Testing for vulnerabilities

PURPOSE & MOTIVATION

- Information systems are inherently vulnerable
- Vulnerability: "System vulnerability is defined to be the intersection of a system susceptibility or flaw, access to the flaw, and the capability to exploit the flaw" [1]
- Presence of vulnerability
- Vulnerability identification
- Vulnerability Exploitation
- Exploitation of vulnerabilities by cybercriminals can cause a lot of inconvenience
- Financial losses, confidential information, human lives
- It is necessary to identify and repair vulnerabilities before they are exploited by attackers: pentesting

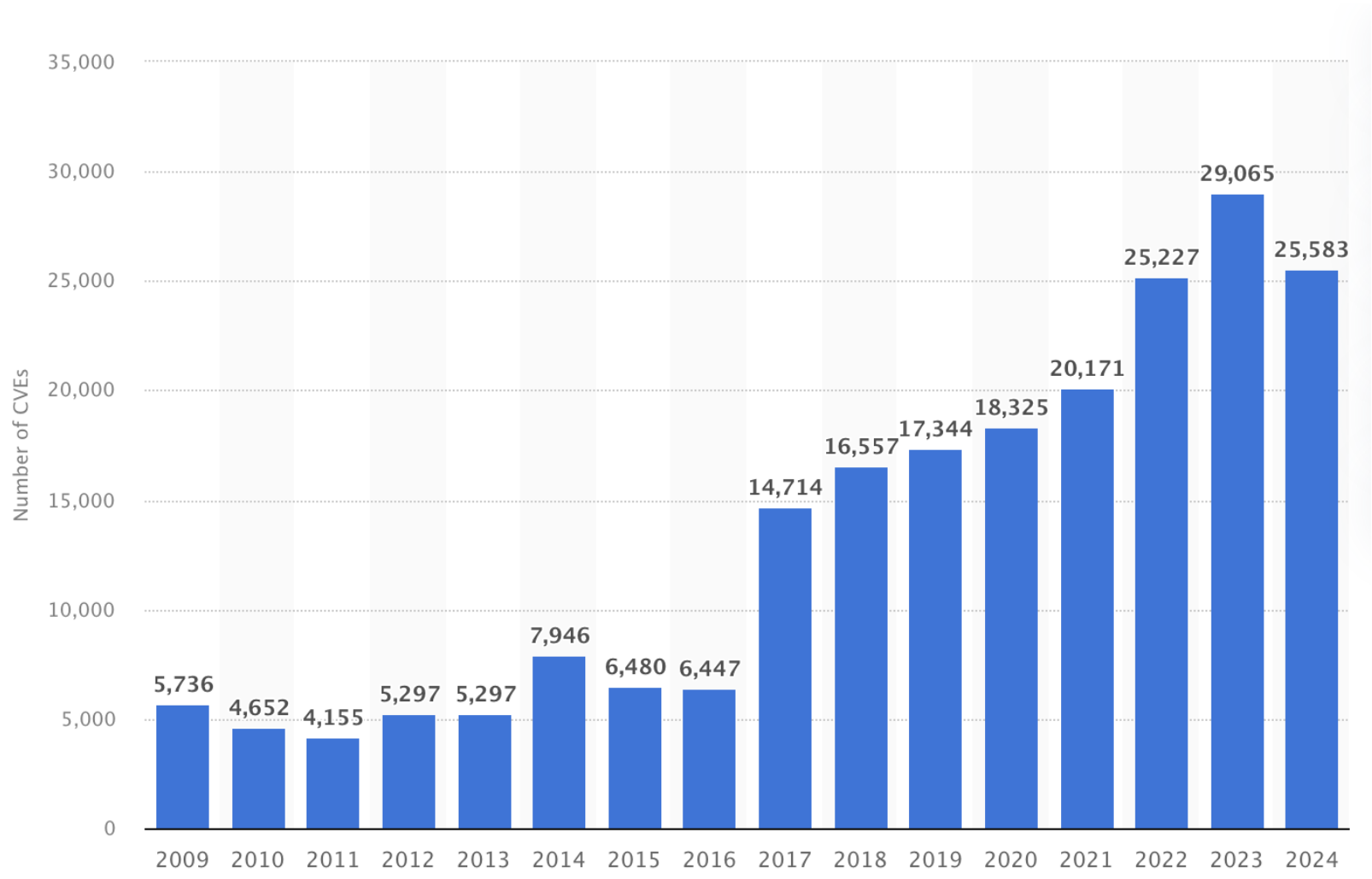
PURPOSE & MOTIVATION (2)

