# Introduction

The imperative for automation in penetration testing has never been more evident. As networks grow in complexity, the demand for efficiency and accuracy intensifies. Traditional manual penetration testing methods remain laborious, resource-intensive, and struggle to keep pace with the dynamic nature of modern cyber threats.

* Background of the Course Topic
* Importance and Relevance of the Course Topic

Steps of PT

PT is a critical aspect of cybersecurity, involving the systematic assessment of network security, identification of vulnerabilities, and evaluation of potential risks.

# Literature Review

## Review 1

### Introduction

During the upcoming European Software Engineering Conference proceedings, researchers Andreas Happe and Jürgen Cito [1] will present a compelling exploration of the integration of Large Language Models (LLMs) into the realm of Penetration Testing (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 acting as virtual sparring partners. This integration could provide guidance to not only “empower existing human security testers,” but could also “counteract the lack of sufficiently educated security professionals,” addressing a current critical shortage of skilled experts in the field. [1]

### 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.

To comprehensively explore the hypothesis, the authors conducted a series of experiments, setting the stage for both high-level and low-level guidance demonstrations. In the realm of high-level guidance, they engaged LLMs to assist in the planning phase, specifically in designing penetration tests, determining tactics andtechniques, and identify potential vulnerabilities. Alternatively, they explored low-level guidance which engaged LLMs to assist in the execution phase of PT, offering more detailed and specific actions. By this point, it is assumed that penetration testers (pentesters) have completed their high-level analysis and procured their TTPs. As such, the low-level guidance is often in a step-by-step format and include activities like identifying and targeting system- 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, the authors aimed to demonstrate the extent, practicality, and effectiveness of deploying LLMs as virtual sparring partners. In order to for the LLMs to meet the authors expectation of a successful hypotheses, not only must their models must produce valid “tactics and corresponding techniques,” but these must be considered “suitable.” [1] In order to provide concrete evidence of this, the authors built upon the framework established in their research question with carefully designed expierments to encompass both levels of guidance. Their approach ranged from broad and theoretical to highly specific and practical, which allowed them to assess the capacity, effectiveness, and applicability of these models.

While the traditional approach to leveraging LLMs in penetration testing 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, the incorporation of "external knowledge and automated feedback” can mitigate the occurrence of fact inventing, known as hallucinations. Both tools can operate independently, eliminating the need for constant intervention by automatically breaking down predefined tasks into smaller, specialized subtasks through the use of "self-prompts." [1] [2] [3] AutoGPT has more decision making capabilities than AgentGPT, but AgentGPT offers a more user-friendly experience, that welcoming a wider range of users, including those without a programming background. [4] Since both AutoGPT and AgentGPT can successfully accomplish an assigned objective from a single directive, they are valid options for PT.

In the High-Level experiments, the authors focused on the LLMs' potential in providing strategic guidance for both a general and specific target during penetration testing using AutoGPT. In a general scenario, they instructed AgentGPT with the task of "becoming a domain admin in an Active Directory." [1] For the specific target, AutoGPT was tasked with creating a penetration testing plan. While both AI agents provided responses which were “realistic, and feasible, and would give a penetration tester good feedback about potential attack vectors.” [1] 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 providing step-by-step guidance, offering detailed actions such as identifying and exploiting system-specific vulnerabilities, executing custom commands and exploits, and furnishing insights on privilege escalation. At this stage, it was assumed that penetration testers 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, for the authors to consider this experiment a success, they expected the LLM to “derive feasible techniques and procedures, given an employed tactic”. [1] The goal of the experiment was to achieve privilege escalation and gain root access on a deliberately vulnerable Linux virtual machine. The authors set up a connection between GPT3.5 and the vulnerable virtual machine 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 virtual machine, analyze the command and its output, identify potential security vulnerabilities, and 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.

### 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 were 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. [1]

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 in a specific detail and losing sight of the broader picture, similar to "going down a rabbit hole." [1] 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 variations in prompts to reduce triggering ethical filters. The simplicity of engineering prompts was shown during requests of "verification commands for vulnerabilities " instead of " exploits for vulnerabilities" and instructing the AI not to “ask questions or provide judgments.” [1] 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 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 penetration testing cannot be overstated. As the field faces a critical shortage of skilled security professionals, it becomes increasingly vital to that the relationship between penetration testers 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 students a comfortable and approachable introduction of 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 its acceptability and ethical boundaries. It also addresses the accessibility of these powerful tools to both security professionals and malicious actors, prompting students to consider the distinctions between their respective ethical codes. Exploring and understanding these ethical complexities is a vital step to exploring PT.

## Review 2

### Introduction

The report by Ghanem and Chen, focuses on the initial step of PT, known as Gathering Information, and how the integration of AI, particularly Reinforcement Learning, can revolutionize this critical phase. Reinforcement Learning (RL) has quickly become one of the most important PT advancements resulting from the recent convergence of AI and cybersecurity. This transformative approach to machine learning (ML) enables systems to learn from experience through interactions with the environment. The incorporation of RL into automated PT techniques not only increases productivity, but also limit “recurrent human errors due to tiredness, omission, and pressure.” [5] However, existing automation systems have limitations in scope and optimization that result in their 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.” [5]

### Summary

Ghanem and Chen's research delves into the complexities of PT, an area that humans themselves often find challenging. They emphasize that blind automation, which entails complete automation without any human intervention, is impractical, particularly during the initial phases of PT. Since these early phases often yield incomplete results, they result in uncertainty and continuous revisitation of tasks and changes in approach. However, the authors suggest that by using RL to automate these phases intelligently, automation can more closely resemble a human expert’s decision-making process.

