Computer Network Security

COMP 178 | Spring 2025 | University of the Pacific | Jeff Shafer

Penetration Testing: Exploitation

Scanning

- Detect hosts
- Detect open ports on hosts
- Detect OS and services running on hosts
- Detect vulnerabilities on hosts

To best of scanner ability…

Penetration Testing

- Pentesting *may* stop at this point
  - Scope of test?
  - Client concerned about risks?
  - Client satisfied with a theoretical security report?
    - *Cheaper/faster*

Attackers don't stop with just a theoretical detection of weaknesses…


PROFIT

Exploitation!

- **Use a vulnerability** to achieve a **malicious end**
  - Run arbitrary commands on target?
  - Change program or OS settings on target?
  - Copy files (malicious code?) to target?
  - Copy files (corporate secrets?) from target?
  - Escalate privileges from normal user to administrator/root?

Exploitation for Pen Testers - Benefits

- **Increased accuracy of reporting for client / Reduces false positives**
  - Just because a vulnerability scanner flags a service doesn't mean that it actually *is* vulnerable. Exploitation may be impossible due to application settings, network filtering, …
  - Better sense of what attackers could accomplish
- **Pivoting**
  - Use one host (externally accessible) to gain access to other hosts on the internet network

Exploitation for Pen Testers - Drawbacks

- **Increased risk for clients**
  - Crash host or target service?
  - Data loss?
- **Increased risk for pen tester**
  - Data exposure (financial records, health records, etc) with legal disclosure requirements
  - What if you're exploiting the *wrong* system?
    - Not owned by client
    - Not authorized to test by client
- **Test Scope & Rules of Engagement**

Penetration Testing

Penetration testers restricts their activities to only systems, machines, facilities, etc. which they have **explicit written permission** to test!

Types of Exploits

Server Exploits

- Host (server) runs an application that is vulnerable
- Firewall permits access to vulnerable service
- Attacker generates specific packets to reach vulnerable service

Client Exploits

- Host (client desktop) runs an application that is vulnerable
- Client is tricked into accessing an attacker-controlled server, retrieves exploit, and runs it
  - Malware in a PDF file?
  - Malware in a Word document?
  - Malware in a media file?
  - Malware in a web page ad?
  - Malware in a Java applet?
  - Bogus link in an email?

What applications are vulnerable?

All of them? (Sooner or later…)

Will vary depending on year of test and rigor at which client applies updates

Client Exploits

- **What applications are the clients running?**
  - Won't necessarily discover this during scanning stage, which is focused more on servers
- Methods to find out
  - Ask the client (part of pen test prep)
    - Client can run automated inventory tools to produce a comprehensive report
  - Discover during reconnaissance stage (documents posted on website)
  - Educated guessing (just like real attackers!)

Client Exploits

Strengths

- Attack will work even if firewall is highly restrictive for inbound traffic
  - Outbound (Internet) traffic is usually permitted
- Attack can target a wide range of client employees (all?) to increase odds of success

Weaknesses

- Pen tester must wait for humans to take an action
- Exploit runs at privilege level of the user
  - Often requires *privilege escalation* exploit to gain sufficient control of host

Privilege Escalation Exploits

- Attacker dilemma: You have arbitrary code (or commands) running as an unprivileged user, but need to run as privileged user
- Solution? Privilege Escalation exploit to achieve
  - Linux: Root / uid 0
  - Windows: Administrator or SYSTEM

Linux Privilege Escalation Exploits

- **sudo** - Published 1/26/2021
  - Has existed in sudo (*upstream*) since July 2011 ☹
  - CVE-2021-3156 / "Baron Samedit"
  - https://blog.qualys.com/vulnerabilities-research/2021/01/26/cve-2021-3156-heap-based-buffer-overflow-in-sudo-baron-samedit
- **polkit** ("Policy Kit") – Published 1/25/2022
  - Has existed in polkit since May 2009 ☹
  - CVE-2021-4034
  - https://blog.qualys.com/vulnerabilities-threat-research/2022/01/25/pwnkit-local-privilege-escalation-vulnerability-discovered-in-polkits-pkexec-cve-2021-4034

https://blog.qualys.com/vulnerabilities-research/2021/01/26/cve-2021-3156-heap-based-buffer-overflow-in-sudo-baron-samedit