- Security is often ignored
- Usability = 1 / Security
- A lot of code, a lot of programmers
- Questionable quality…
- Theoretical solutions are insufficient
- Crypto, word, etc.
- Contemporary attacks are focused on financial gain or espionage

PURPOSE & MOTIVATION (3)

- New/unknown vulnerabilities
- CVE – Common Vulnerabilities and Exposures
- CVD – Coordinatied Vulnerability Disclosure
- Notify the vendor!
- Get rid of potential legal problems
- You are credited with discovering vulnerability
- Be responsible!
- Most of the time, you'll be relying on known vulnerabilities
  - https://www.exploit-db.com

PURPOSE & MOTIVATION (4)



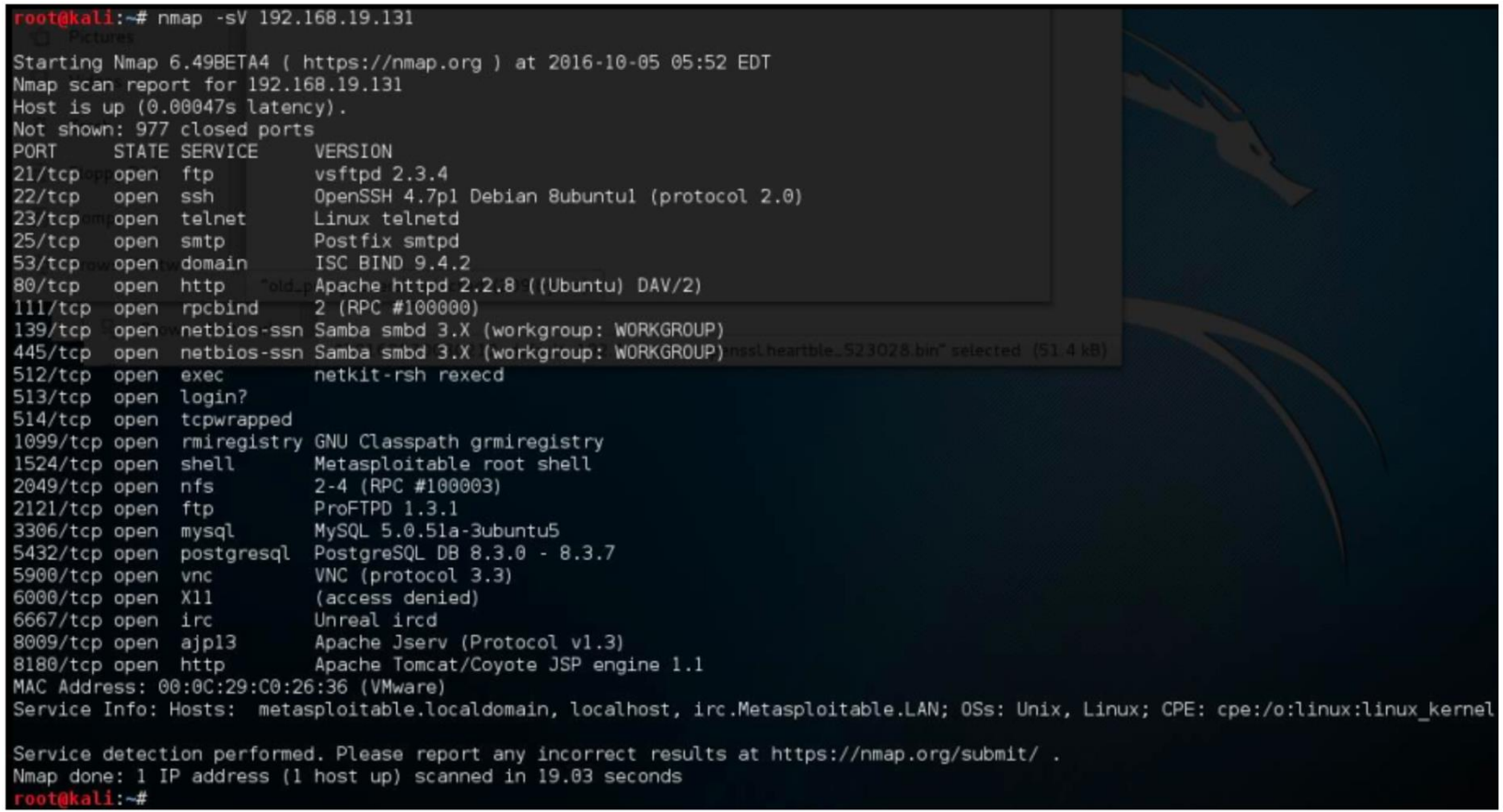
https://www.cvedetails.com/browse-by-date.php

