulate an actual intrusion by accessing data within the

target system, intentionally triggering failures, or making unauthorized changes [5]. While this critical phase mimics the actions of a genuine attacker, the author emphasizes the importance of maintaining ethical and controlled PT practices by focusing on understanding vulnerabilities rather than causing actual damage.

After successful exploitation, the testing focus shifts from gaining initial access to the 'Post-Exploitation' phases: 'Maintaining Access,' 'Covering Tracks,' and 'Reporting.' These phases, encompassing Steps 4 to 6, align with this primary purpose of assessing and improving security measures rather than engaging in malicious actions. Unlike authentic attacks, these phases aim to evaluate and enhance security by examining the system's ability not only to detect unauthorized access but also to log and store data related to security incidents [5]. Then, by reporting their insights in detail, pentesters enable organizations to fortify their defenses against real-world cyber threats.

d) Main Findings

The main findings in the article revolve around the importance of PT in the context of cybersecurity. The report highlights that businesses undergoing significant growth are more likely to attract the attention of cybercriminals and emphasizes PT as an essential strategy for protecting their digital assets. The testing process is introduced as a proactive and offensive method for identifying system vulnerabilities by simulating cyberattacks to uncover weak points, anticipate potential attack scenarios, and assess the severity of vulnerabilities.

The article also emphasizes Splunk’s multi-phase testing process, with each stage crucial in systematically replicating real-world attack scenarios. 'Information Gathering' provides critical insights into the target, while 'Scanning' refines the focus by identifying specific vulnerabilities, thus preventing inefficient testing of unrelated weaknesses. The 'Exploitation' phase, while simulating an actual attack, strictly adheres to ethical principles to avoid harming the target system. After successful exploitation, the post-exploitation phases, ‘Maintaining Access’ and ‘Covering Tracks,' evaluate an attacker's ability to

sustain a persistent presence and evade detection. The final ‘Reporting’ phase is crucial for outlining and prioritizing vulnerabilities to guide businesses to address easily exploitable weaknesses first.

Overall, this blog highlights that PT is a proactive, systematic, and highly effective approach for identifying and addressing security vulnerabilities within an organization's system. Simulating the actions of potential attackers enables organizations to fortify their security measures and safeguard their valuable assets from complex cyber threats. The article not only emphasizes the significance of PT but also provides a foundational framework for conducting the testing process through well-defined phases, all while upholding essential ethical considerations. This structured and ethical approach ensures that PT not only identifies and exploits vulnerabilities but utilizes this data to equip organizations with practical strategies to improve their overall security.

e) Implications

This source emphasizes the critical role of structured PT methodologies in reinforcing cybersecurity defenses. [5] dissection of the PT process into distinct phases, from Information Gathering to Reporting, offers a comprehensive framework that is essential for understanding the intricacies and significance of each stage. While the specific phases of PT may vary in terminology across different sources, this article outlines an offensive strategy to preemptively identify and address system vulnerabilities before they can be exploited by malicious actors.

Additionally, [5] bridges the gap between theoretical knowledge and real-world application through various practical demonstrations and examples. While highlighting the practical utility of PT tools such as Nmap, Metasploit, and Burp Suite,the article also points out the inherent limitations of their manual configurations. This observation is crucial as it introduces the potential for integrating Artificial Intelligence (AI) and Machine Learning (ML) to enhance these tools and facilitate more dynamic and adaptive security strategies.

Therefore, while the article does not explicitly discuss AI-driven techniques, itlays a solid foundation for exploring how AI and ML can be integrated into PT. AI technologies can augment traditional PT methods by automating vulnerability detection, analyzing large datasets to identify threat patterns, and optimizing exploitation techniques. This integration not only enhances the efficacy and efficiency of cybersecurity operations, but also enables innovative solutions to emerging cyber threats.

Review 2: Gathering Information with AI and Reinforcement Learning

a) 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].”

b) 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).

c) 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 [6]. 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 [6].

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 [6]. 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.

d) 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.

e) Implications

This research paper offers a comprehensive overview of PT, including its purpose, advantages, disadvantages, and unique challenges present in 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 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.

Review 3: Enhancing the Scanning Phase with GyoiThon

a) 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 [7]. 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.

b) Summary

In this study, the researchers set out to address the fundamental research question: 'How effective is the GyoiThon tool in detecting vulnerabilities [7]?' 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 [7]. 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.

c) 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 [7].

