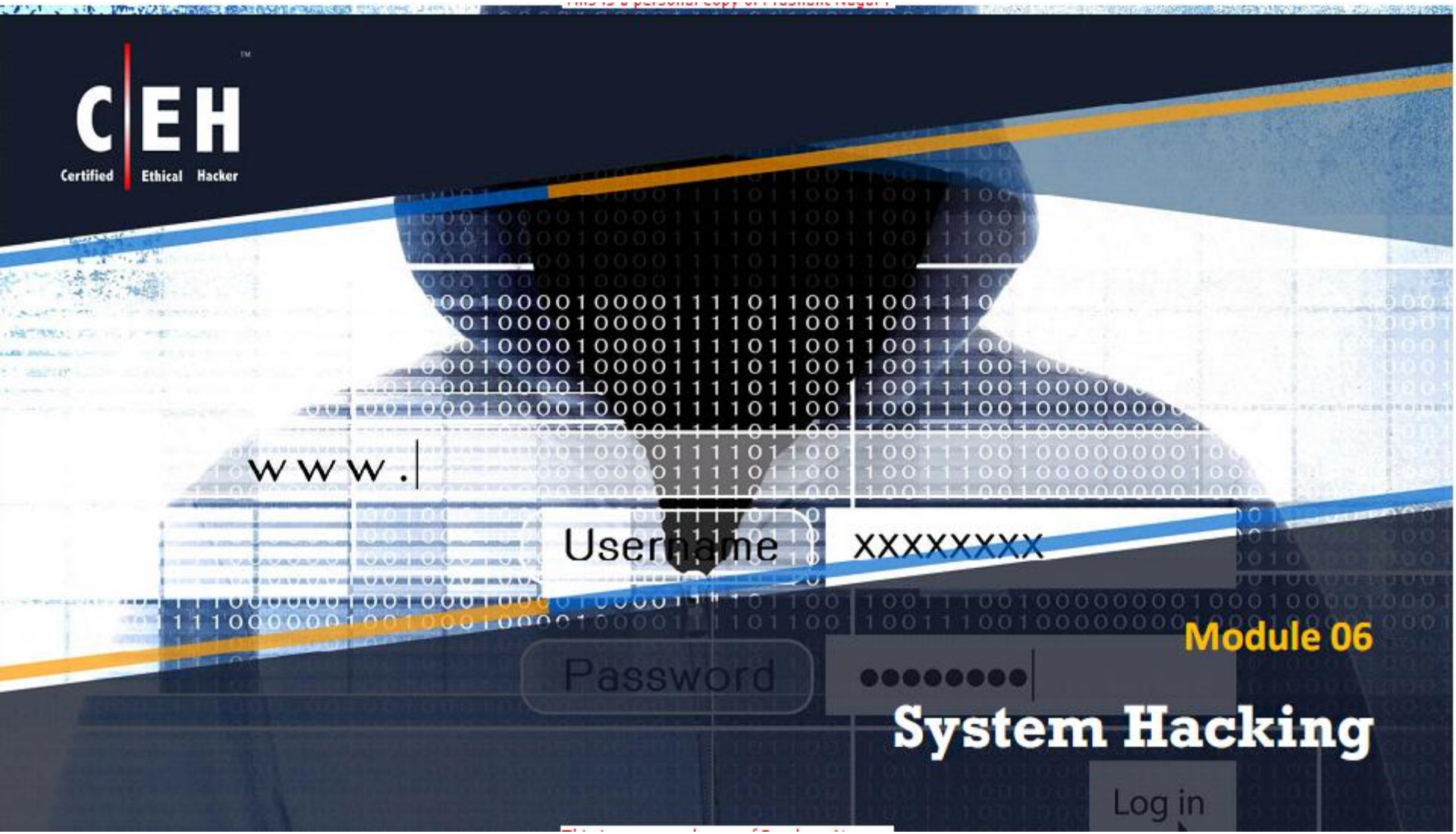



Module 06

# System Hacking

# Module Objectives

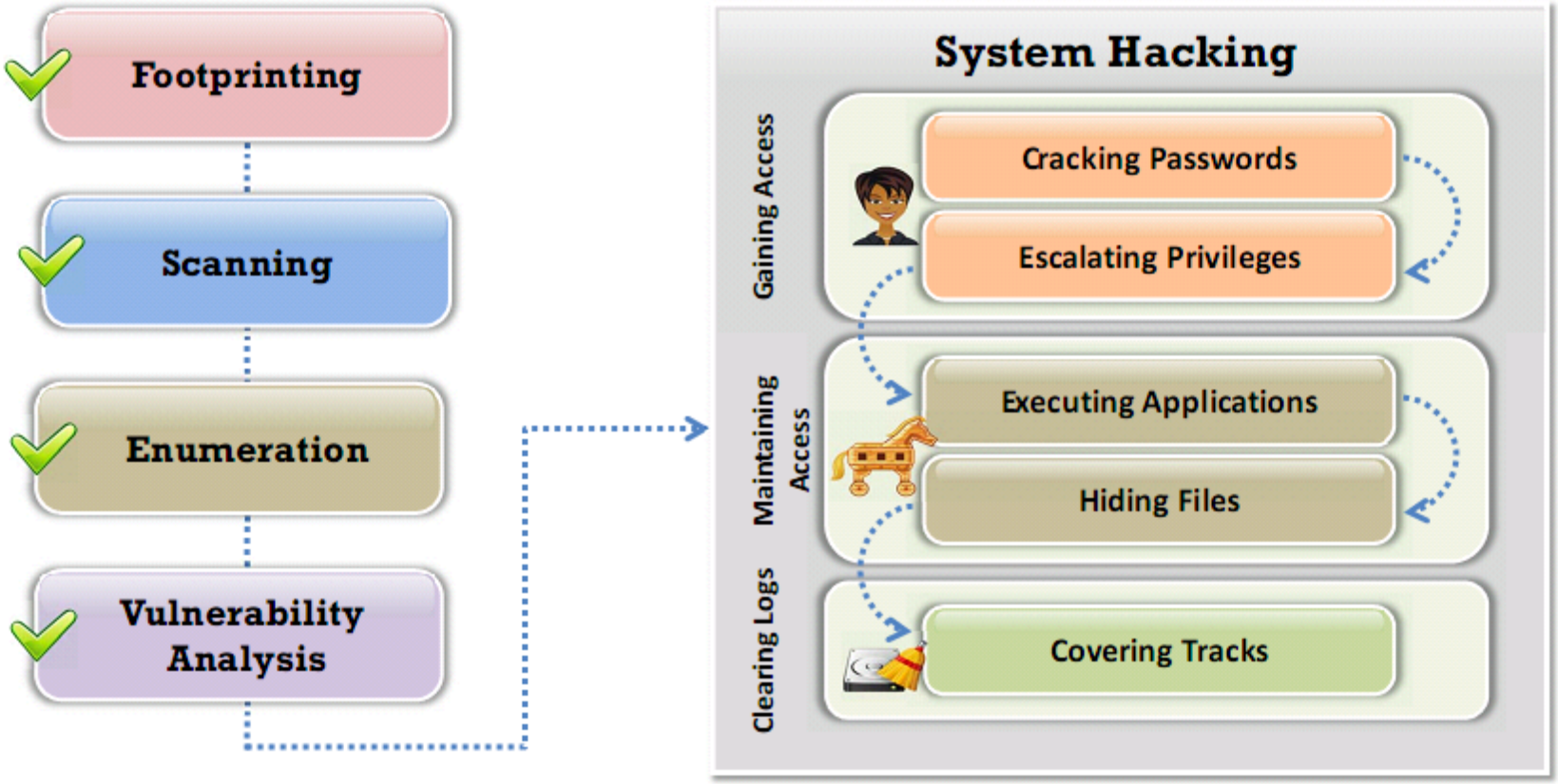
Module Objectives

- Overview of CEH Hacking Methodology
- Understanding Techniques to Gain Access to the System
- Understanding Privilege Escalation Techniques
- Understanding Techniques to Create and Maintain Remote Access to the System
- Overview of Different Types of Rootkits
- Overview of Steganography and Steganalysis Techniques
- Understanding Techniques to Hide the Evidence of Compromise
- Overview of System Hacking Penetration Testing

# Module Flow

1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# CEH Hacking Methodology (CHM)



# System Hacking Goals

| Hacking-Stage | Goal | Technique/Exploit Used |
|---|---|---|
| Gaining Access | To bypass access controls to gain access to the system | Password cracking, social engineering |
| Escalating Privileges | To acquire the rights of another user or an admin | Exploiting known system vulnerabilities |
| Executing Applications | To create and maintain remote access to the system | Trojans, spywares, backdoors, keyloggers |
| Hiding Files | To hide attackers malicious activities and data theft | Rootkits, steganography |
| Covering Tracks | To hide the evidence of compromise | Clearing logs |

# Module Flow

1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# Password Cracking



Password cracking techniques are used to **recover passwords** from computer systems

Attackers use password cracking techniques to **gain unauthorized access** to vulnerable system

Most of the password cracking techniques are successful due to weak or easily **guessable passwords**

# Types of Password Attacks



| | |
|---|---|
| **Non-Electronic Attacks** | Attacker need not posses **technical knowledge** to crack password, hence known as non-technical attack<br>• Shoulder Surfing • Social Engineering • Dumpster Diving |
| **Active Online Attacks** | Attacker performs password cracking by **directly communicating** with the victim machine<br>• Dictionary and Brute Forcing Attack • Hash Injection and Phishing • LLMNR/NBT-NS Poisoning<br>• Trojan/Spyware/Keyloggers • Password Guessing |