METHODOLOGY

1. information gathering
2. service enumeration
3. exploitation
4. persistence
5. post exploitation enumeration
6. house keeping

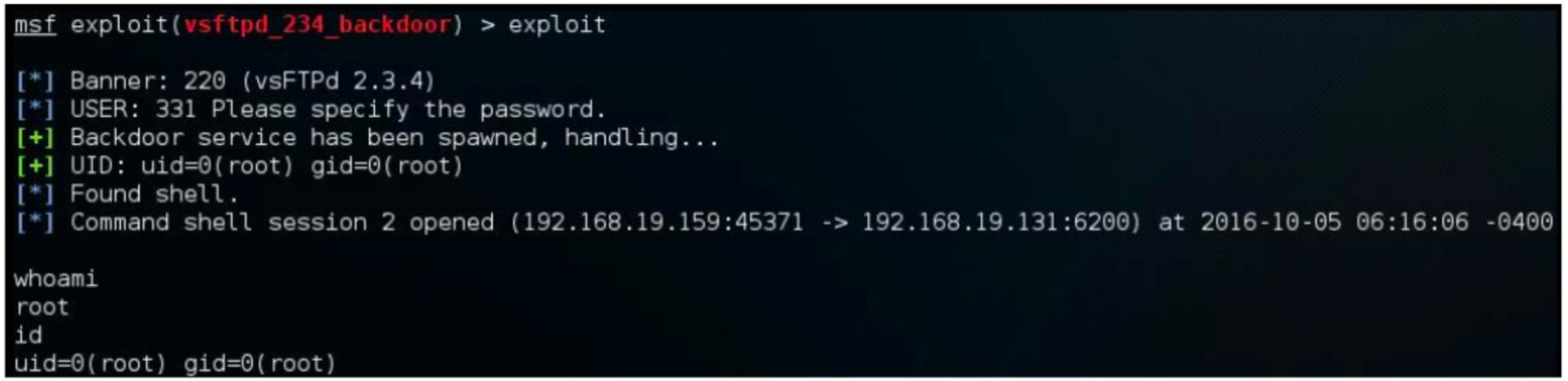
METHODOLOGY (II)

- service enumeration
  - port scanning, SNMP, DNS, SMTP, SQL, etc.



METHODOLOGY (III)