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 [8]. In contrast, the ML Mode incorporated all the steps from the Default Mode, but additionally utilized the Naïve Bayes algorithm for product/version identification [8]. This setup enabled researchers to directly evaluate the effectiveness of the Naïve Bayes algorithm in the realm of PT, aligning with their hypothesis.

d) 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 [7]. 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 [7]. 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.

e) Implications

In [7], researchers extensively explore the application of ML in PT, using GyoiThon as a focal point. By showing a practical example of how AI techniques can be effectively harnessed for the second phase of the PT process, scanning, their study directly aligns with the central theme of this project 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.

Review 4: Exploitation in PT with RL

a) 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 [3]. 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.

b) 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 [3]. 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 [3]." 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 [3].”

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.

c) 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 [9].” This approach often results in the agent executing calculated and cautious actions to minimize risks [3].

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 [3]. 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" [3]

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 [3]." 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.

d) 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 [3]. 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.

e) Implications

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 this reports 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 Phase 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.

Review 5: Deep RL in PT

a) Introduction

The article "Autonomous Security Analysis and Penetration Testing" from Arizona State University introduces an innovative framework designed to address the growing challenge of evaluating network security amidst the complexity of expanding networks and the shortage of cybersecurity professionals. By leveraging advanced RL techniques based on DeepQ Networks (DQN), this framework in this study integrates vulnerability information into the PT processes. It associates RL reward values with Common Vulnerability Scoring System (CVSS) scores, enabling prioritization of the most critical vulnerabilities.

The result is a highly efficient, automated PT system that can significantly reduce assessment time and improve overall efficiency.

b) Summary

Previous research on using RL for automating PT has focused predominantly on smaller networks and often failed to harness vulnerability information effectively. These traditional AI models have struggled to grasp the intricacies of real-world networks, falling short in accounting for the specific network structure, distribution of vulnerabilities, or correlation between vulnerabilities and exploitation probabilities [10]. This limitation has led to difficulty in prioritizing vulnerabilities, resulting in less accurate and efficient security assessments. To obtain essential information about a target and its associated vulnerabilities, these methods often rely on known sources, scans, or manual analysis for identification. This overall failure of traditional AI models to comprehensively understand the nuances of the dynamic and complex security landscapes of modern networks has resulted in a desperate need for a more comprehensive approach to PT.

Recognizing these limitations, the authors introduced the Autonomous Security Analysis and Penetration Testing framework (ASAP) as an innovative approach to security analysis and PT. This autonomous system not only understands the interconnectedness of vulnerabilities and their relation to a network's structure, but it also leverages an RL reward system based on vulnerability severity and exploitability. The approach adopted by the authors emphasizes domain-specific modeling by integrating the CVSS to quantify known vulnerabilities. This system tracks the severity of vulnerabilities and the complexity of exploiting them, allowing for a more comprehensive understanding of the network's security landscape. This modeling approach harnesses state-transition diagrams to visualize the most optimal PT policy for the network. These diagrams represent different network states and the associated actions, including probability values derived from the vulnerability's Access Complexity (AC). By generating autonomous attack plans and validating them against real-world networks, ASAP creates a comprehensive

map of security threats and potential attack paths. This approach ensures efficiency not only in smaller networks but also in large-scale environments, demonstrating its exceptional performance and scalability.

To enable autonomous PT, the authors adopt an RL-based AI algorithm to identify the optimal "attack path that maximizes the reward value for the pentester [10].” RL is a concept where an agent learns through the consequences of its interactions within an environment, focusing on long-term objectives; this can be compared to security professionals experimenting with attack strategies against vulnerabilities until successful exploitation. However, what sets their RL model apart from other traditional AI models in the PT domain is that the authors propose using a DQN-based RL model. Since DQ models learn directly from interactions with the environment by utilizing neural networks, it is more equipped to handle diverse network conditions, including those that may not have been encountered during training. As such, their RL approach involves dynamic interactions with the environment by considering the current user privilege level, actions linked to vulnerability exploitation, the difficulty and probability of a successful action, reward values, and the decision-making process. The outcome of this method is a carefully designed attack plan that "guides the security professional" through subsequent actions based on their user privilege and progression strategy [10].

c) Methodologies

The methodologies of ASAP involve a structured series of steps that enable efficient and effective PT. First, researchers employ popular scanners such as Nessus and OpenVAS to scan for vulnerabilities in the target network. The obtained scan information about the availability and accessibility of network services (e.g., open ports, protocols) and vulnerabilities within those services are then generated into an attack graph. This graph creates a visual representation of potential attack paths, relationships between different elements of a network, “and dependencies between the vulnerabilities [10]." Essential information from the attack graph is then converted into a structured format, known as a State Graph, and passed to the RL algorithm for further analysis.