| **Passive Online Attacks** | Attacker performs password cracking **without communicating** with the authorizing party<br>• Wire Sniffing • Man-in-the-Middle Attack • Replay Attack |
| **Offline Attacks** | Attacker copies the target's **password file** and then tries to crack passwords in his own system at different location<br>• Rainbow Table Attack (Pre-Computed Hashes) • Distributed Network Attack |

# Non-Electronic Attacks



## Social Engineering

Convincing people to reveal passwords

## Shoulder Surfing

Looking at either the user's keyboard or screen while he/she is logging in

## Dumpster Diving

Searching for sensitive information in the user's trash-bins, printer trash bins, and user desk for sticky notes

# Active Online Attack: Dictionary, Brute Forcing, and Rule-based Attack



## Dictionary Attack

A **dictionary file** is loaded into the cracking application that runs against **user accounts**

## Brute Forcing Attack

The program tries **every combination of characters** until the password is broken

## Rule-based Attack

This attack is used when the attacker gets some **information about the password**

# Active Online Attack: Password Guessing



The attacker creates a list of all possible passwords from the information collected through **social engineering** or any other way and tries them manually on the victim's machine to **crack the passwords**

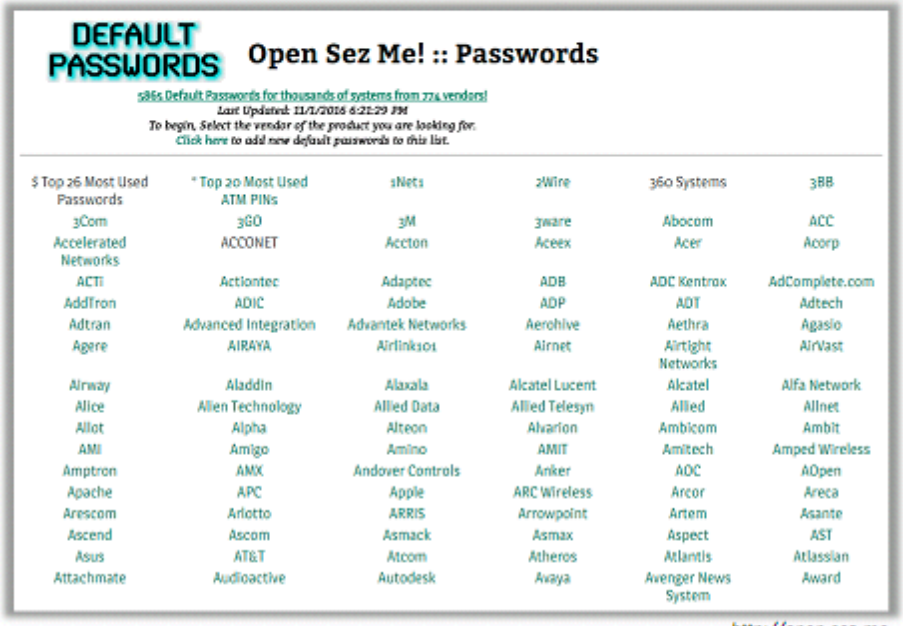
**Frequency of attacks is less**

**The failure rate is high**

1. Find a valid user
2. Create a list of possible passwords
3. Rank passwords from high probability to low
4. Key in each password, until correct password is discovered

# Default Passwords

- A default password is a **password supplied by the manufacturer** with new equipment (e.g. switches, hubs, routers) that is password protected
- Attackers use **default passwords** present in the list of words or dictionary that they use to **perform password guessing attack**