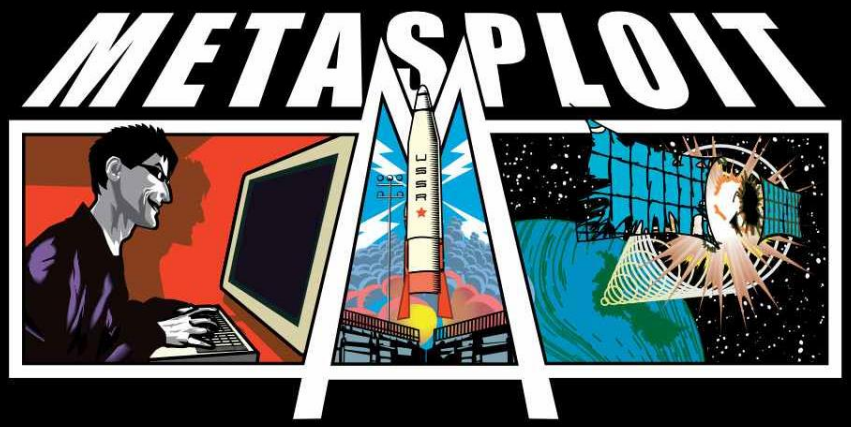
PWNKIT Vulnerability

Voice of:
Bharat Jogi
Director, Vulnerability and Threat Research

Qualys.

https://blog.qualys.com/vulnerabilities-threat-research/2022/01/25/pwnkit-local-privilege-escalation-vulnerability-discovered-in-polkits-pkexec-cve-2021-4034


Where do we get our exploits?
Do we hand craft them one at a time?



Metasploit Framework

Metasploit Framework

- Open source platform for vulnerability research, exploit development, and the creation of custom security tools
- Emphasis on **framework**
  - Ability to run myriad exploits via a **common interface**
  - Ability to create new exploits while **reusing existing foundational components** and/or **payloads**
- Cross-platform *(typically run under Linux)*