The State Graph represents how privileges transition within the network and if a specific vulnerability leads to an exploit. When a vulnerability is discovered and linked to an exploit, certain attributes such as the CVSS and AC are extracted and saved for future reference. The reward value is determined by the vulnerability's CVSS score, where higher severity vulnerabilities, with a more critical potential impact if exploited, earn a higher reward. This information is vital for calculating exploit success probabilities as it is used to define and build the RL algorithm through parameters including the state of user privilege, actions, transition probability, reward values, and the agent's decision policy [10]. After confirming the success of their exploits through log analysis, the state graph and any relevant threat information are generated into an attack plan.

Finally, after the attack plan is generated, a Python wrapper for the Metasploit framework is used to validate its effectiveness. If vulnerabilities and weaknesses are found in the target organization's network or systems, the findings are used to recommend actions to improve the security of the organization. These actions may include applying patches or making changes to the network based on the vulnerabilities and weaknesses discovered during the test. Once these changes are implemented, the attack graph, which represents the network's vulnerabilities and potential attack paths, can be updated to reflect the new security measures. The system can then be retested to ensure that the implemented changes have effectively addressed the identified security issues and that the network's security posture has improved. This cyclical process of testing, improving, and retesting to enhance the organization's security is vital in cybersecurity, as it ensures that security measures remain robust, modern, and adaptive to evolving threats.

d) Main Findings

Overall, the main findings of the article emphasize that the ASAP framework, with its use of RL and attack graphs, offers a more efficient and effective approach to PT. It not only reduces the manual effort and time required, but it also reveals previously undiscovered attack paths that manual testing might miss, ultimately improving the overall security assessment process.

A case study involving the PT of an enterprise network with an industrial control system and IoT devices was conducted. The network consisted of 16 hosts distributed across three networks and offered a mix of Windows and Linux systems. The goal was to compromise email information by exploiting vulnerabilities on the SMTP service and infiltrating the IoT subsystem through a vulnerability in the gateway machine. The study investigated the effect of changes in the Discount Factor (DF) on determining the degree of significance assigned to future rewards in the decision-making process. Values closer to 0 prioritize immediate rewards, while values closer to 1 prioritize future rewards. The researchers also explored variations in batch size (BS) to explore the number of interactions the AI system uses to learn and improve its policy. These variations were analyzed to assess their influence on the RL agent's ability to make effective decisions.

The case study findings revealed that the DQN algorithm reached an effective solution quickly for different variations in DF, with the optimal value being around 0.8. Higher DF variations, 0.9 and 0.99, caused the agent to take more time to learn and make decisions as it required more time to explore each potential future outcome [10]. Similarly, reward value diminishes considerably the more the agent prioritizes long-term rewards, regardless of the number of interactions the system used to learn and improve. The agent's reward value was highest for the optimal DF around 0.8, especially with a BS of 16. The study showed that larger BS, such as 32 or 64, caused the AI's performance to decline [10]. However, researchers note that this observation might only hold true for a small network due to the increased complexity and scale of large networks, which can lead to different dynamics in the learning process for the AI system.

Researchers also conducted a scalability experiment on a simulated flat network comprising 300 hosts and three vulnerabilities. This experiment aimed to highlight the framework's scalability in situations where determining the balance between exploiting actions that appear promising based on its current knowledge and exploring new actions to discover potentially better strategies is challenging. Pentesters often experience this challenge in real-world situations, where they must decide how to allocate their resources, time, and efforts effectively.

In such situations, the experiment demonstrated the framework's ability to provide an attack plan within a short timeframe, about 70 seconds, when BS and DF parameters were set to their optimal values. This time frame is notably faster than research that utilized autonomous methods for PT, where a similar process took approximately 300 seconds (5 minutes) to perform on a network with only seven hosts. Even when the framework faced challenging scenarios with extreme BS and DF parameter values, it consistently generated effective attack plans within approximately 350 to 400 seconds [10]. These results indicate that the tested framework is highly efficient and capable of providing rapid and effective attack plans in various scenarios, significantly outperforming traditional PT methods.