DEFAULT PASSWORDS

**Open Sez Me! :: Passwords**

5865 Default Passwords for thousands of systems from 774 vendors!
*Last Updated: 11/1/2016 6:21:29 PM*
*To begin, Select the vendor of the product you are looking for.*
*Click here to add new default passwords to this list.*

| | | | | | |
|---|---|---|---|---|---|
| $ Top 26 Most Used Passwords | * Top 20 Most Used ATM PINs | 1Nets | 2Wire | 360 Systems | 3BB |
| 3Com | 3GO | 3M | 3ware | Abocom | ACC |
| Accelerated Networks | ACCONET | Accton | Aceex | Acer | Acorp |
| ACTi | Actiontec | Adaptec | ADB | ADC Kentrox | AdComplete.com |
| AddTron | ADIC | Adobe | ADP | ADT | Adtech |
| Adtran | Advanced Integration | Advantek Networks | Aerohive | Aethra | Agasio |
| Agere | AIRAYA | Airlink101 | Airnet | Airtight Networks | AirVast |
| Airway | Aladdin | Alaxala | Alcatel Lucent | Alcatel | Alfa Network |
| Alice | Alien Technology | Allied Data | Allied Telesyn | Allied | Allnet |
| Allot | Alpha | Alteon | Alvarion | Ambicom | Ambit |
| AMI | Amigo | Amino | AMIT | Amitech | Amped Wireless |
| Amptron | AMX | Andover Controls | Anker | AOC | AOpen |
| Apache | APC | Apple | ARC Wireless | Arcor | Areca |
| Arescom | Arlotto | ARRIS | Arrowpoint | Artem | Asante |
| Ascend | Ascom | Asmack | Asmax | Aspect | AST |
| Asus | AT&T | Atcom | Atheros | Atlantis | Atlassian |
| Attachmate | Audioactive | Autodesk | Avaya | Avenger News System | Award |

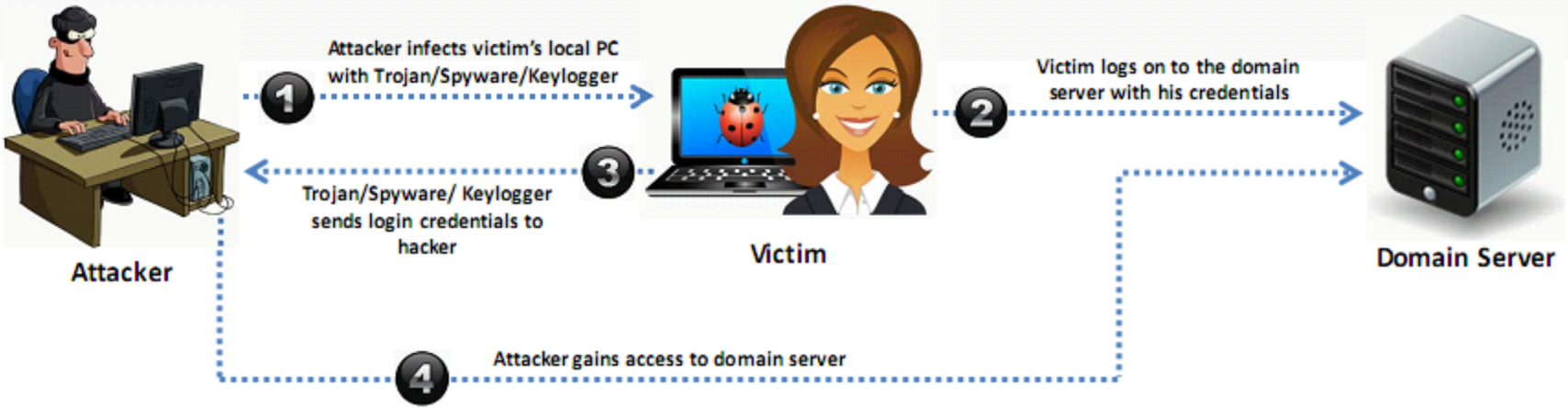
*http://open-sez.me*

## Online Tools to Search Default Passwords

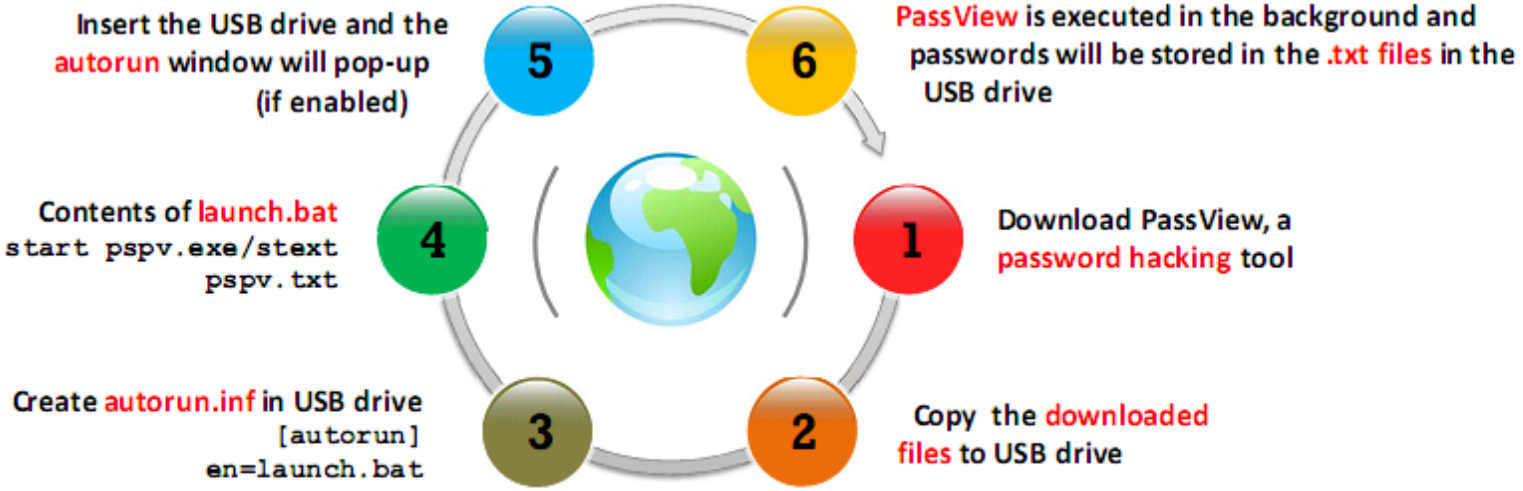
- *https://www.fortypoundhead.com*
- *https://cirt.net*
- *http://www.defaultpassword.us*
- *http://defaultpasswords.in*
- *http://www.routerpasswords.com*
- *http://www.defaultpassword.com*
- *https://default-password.info*

# Active Online Attack: Trojan/Spyware/Keylogger

- Attacker installs Trojan/Spyware/Keylogger on victim's machine to collect victim's **user names and passwords**
- Trojan/Spyware/Keylogger **runs in the background** and sends back all user credentials to the attacker