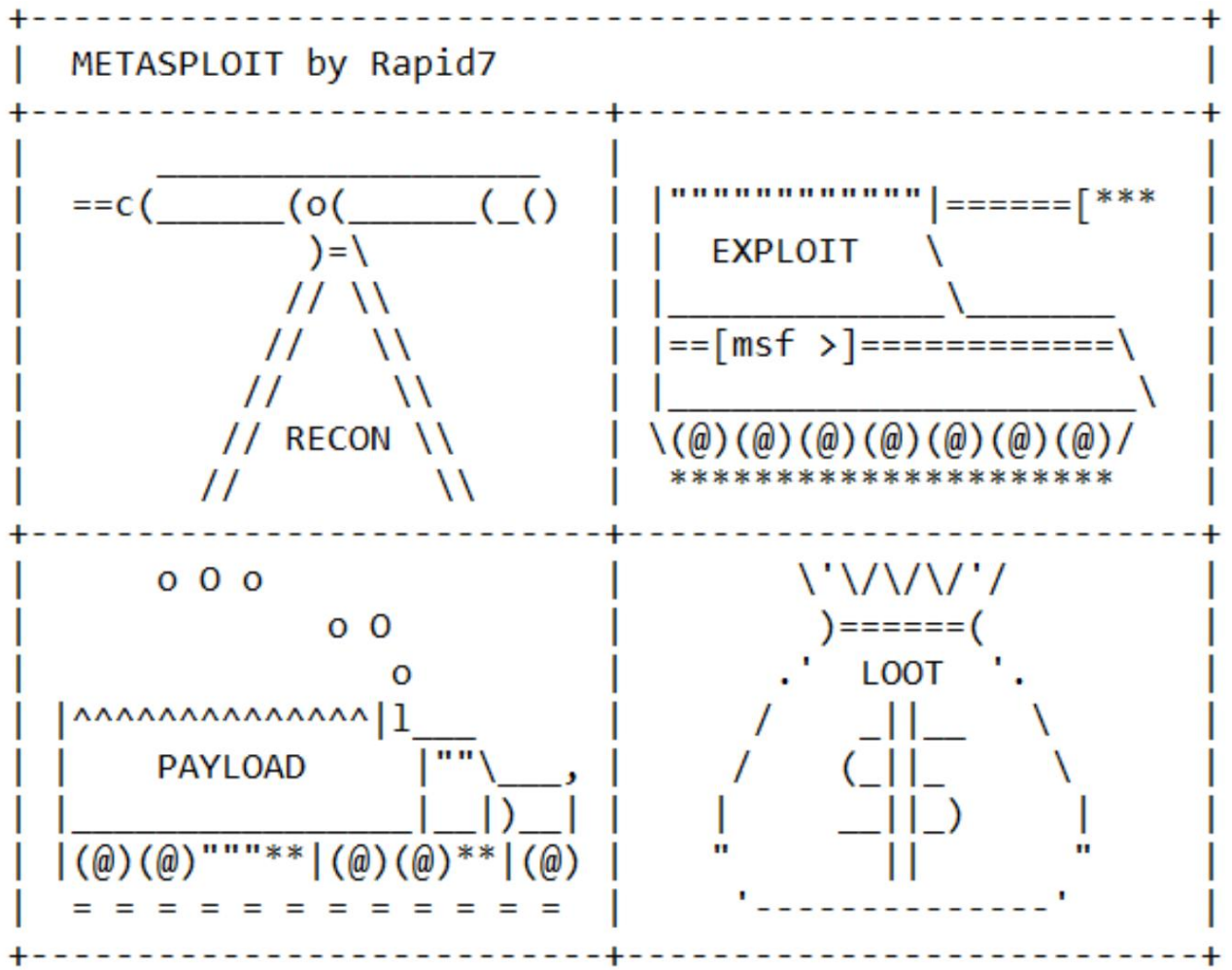
- exploitation
  - obtaining access to the attacked system
    - usually using exploits for vulnerabilities discovered in step 2



METHODOLOGY (IV)

- Persistence of access
  - Installation of backdoors
  - Adding new users
  - Obtaining hashes/passwords for later access
- Post-attack enumeration
  - Confidential information, hashes, documents, etc.
- House-keeping
  - Rootkits for hiding components "installed" on the attacked system
  - Cleaning traces
  - Cleaning access logs, history, etc.
  - Delete installed files/components

METHODOLOGY (V)



Tools

- Operating System
  - Kali Linux
    - Debian-based distribution
    - Contains a lot of specific tools
- Mass scanning/mining tools (not recommended)
  - OpenVAS, CoreImpact, SAINT, Nessus, NeXpose
- Basic tools
  - Metasploit, nmap

Tools (II)

- Obtaining information
  - Netdiscover, nmap, Maltego, etc.
- Vulnerability analysis
  - Nmap, Golismero, OpenVAS, etc.
- Web Application Analysis
  - Burpsuite, WebScarab, etc.
- Database analysis
  - SQLMap, SQLNinja, etc.
- Password attacks
  - John, Hashcat, Rainbowcrack, etc.

Tools (III)

- For wireless networks
  - aircrack-ng, etc.
- Reverse engineering
  - OllyDbg, NASM, clang, apktool, etc.
- Exploitation
  - Metasploit, SQLMap, armitage, etc.
- Sniffing & spoofing
  - Wireshark, ettercap, reply, etc.
- Post-exploitation
  - ProxyChains, bdfproxy, etc.

Report

- The pentesting procedure ends with a report
- The report must include:
  - Every vulnerable system
  - Each vulnerability identified
  - Methodology/Exploitation Steps
  - Risk analysis
  - Solutions
- The report is prepared by the pentester and handed over to the customer
- Information must be communicated clearly and effectively
  - The client may not be very educated in the field of security…