What sets the ASAP framework apart from manual methods is its distinct strategy for PT. Unlike traditional manual testing, AI-based approaches, like ASAP, prioritize exploiting certain vulnerabilities before others, resulting in more efficient and effective PT. This data-driven approach involves adapting to the characteristics of vulnerabilities within unique network environments. Sometimes, starting with less challenging vulnerabilities can lead to a more efficient overall PT. The ASAP framework's adaptability and its consideration of vulnerability characteristics make it an asset in the field of cybersecurity and offers a significant reduction in the time and effort required over traditional manual approaches.

e) Implications

In the context of leveraging AI for penetration testing (PT), the article "Autonomous Security Analysis and Penetration Testing" [10] presents compelling implications for the field. This study highlights a significant shift towards AI-driven methodologies in PT, particularly through the use of DRL and DQN. These technologies address both the complexities of modern networks and the pressing shortage of cybersecurity professionals by automating and optimizing the PT process.

The research also highlights how traditional AI models can struggle with the intricacies of real-world networks, which leads to challenges in prioritizing vulnerabilities and conducting efficient security assessments. In contrast, the ASAP framework integrates vulnerability information seamlessly into the PT

process by correlating RL reward values with CVSS scores. This approach enables the system to prioritize critical vulnerabilities and optimize the overall PT process.

Additionally, by integrating established vulnerability scanners such as Nessus and OpenVAS with sophisticated attack and state graphs, [10] illustrates a structured and strategic approach to PT. This organization reduces the reliance on manual testing and allows for a more comprehensive exploration of potential security breaches. This is vital as it reduces the reliance on manual testing and allows for a more comprehensive exploration of potential security breaches. By leveraging RL-based AI algorithms, ASAP adapts dynamically to the network environment by considering factors such as vulnerability severity, exploitability, user privilege levels, and decision-making processes. This comprehensive approach not only reduces manual effort, but also uncovers previously undiscovered attack paths.

The article's exploration of the practical applications of AI in PT aligns seamlessly with the broader research objectives of enhancing PT efficiency through AI. It provides a robust example of how AI-driven frameworks can transform PT practices and demonstrates the potential of AI to not only support but significantly advance cybersecurity measures. Thus, [10] is vital for understanding the broader implications of AI in PT as it enriches the discourse on AI's role in cybersecurity by providing a concrete example of how AI can enhance traditional PT methodologies.

IV. Implementation and Evaluation

This section delves into the practical applications and effectiveness of three pivotal technological approaches in offensive penetration testing: Automated Decision-Making Systems (ADMS), Self-Improving Systems, and Human-in-the-Loop (HITL). Each system offers unique capabilities in enhancing AI's role in cybersecurity by addressing critical limitations such as excessive human reliance, insufficient data accuracy, and inadequate model robustness and environmental adaptability. Through advanced AI techniques such as Deep Reinforcement Learning (DRL) and Natural Language Processing (NLP), ADMS aims to automate and streamline decision-making processes. For example, Self-Improving Systems

leverage real-time data to continuously refine their operations and adapt to evolving threats autonomously. In contrast, HITL systems integrate human expertise directly into the AI learning loop, ensuring that models benefit from human insight while progressively reducing the need for human intervention. The evaluation of these systems will explore their respective impacts on PT methods' efficiency, accuracy, and adaptability by providing a comprehensive overview of their roles in advancing offensive cybersecurity.

A. Automated Decision-Making Systems

Automated Decision-Making Systems (ADMS) are foundational in the evolution of AI for PT due to their ability to reduce human reliance while simultaneously minimizing the previously discussed critical limitations such as data accuracy, model robustness, and environmental adaptability. These systems leverage techniques such as DRL, transfer learning, text-mining, NLP, and more to automate critical decision-making processes and allow faster and more accurate identification of potential vulnerabilities and weaknesses in cybersecurity defenses [11].

This automation is vital for pentesters as it significantly minimizes manual intervention, thereby reducing the potential for errors and improving the overall effectiveness of AI models in penetration testing scenarios. For example, ADMS can leverage formal decision support systems (DSS) and attack graph modeling to assist pentesters with identifying optimal attack vectors and vulnerabilities [11]. By automating the process of evaluating risk factors, integrating vulnerability intelligence extracted from public vulnerability datasets such as CVE and NVD, and determining the best course of action, ADMS streamlines and enhances the accuracy of penetration testing strategies.