# Example of Active Online Attack Using USB Drive

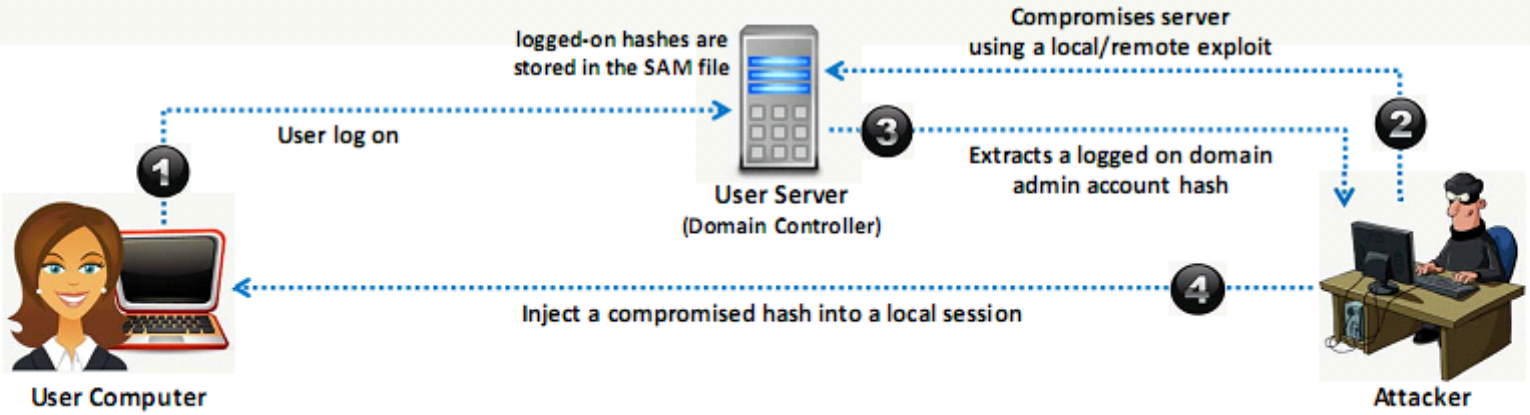
# Active Online Attack: Hash Injection Attack



A hash injection attack allows an attacker to **inject a compromised hash** into a local session and use the hash to validate network resources

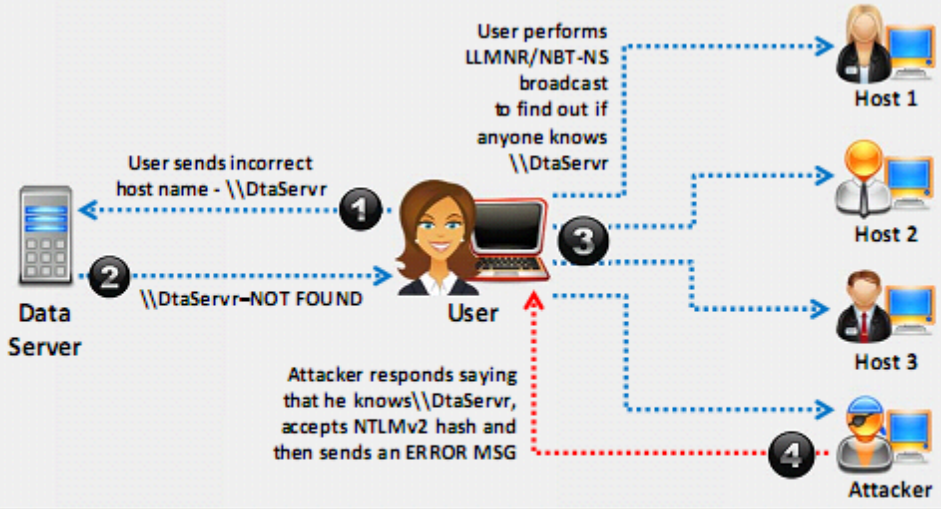
The attacker finds and extracts a logged on **domain admin account hash**

The attacker uses the extracted hash to log on to the **domain controller**

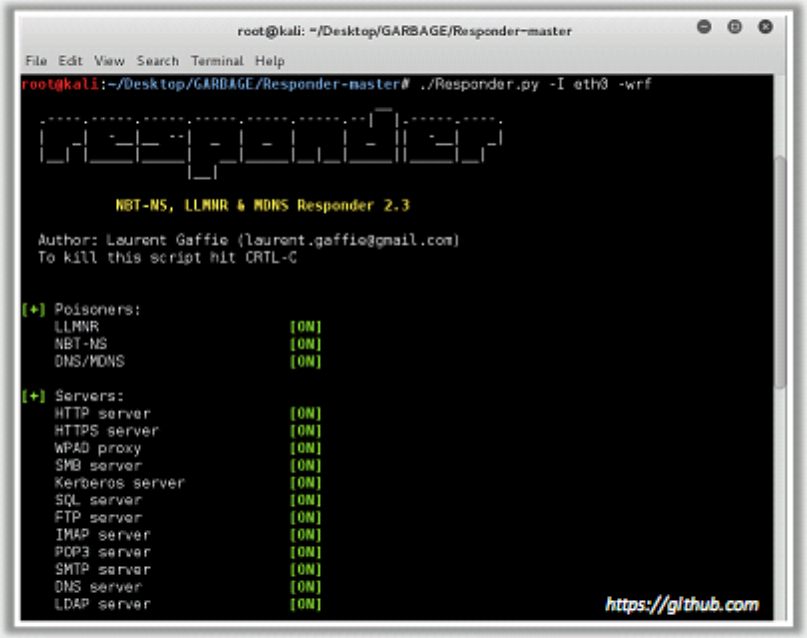


# Active Online Attack: LLMNR/NBT-NS Poisoning

- LLMNR and NBT-NS are two main elements of **Windows operating systems** used to perform **name resolution** for hosts present on the same link
- The attacker cracks the **NTLMv2 hash** obtained from the victim's authentication process
- The extracted credentials are used to log on to the **host system in the network**




## LLMNR/NBT-NS Spoofing Tool: Responder




**LLMNR/NBT-NS Spoofing Tools**

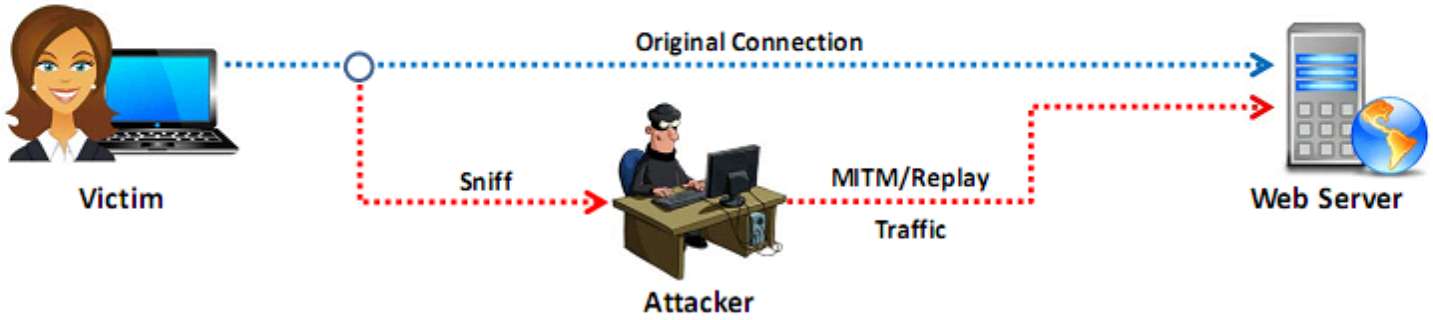
- Metasploit (*https://www.metasploit.com*)
- NBNSpoof (*https://github.com*)
- Inveigh (*https://github.com*)

