# Introduction

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

# 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][agpt] [ggpt] 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. [AvA] 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

### Summary

### Methodologies

### Main Findings

### Relevance to Your Course Content

## 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

* [agpt]
  + <https://autogpt.net/auto-gpt-vs-chatgpt-how-do-they-differ-and-everything-you-need-to-know/>
* [ggpt]
  + https://www.linkedin.com/pulse/autogpt-vs-agentgpt-ozgur-oscar-ozkan
* [AvA]
  + https://openaimaster.com/auto-gpt-vs-agentgpt/
* [tflow]
  + <https://www.tensorflow.org/agents>