For example, recent work by [12] demonstrates the effectiveness of automation in identifying vulnerabilities in Internet of Things (IoT) systems. They use a Named Entity Recognition (NER)-based solution to analyze security data sources in natural language, which can be time-consuming for security experts. The proposed system incorporates considerations for "ongoing changes in the CVE database and the current situation of the IoT system" to assist pentesters in quickly identifying potential vulnerabilities, assessing their impact, and recommending targeted security measures [11].

Overall, ADMS represents a significant advancement in PT, as it offers a viable solution to the challenges posed by fully automating PT. By integrating advanced AI technologies such as DRL, NLP, and formal decision support systems, ADMS enhances PT's efficiency, accuracy, and responsiveness. Therefore, since these systems automate the detection and analysis of vulnerabilities, ADMS ensures that PT evolves along with emerging threats instead of falling behind.

B. Self-Improving Systems

Integrating self-improving systems is another critical part of enhancing the effectiveness of AI within offensive pentesting methods by significantly reducing human input, enhancing data accuracy, increasing model robustness, and improving environmental adaptability. As highlighted by [13], there has been a recent trend toward integrating ML models into feedback loops through RL methodologies to interact with environments, take actions, receive feedback, and adapt accordingly. This approach allows AI applications to "operate in real environments," "react to sensory data," "perform continuous micro-simulations," and execute contextually appropriate actions [13]. By incorporating this action-reward framework into automated systems, researchers provide a means to quickly detect and respond to threats - all with millisecond latency and high throughput, while minimizing human oversight. Therefore, these systems reduce overall manual workload and minimize the risk of human error, thereby ensuring data accuracy while augmenting model robustness and environmental adaptability.

As mentioned, these systems leverage real-time data to continuously refine their decision-making processes to adapt to changes and learn from interactions within their environment. This capability is crucial in PT, where dynamic assessment and response are paramount. As a result, these systems use their low latency, high throughput, and robust fault tolerance capabilities to efficiently handle important model requirements such as "dynamic task creation," "heterogeneous tasks," and "arbitrary dataflow dependencies [13]." These attributes are essential for the real-time analysis of security vulnerabilities, rapid threat identification, and managing the complex computational demands inherent to pentesting scenarios.

Self-improving systems also improve the robustness of AI models by enabling them to handle diverse and dynamic computational tasks. Task graphs, for instance, are utilized within real-time ML systems to organize heterogeneous tasks, which vary in complexity, resource requirements, and processing types - all while meeting stringent performance benchmarks [13]. Essentially, this capability ensures that the AI systems can maintain high performance and reliability, even when faced with varying operational demands. Therefore, by introducing self-improvement mechanisms, AI models gain unique flexibility to dynamically adjust their computational strategies. This not only increases the model's robustness by allowing it to maintain optimal performance across a variety of conditions, but it also enhances the environmental adaptability.

C. Human-in-the-loop

In addition to Automated Decision-Making Systems and Self-Improving Systems, Human-in-the-loop (HITL) systems offer an alternative approach to overcoming the challenge of overt human reliance in offensive pentesting methods. While the former systems prioritize reducing human input through automation and self-improvement, HITL systems acknowledge the value of human expertise and decision-making capabilities. By "integrating human knowledge and experience" into AI models, HITL systems adopt a hybrid approach that not only enhances data accuracy, model robustness, and environmental adaptability, but also ensures that AI-driven PT tools are guided by human expertise rather than being weighed down by it [14].

HITL systems are particularly successful at refining the capabilities of AI models in fields requiring high accuracy and adaptability, such as PT. By incorporating human intelligence into the ML loop during the training phase, the model is able to improve understanding and performance. As highlighted by [14], human intervention during this phase is vital for the success of DRL, given its significant reliance on high-quality data and precise parameter adjustments. This is especially true when dealing with complex, high-dimensional data where finding optimal parameters can be challenging and often requires significant

expertise. However, HITL can address this challenge through an "interactive parameter adjustment mode," allowing users to actively refine parameters based on their intentions or goals.