# Passive Online Attack: Wire Sniffing

- Attackers run **packet sniffer tools** on the local area network (LAN) to access and record the raw network traffic
- The captured data may include **sensitive information** such as **passwords** (FTP, rlogin sessions, etc.) and emails
- Sniffed credentials are used to **gain unauthorized access** to the target system


**Wire Sniffing** ····▸ **Computationally Complex** ····▸ **Hard to Perpetrate**

# Passive Online Attacks: Man-in-the-Middle and Replay Attack




- In a MITM attack, the attacker acquires **access** to the communication channels between victim and server to extract the information
- In a replay attack, packets and authentication tokens are captured using a **sniffer**. After the relevant info is extracted, the tokens are placed back on the network to gain access

## Considerations

- Relatively **hard to perpetrate**
- Must be **trusted** by one or both sides
- Can sometimes be broken by **invalidating traffic**

# Offline Attack: Rainbow Table Attack



## Rainbow Table

A rainbow table is a precomputed table which contains word lists like dictionary files and brute force lists and their hash values

## Compare the Hashes

Capture the hash of passwords and compare it with the precomputed hash table. If a match is found then the password is cracked

## Easy to Recover

It is easy to recover passwords by comparing captured password hashes to the precomputed tables

## Precomputed Hashes

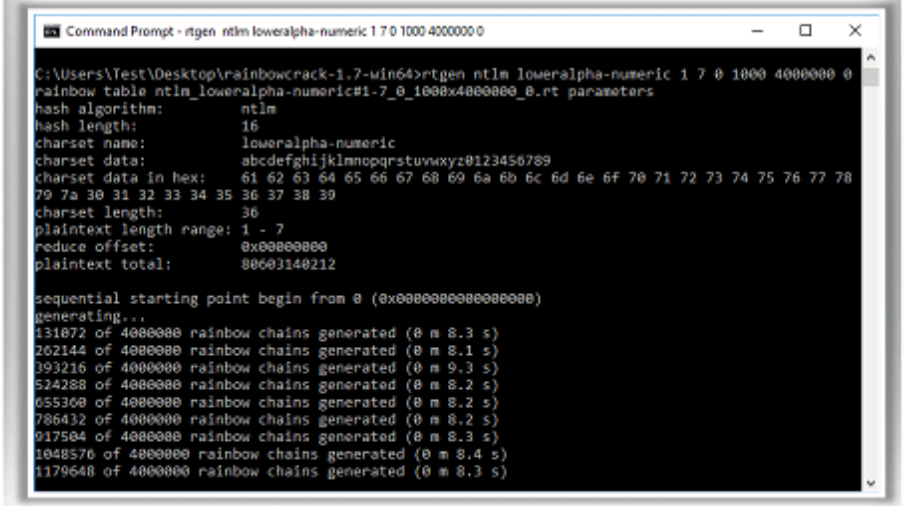
| | |
|---|---|
| 1qazwed | 4259cc34599c530b28a6a8f225d668590 |
| hh021da | c744b1716cbf8d4dd0ff4ce31a177151 |
| 9da8dasf | 3cd696a8571a843cda453a229d741843 |
| sodifo8sf | c744b1716cbf8d4dd0ff4ce31a177151 |

# Tools to Create Rainbow Tables: rtgen and Winrtgen

## rtgen

- The rtgen program needs **several parameters** to generate a rainbow table. Syntax for the command line is:

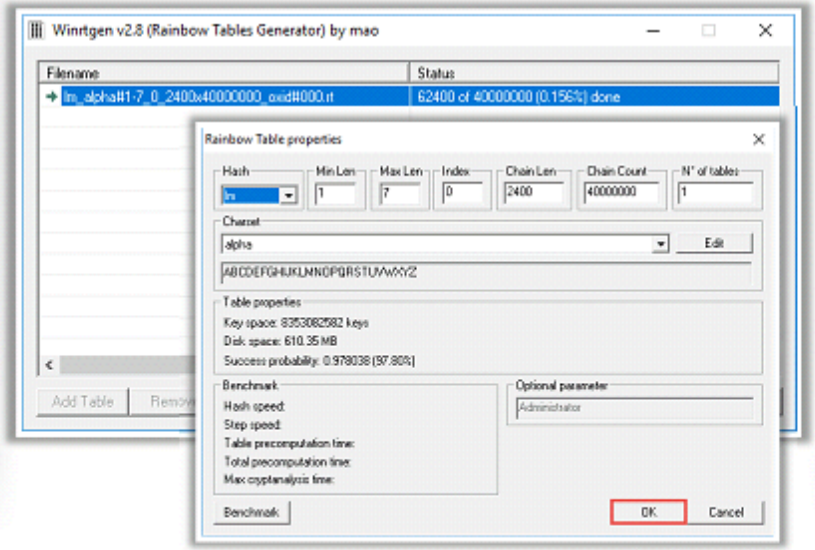
**Syntax:** `rtgen hash_algorithm charset plaintext_len_min plaintext_len_max table_index chain_len chain_num part_index`




*http://project-rainbowcrack.com*

## Winrtgen

- Winrtgen is a graphical **rainbow table generator** that supports LM, FastLM, NTLM, LMCHALL, HalfLMCHALL, NTLMCHALL, MSCACHE, MD2, MD4, MD5, SHA1, RIPEMD160, MySQL323, MySQLSHA1, CiscoPIX, ORACLE, SHA-2 (256), SHA-2 (384), and SHA-2 (512) hashes




*http://www.oxid.it*

# Offline Attack: Distributed Network Attack



- A Distributed Network Attack (DNA) technique is used for **recovering passwords from hashes or password protected files** using the unused processing power of machines across the network to decrypt passwords

The DNA Manager is installed in a **central location** where machines running on DNA Client can access it over the network

DNA Manager coordinates the attack and **allocates small portions of the key search** to machines that are distributed over the network