The challenges associated with automation in PT are not new, as autonomous systems are regularly used in the industry. However, these current systems often require substantial hands-on guidance, extensive time, and resources, limiting their applicability to smaller networks. Especially considering “PT should be repeated and performed on a regular basis to ensure continuous security,” [5] Ghanem and Chen's work suggests that automation holds the key to significantly improve various aspects of PT. Automation would not only reduce the cost of manual, repetitive, and methodical testing but also make PT more efficient and targeted, alleviating the strain on tested assets. This streamlining and automation of repetitive tasks reduces testing time, fostering adaptability and facilitates the exploration of innovative and unconventional techniques.

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. Unlike analyst-driven solutions with predetermined rules, 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 real-world PT challenges by translating them into a formal mathematical model. The POMDP model encompasses essential elements such as state observations, actions, rewards, and transition dynamics. Within this framework, an RL agent learns to make decisions based on its observations, aiming to maximize cumulative rewards. The strategies acquired by the RL agent that return the largest reward value are then stored in memory for similar cases in the future, thus enabling it to tackle complex PT problems systematically and autonomously.

One key methodology employed is the Policy Search method, which involves determining optimal sequences of actions, known as policies, that lead to the highest cumulative rewards over time within the POMDP framework. To find these optimal strategies, the researchers combined the POMDP model with specific algorithms. Notable among these is the "randomized point-based value iteration" algorithm known as PERSEUS, designed for solving POMDPs in scenarios with limited resources, making it suitable for addressing PT challenges in large networks. [5] Additionally, the PEGASUS algorithm plays a crucial role by estimating and seeking high-value sets of policies that determine the actions taken by the RL agent, simplifying the search for optimal decision-making strategies.

Initially, the learning process for their proposed system, IAPTS, relies on human input, as experts teach the system and provide knowledge. However, over time, the system evolves, with the potential to develop autonomous learning modules that reduce the need for manual interventions. 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: Simple Simulation and Experience Replay, both within controlled environments. 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, assessed execution times, and identify potential weaknesses. In 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 enhancing penetration testing 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 ability to learn from past experiences, a crucial feature proving highly beneficial in the PT field. 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 while outperforming 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 forward- thinking design ensures IAPTS remains a versatile and evolving tool in the field of penetration testing, continually enhancing its capabilities and performance.

### Relevance to Your Course Content

The research paper by Ghanem and Chen offers a comprehensive overview of PT, including its purpose, advantages, disadvantages, and the intricate challenges involved, especially as it pertains to the first step in PT - Gathering Information. They emphasize the extensive data collection and assessment required during this phase and set the stage for discussions on automation and AI as potential solutions to address the growing complexity of threats. 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 process more efficient.

The authors introduce advanced techniques such as Reinforcement Learning (RL) and Partially Observable Markov Decision Processes (POMDPs) within the context of PT. RL, as a subset of artificial intelligence (AI), is particularly relevant in the context of automating PT phases. By incorporating RL and POMDPs, the paper demonstrates the direction of future research, showing that automated PT is an evolving field with ongoing developments.

Incorporating this paper into my seminar course can provide a well-rounded understanding of PT, especially in the initial Gathering Information phase, and how advanced AI-driven techniques like RL are transforming the field. It serves as an excellent resource to introduce the challenges of PT, automation as a solution, and the application of AI in enhancing cybersecurity practices.

## Review 3

### Introduction

### Summary

### Methodologies

### Main Findings

### Relevance to Your Course Content

## Review 4

### Introduction

### Summary

### Methodologies

### Main Findings

### Relevance to Your Course Content

## Review 5

### Introduction

### Summary

### Methodologies

### Main Findings

### Relevance to Your Course Content

# Incorporation of Findings into the Course

* Discussion on how the findings from the literature review inform your course development
* Preliminary Ideas for Course Content based on the Literature Review

# Table of Contents for the Course

## Introduction

## Module 1: Introduction to AI and ML in Penetration Testing

### Module Overview and Brief Description

### 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

### Module Overview and Brief Description

### 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

### Module Overview and Brief Description

### 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

### Module Overview and Brief Description

### Submodule 4.1: Deep Learning and Advanced Techniques

### Submodule 4.2: Review/Conclusion

## Conclusion

## References and Further Reading

# Key Components of the Newly Developed Course

* Learning Objectives
* Target Audience
* Assessment Strategies
* Supplementary Materials

# Conclusion

* Summary of Key Insights
* Potential Impact of the Course

# References

* [2]
  + <https://autogpt.net/auto-gpt-vs-chatgpt-how-do-they-differ-and-everything-you-need-to-know/>
* [3]
  + <https://www.linkedin.com/pulse/autogpt-vs-agentgpt-ozgur-oscar-ozkan>
* [4]
  + <https://openaimaster.com/auto-gpt-vs-agentgpt/>
* [5]
  + Reinforcement Learning for Intelligent Penetration Testing
  + Mohamed C. , Thomas M. Chen
  + https://ieeexplore.ieee.org/document/8611595

[tflow]: <https://www.tensorflow.org/agents>