HITL systems are a viable option for mitigating challenges caused by excessive human input because, despite initially seeming counterintuitive due to an emphasis on human involvement, these systems leverage AI models trained within them to adeptly mimic human decision-making processes. In other words, as AI models trained under these systems become more proficient at simulating human-like decision-making processes, the need for continuous human intervention gradually decreases. This is especially crucial for security systems, which rely heavily on human operation, as human errors are often amplified due to factors such as "inattentiveness" or "nonproficiency" [14]. Therefore, through the collaborative learning process with humans, HITL systems improve their ability to autonomously predict and manage operational states. For example, [14] illustrates how automated feedback loops that interpret real-time interactions provide precise monitoring and response mechanisms that an AI trained without these insights might miss. This precision is vital for PT because it allows for the immediate detection and mitigation of security vulnerabilities that could be exploited in real attacks.

HITL also significantly increases the robustness of AI models. As these systems encounter diverse scenarios, they learn, adapt, and strengthen their predictive and reactive capabilities. This adaptability is vital in offensive penetration methods, where AI systems must robustly handle unpredictable and evolving security threats. As a result, these systems develop resilience and flexibility by continuously learning and adapting from human experts. By dynamically allocating roles and responsibilities between humans and AI - effectively deciding "who should do what" based on situational demands - these systems can adapt to changing environmental conditions and requirements [14]. In the context of offensive PT, this adaptability is crucial as it enables AI systems to navigate and secure diverse network environments and configurations that are constantly shifting due to technological advancements and evolving threat landscapes.

Therefore, by integrating HITL, AI systems achieve higher data accuracy, increased robustness against diverse threats, and improved adaptability to different operational environments. These enhancements are crucial for the effectiveness of AI in offensive penetration methods, thus enabling these systems to

preemptively identify vulnerabilities, efficiently adapt to new attack vectors, and respond with precision. Ultimately, integrating automated PT with this system reduces reliance on continuous human input and provides more secure and resilient network infrastructures.

V. Comparison

In evaluating the implementation of ADMS, Self-Improving Systems, and HITL ifooffensive penetration methods, it is crucial to compare their contributions, limitations, and overall impact on enhancing AI's effectiveness in cybersecurity.

A. Reducing Human Input

ADMS are primarily designed to minimize human intervention by automating decision-making processes. This attribute is critical in fast-paced environments where quick response times are essential. For example, ADMS's ability to rapidly process and react to new vulnerabilities through automated workflows significantly speeds up the PT process. In contrast, while Self-Improving Systems also reduce human input, they focus more on learning from their environment to enhance their performance autonomously. These systems adapt over time, improving operations based on continuous feedback without human intervention.

HITL systems, however, adopt a different approach. Instead of simply removing the human element, they maintain a balanced level of human involvement during key model development phases. This integration enables the AI system to benefit from human expertise while gradually reducing its dependency until the AI's decision-making capabilities mature. Although this approach does not reduce human input as rapidly as ADMS, HITL prioritizes quality and accuracy by gradually offering a more controlled and gradual decrease, thus preventing potential errors that could arise from premature autonomy.

B. Data Accuracy and Model Robustness

In terms of enhancing data accuracy, HITL systems stand out due to their integration of human expertise into the data verification and training process. This method is invaluable in scenarios which require nuanced understanding or complex decision-making, such as in offensive PT. For example, in PT aimed at identifying vulnerabilities that could be exploited by Advanced Persistent Threats (APTs), HITL systems can leverage human insights to distinguish between normal network behaviors and more subtle, malicious activities [11]. Therefore, by incorporating human expertise, models developed through this system are adeptly trained to identify sophisticated attack vectors that traditional automated systems might miss.

Alternatively, Self-Improving Systems indirectly enhance data accuracy by continuously refining their algorithms as they encounter new data, thereby adapting and improving over time.

Meanwhile, ADMS improve data accuracy through the integration of various automated tools and databases, such as CVE and NVD, which help to identify and classify vulnerabilities more accurately. However, unlike Self-Improving Systems and HITL configurations, ADMS may not adapt as dynamically to new or unexpected types of data.

C. Environmental Adaptability

Environmental adaptability is another critical area for comparison among these advanced AI-driven systems. Self-improvement systems excel in this aspect as they are designed to adapt to changing environments through continuous learning and evolution. These systems are particularly effective in dynamic scenarios where attack patterns frequently change. For example, these systems can quickly adapt to new network configurations or security patches while still ensuring their attack strategies remain effective.

In contrast, ADMS is highly effective in predefined environments but may lack the flexibility to quickly adapt to unexpected changes without manual updates or reconfiguration. HITL systems, however, benefit from human input to guide their adaptation processes, thus making them highly versatile for handling new

and evolving threats. The human component allows these systems to integrate new insights that are not explicitly included in training data or recognized by purely automated systems.