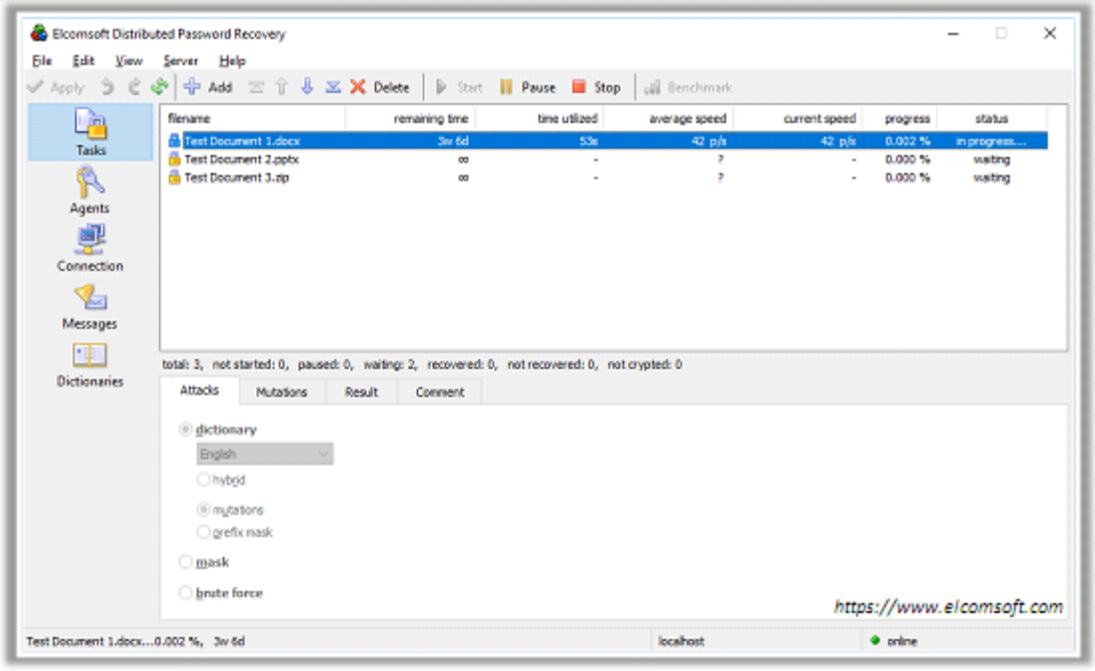
DNA Client **runs in the background** consuming only unused processor time

The program combines processing capabilities of all the clients connected to network and uses it to **crack the password**

# Password Recovery Tools

## Elcomsoft Distributed Password Recovery

- Elcomsoft Distributed Password Recovery breaks **complex passwords**, recovers strong **encryption keys**, and **unlocks documents** in a production environment




- **Passware Kit Forensic**
  *https://www.passware.com*
- **WINDOWS PASSWORD RECOVERY TOOL ULTIMATE**
  *https://www.tenorshare.com*
- **Stellar Phoenix Password Recovery**
  *https://www.stellarinfo.com*
- **Windows Password Recovery Tool**
  *https://www.windowspasswordsrecovery.com*
- **PCUnlocker**
  *https://www.top-password.com*

# Microsoft Authentication



## Security Accounts Manager (SAM) Database

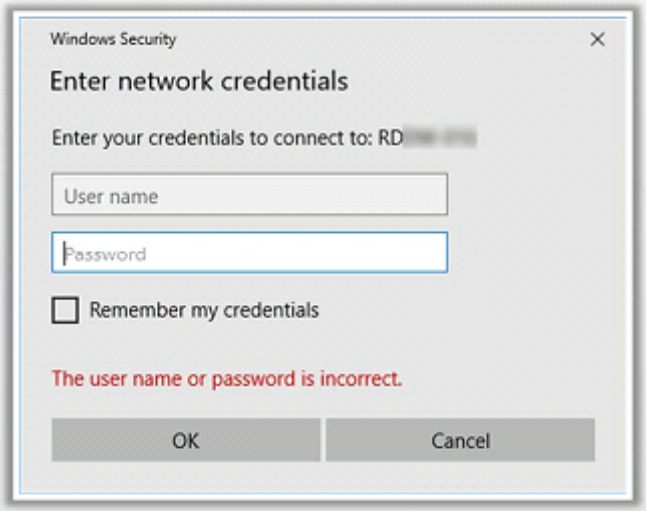
- Windows stores user passwords in SAM, or in the **Active Directory database** in domains. Passwords are never stored in clear text; passwords are hashed and the results are stored in the SAM

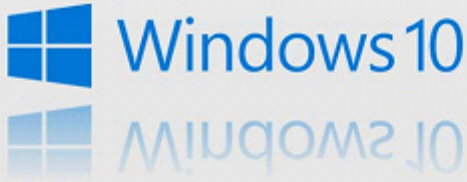
## NTLM Authentication

- The NTLM authentication protocol types are: **NTLM authentication protocol** and **LM authentication protocol**
- These protocols store user's password in the SAM database using different hashing methods

## Kerberos Authentication

- Microsoft has upgraded its **default authentication protocol** to Kerberos which provides a stronger authentication for client/server applications than NTLM



# How Hash Passwords Are Stored in Windows SAM?

Shiela/test

Password hash using LM/NTLM

```
Shiela:1005:NO PASSWORD*****************
****:0CB6948805F797BF2A82807973B89537:::
```