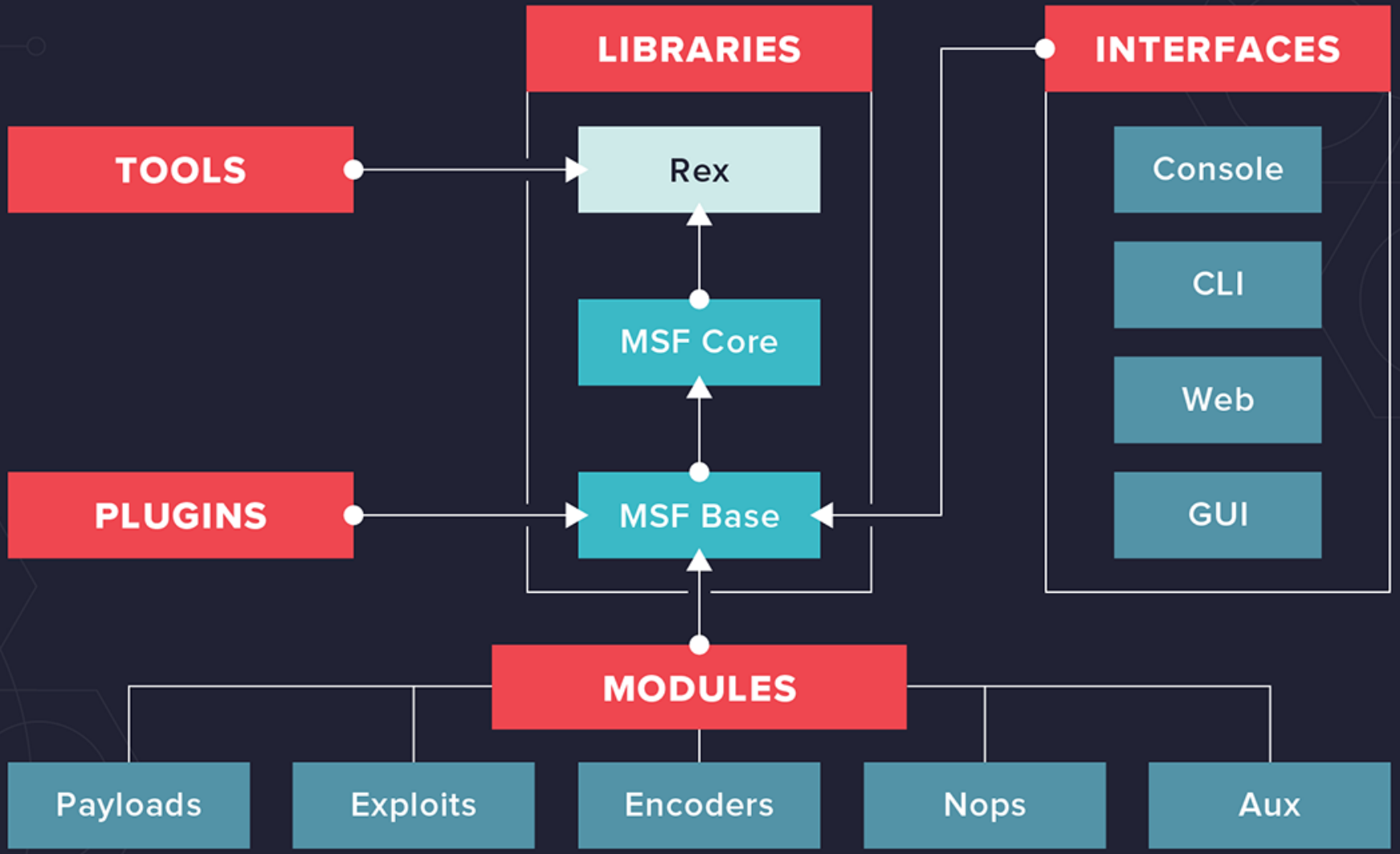
https://metasploit.com/

Metasploit Demo

Metasploit Framework
Quick Start Demo

https://asciinema.org/a/118945

https://www.kali.org/tools/metasploit-framework/


LIBRARIES
INTERFACES
TOOLS
Rex
Console
CLI
MSF Core
Web
PLUGINS
MSF Base
GUI
MODULES
Payloads
Exploits
Encoders
Nops
Aux

Metasploit: UI

- Multiple options to interact with Metasploit
  - **msfconsole** – Command-line interface
    - **Primary method used**
  - **msfrpcd** – RPC (Remote Procedure Call) interface
    - Allows 3rd party programs to interface with Metasploit core functionality
  - **msfvenom** – Package tool
    - Take a Metasploit payload and convert into a standalone executable (for Windows, Linux, etc…)
  - **Web GUI** – With the commercial ($$) Metasploit Pro
    - $15k/year *(has more features than just GUI…)*

Metasploit Modules: Exploits

- **Code that takes advantage of vulnerability**
- Targeting many operating systems (Windows, Linux, OS X, Android, etc.…) and applications that run on those systems

Metasploit Modules: Payloads

- **Code that runs after an exploit**
- Examples
  - Open remote command shell
  - Open remote GUI interface
  - Upload / download files
  - Pivot (tunnel network access through host)
- Significant code re-use opportunities here!
  - Exploit authors don’t have to re-write a payload each time

Metasploit Modules: Payloads

- Payloads can be divided internally to increase code re-use
- Singles – Stand-alone payloads that include communication and functionality
- Stagers – 1st half of payload – Loads and provides for communication
- Stages – 2nd half of payload – Loaded via stager and provides for desired functionality

Mix and match for code re-use!