D. Overall Impact

Each AI-driven system offers distinct benefits for offensive penetration methods. ADMS, for example, excels in rapid, large-scale vulnerability scans where quick decision-making is essential. They can automate exploiting known vulnerabilities across thousands of machines, thereby maximizing efficiency. Meanwhile, Self-Improving Systems are invaluable in long-term engagement scenarios where the threat landscape is constantly changing. These systems can autonomously update attack strategies based on real-time feedback from ongoing pentests. This adaptation allows them to stay ahead of evolving security measures. Lastly, HITL systems are particularly beneficial in complex attack simulations that require a nuanced understanding of the target environment — an area where human insight is irreplaceable.

In conclusion, while each system has its strengths, the choice between ADMS, Self-Improving Systems, and HITL systems often depends on the specific requirements of the PT environment. These may include needs for speed, accuracy, or adaptability. Often, a hybrid approach that incorporates elements from each system is the most effective strategy to comprehensively address all challenges of advanced automated PT methods.

VI. Conclusion and Discussion

As technology relentlessly advances, the cybersecurity field must evolve by embracing more sophisticated automation or risk falling behind. The integration of automation technologies and systems into PT offers significant potential to transform cybersecurity practices. These methods offer the potential for more efficient and effective PT processes by empowering organizations to identify and mitigate vulnerabilities proactively, automatically, and intelligently. However, navigating the dynamic intersection of technology and security demands confronting challenges such as ensuring data accuracy, maintaining model robustness, and facilitating environmental adaptability.

This report comprehensively explores integrating AI into PT by addressing these challenges and exploring their future potential. It delves into current methods and the potential novel systems including ADMS, Self-Improving Systems, and HITL, each of which offer distinct advantages for enhancing the efficiency and effectiveness of PT. ADMS enhances PT methods by applying deep DRL and NLP to enable quick and precise vulnerability assessments, though they may need further adaptation for unexpected

environmental changes. In contrast, Self-Improving Systems excel in these more dynamic settings by continuously refining their strategies. Meanwhile, HITL systems incorporate human expertise to handle complex or novel threats and balance autonomy and human insight. Together, these systems

embody a comprehensive approach to modernizing and advancing PT effectiveness. The comparison among these systems reveals that while each has its unique capabilities, often a hybrid approach that incorporates elements from each can most effectively meet the diverse needs of PT environments.

Ultimately, the research discussed in this report contributes to the advancement of automated PT methodologies and proposes viable solutions to critical challenges that hinder the evolution of PT practices. By pushing for more integrated and intelligent automation in PT, we can anticipate a future where AI is not only a tool wielded by criminal hackers but also a critical asset for cybersecurity professionals to stay one step ahead. In this way, the integration of ADMS, Self-Improving Systems, and HITL into current automated PT methods not only enhances offensive strategies, but also ensures the resilience and security of our digital future.

VII. 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] H. M. Z. A. Shebli and B. D. Beheshti, "A study on penetration testing process and tools," in *Long Island Systems, Applications and Technology Conference (LISAT)*, Farmingdale, 2018.

[3] 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.

[4] C. Kidd, *What Is Splunk & What Does It Do? An Introduction To Splunk,* 2022.

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

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

[7] 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.

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

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

[10] A. Chowdhary, D. Huang, J. S. Mahendran, D. Romo, Y. Deng and A. Sabur, "Autonomous Security Analysis and Penetration Testing," in *16th International Conference on Mobility, Sensing and Networking (MSN)*, Tokyo, 2020.

[11] R. Kaur, D. Gabrijelčič and T. Klobučar, "Artificial intelligence for cybersecurity: Literature review and future research directions," in *Information Fusion*, 2023.

[12] T.-M. Georgescu, B. Iancu and M. Zurini, "Named-Entity-Recognition-Based Automated System for Diagnosing Cybersecurity Situations in IoT Networks," *Sensors,* vol. 19, no. 15, 2019.

[13] R. Nishihara, P. Moritz, S. Wang, A. Tumanov, W. Paul, J. Schleier-Smith, R. Liaw, M. I. Jordan and I. Stoica, "Real-Time Machine Learning: The Missing Pieces," March 2017.

[14] X. Wu, L. Xiao, Y. Sun, J. Zhang, T. Maa and L. He, "A survey of human-in-the-loop for machine learning," *Future Generation Computer Systems,* vol. 135, pp. 364-381, 2022.