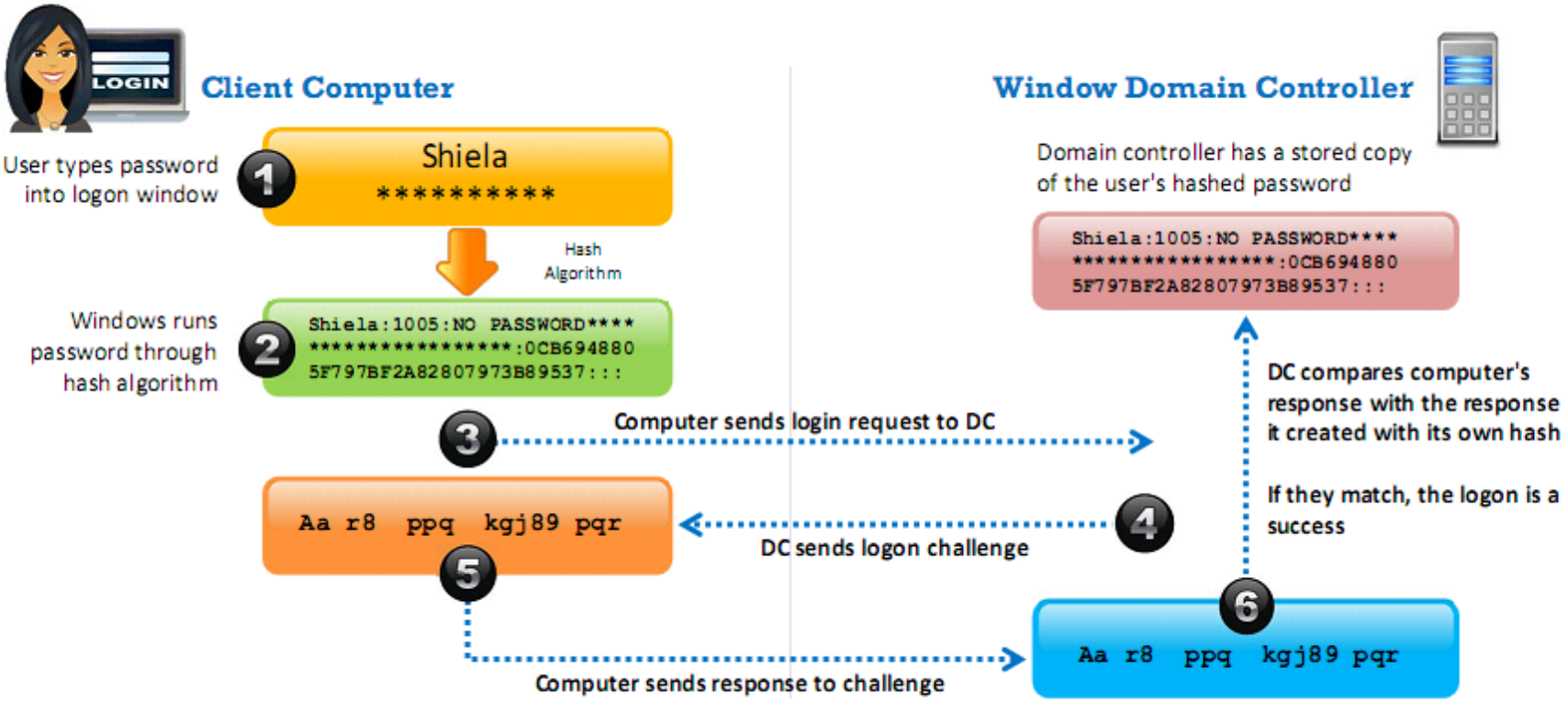
SAM File is located at `c:\windows\system32\config\SAM`

```
Administrator:500:NO PASSWORD*********************:61880B9EE373475C8148A7108ACB3031:::
Guest:501:NO PASSWORD*********************:NO PASSWORD*********************:::
Admin:1001:NO PASSWORD*********************:BE40C450AB99713DF1EDC5B40C25AD47:::
Martin:1002:NO PASSWORD*********************:BF4A502DA294ACBC175B394A080DEE79:::
Juggyboy:1003:NO PASSWORD*********************:488CDCDD2225312793ED6967B28C1025:::
Jason:1004:NO PASSWORD*********************:2D20D252A479F485CDF5E171D93985BF:::
Shiela:1005:NO PASSWORD*********************:0CB6948805F797BF2A82807973B89537:::
```

User name | User ID | LM Hash | NTLM Hash

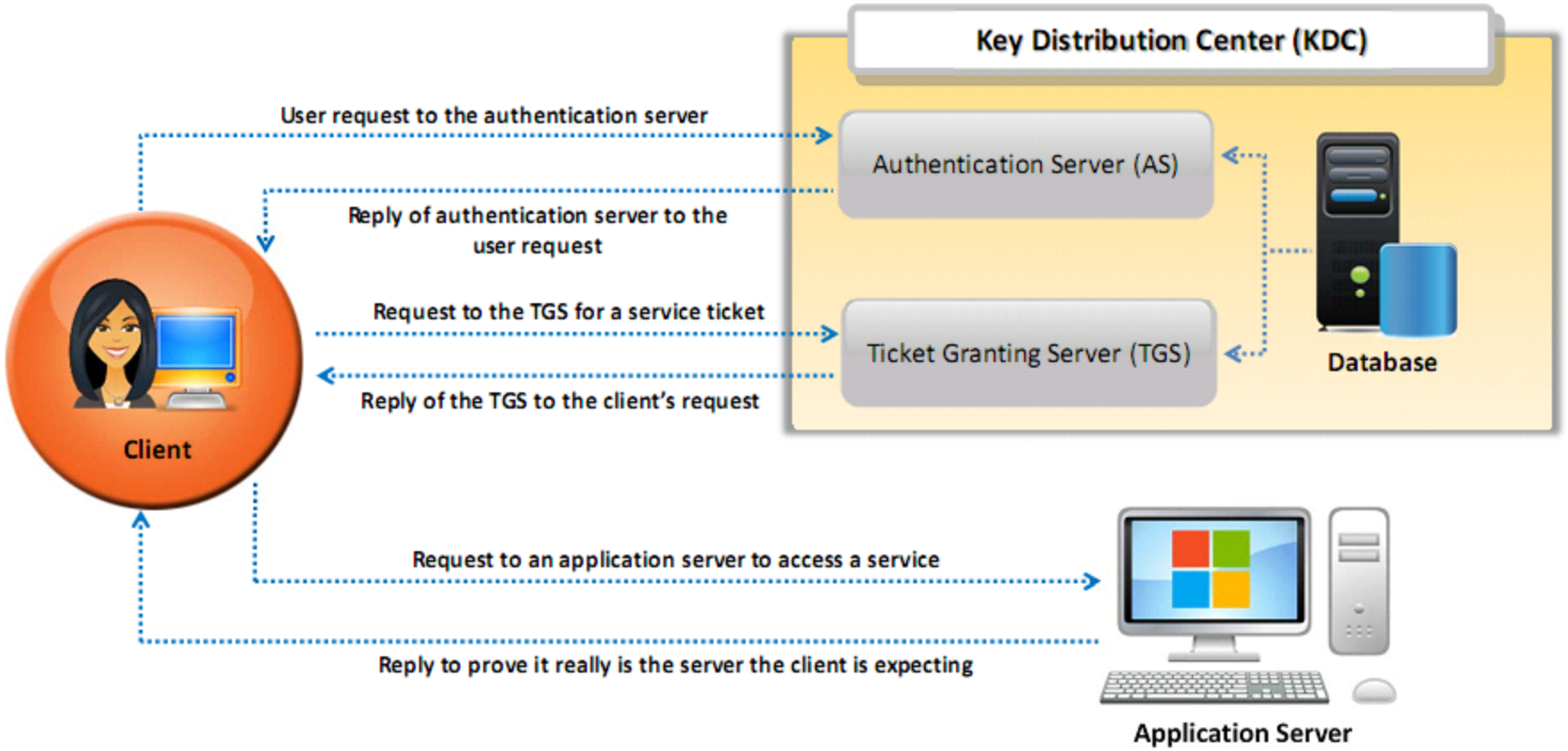
"LM hashes have been disabled in Windows Vista and later Windows operating systems, LM will be blank in those systems."

# NTLM Authentication Process




Note: Microsoft has upgraded its default authentication protocol to Kerberos, which provides strong authentication for client/server applications than NTLM.