Metasploit Modules: Encoders

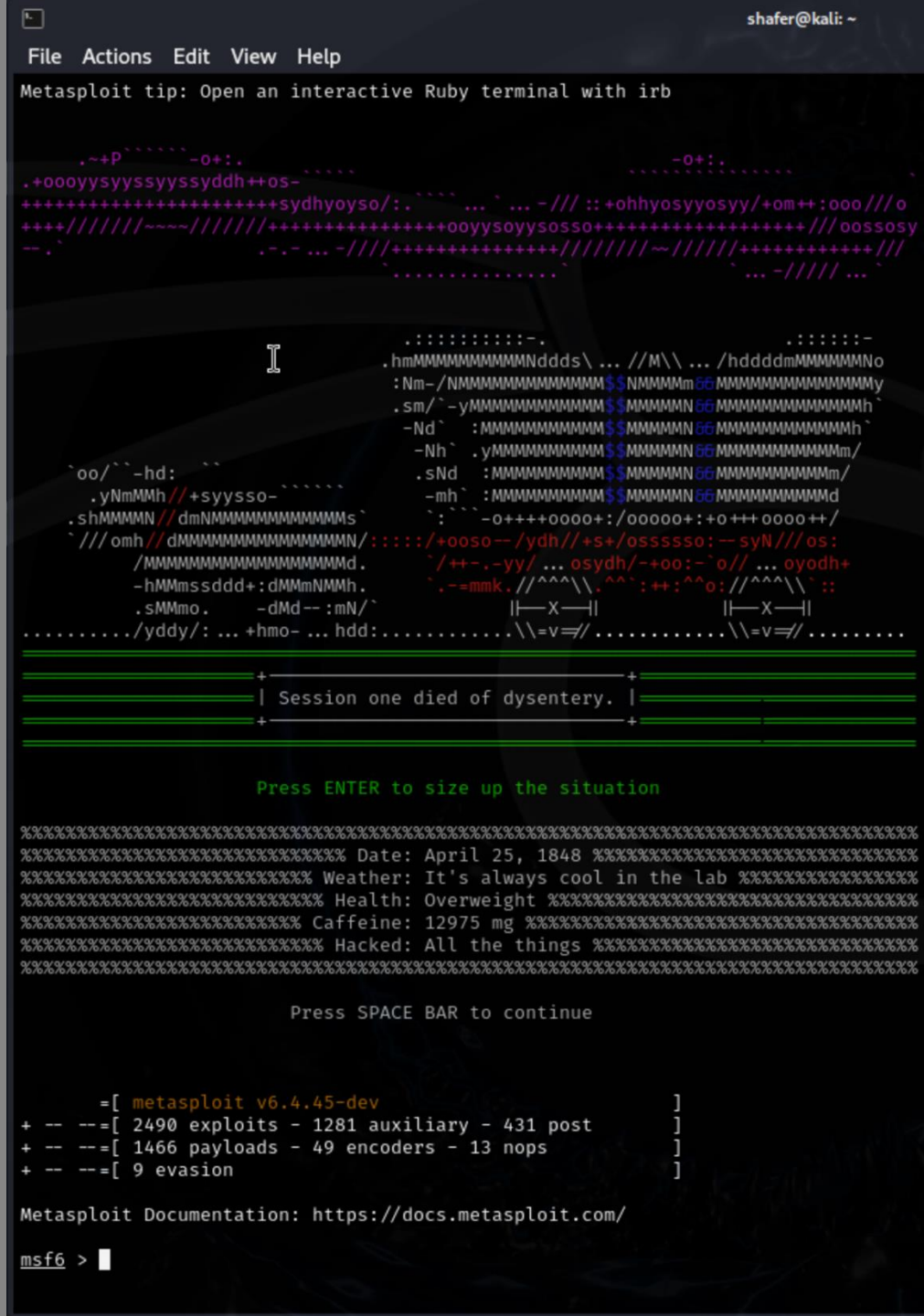
- **Code that reformats an exploit to appear benign**
- Goal is to avoid matching known signatures that intrusion detection systems look for

Metasploit Modules: NOP

- **Code that generates a NOP sled**

Metasploit Modules: Auxiliary

- Related attack and penetration testing functions
- Scanning and vulnerability testing tools
- Denial of service attacks



As-of January 2025, Metasploit Framework comes with a sizable number of exploits, payloads, encoders, …

Metasploit Resources

- **Metasploit Unleased**
  - https://www.offsec.com/metasploit-unleashed/
  - **Free Ethical Hacking Course**
- **Developer Documentation (for Exploit and Payload authors) and FAQs**
  - https://docs.metasploit.com/

Other Resources

- Lists of myriad vulnerabilities, some with with proof-of-concept code, updated frequently
- Exploit Database - https://www.exploit-db.com/
- PacketStorm - https://packetstormsecurity.com/

Lab Discussion

- **Lab 4** (Nessus Scanning)
  - Watch the disk space with: `df -h /`
    - Don't want to run out!
  - Watch in Windows Task Manager / Mac Activity Manger
    - Sufficient free RAM in system?
    - CPU utilization *not* pegged at 100% for long periods?
- ***Review*** **- Lab 3** (Scanning with Nmap)
  - Host discovery *within* local subnet
  - Host discovery *outside* local subnet
  - Capturing behavior in Wireshark