# Kerberos Authentication



# Password Salting



- Password salting is a technique where **random string of characters are added** to the password before calculating their hashes

- **Advantage**: Salting makes it more difficult to reverse the hashes and defeat pre-computed hash attacks

Alice:root:b4ef21:3ba4303ce24a83fe0317608de02bf38d

Bob:root:a9c4fa:3282abd0308323ef0349dc7232c349ac

Cecil:root:209be1:a483b303c23af34761de02be038fde08

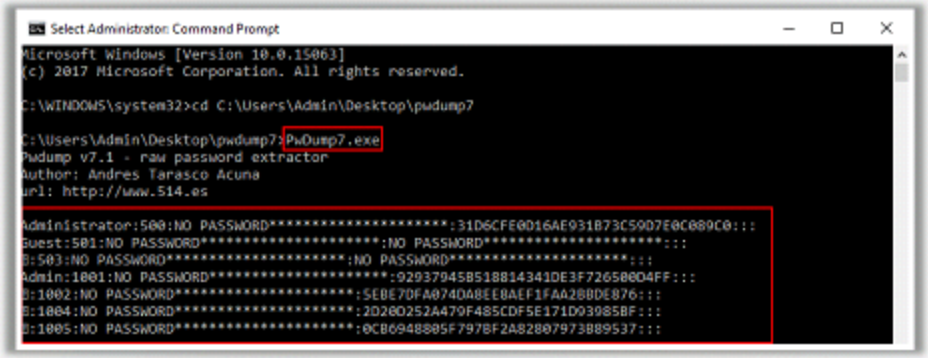
**Same password but different hashes due to different salts**

**Note: Windows password hashes are not salted**

# Tools to Extract the Password Hashes

## pwdump7

- pwdump7 **extracts LM** and **NTLM** password hashes of local user accounts from the **Security Account Manager** (SAM) database



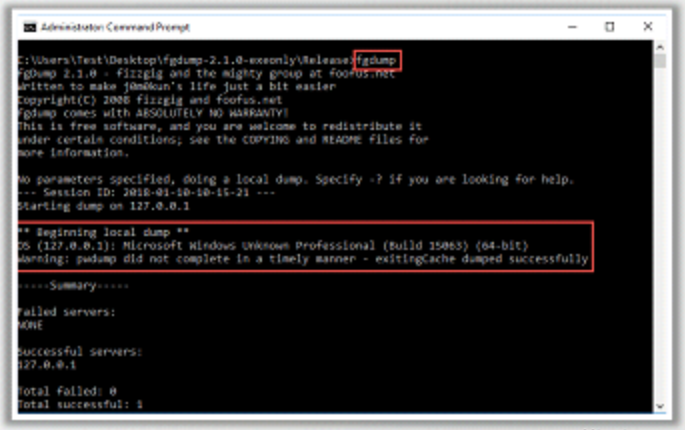

*https://www.tarasco.org*

## fgdump

- fgdump works like pwdump but also extracts **cached credentials** and allows **remote network** execution

```
fgdump.exe -h 192.168.0.10 -u AnAdministrativeUser
-p l4mep4ssw0rd
```

Dumps a remote machine (192.168.0.10) using a specified user



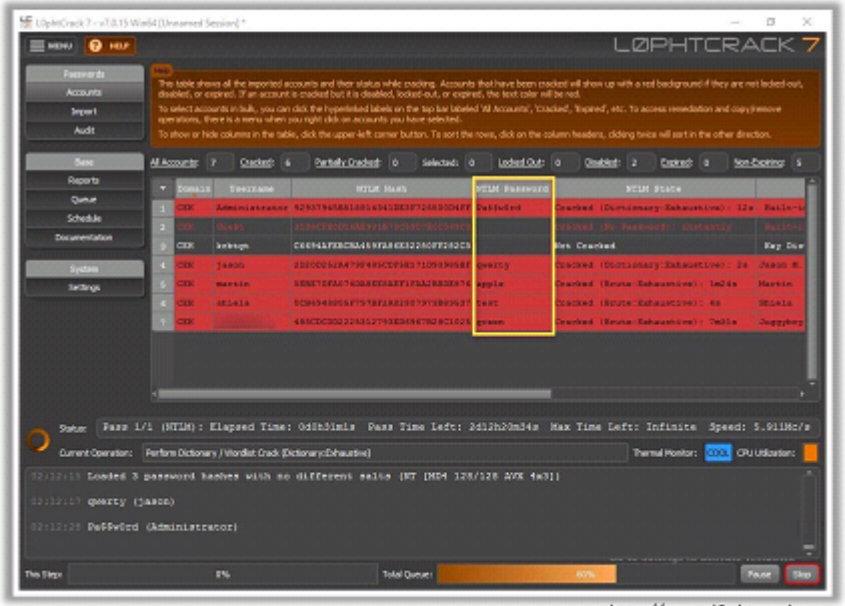

*http://foofus.net*

**Note: These tools must be run with administrator privileges**

# Password Cracking Tools: L0phtCrack and ophcrack

## L0phtCrack

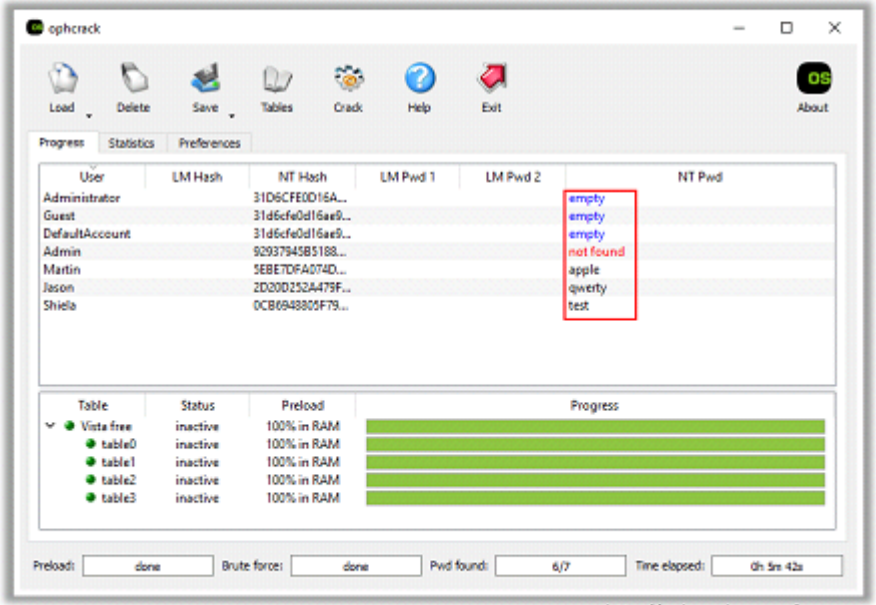
L0phtCrack is a password **auditing** and **recovery** application packed with features such as scheduling, hash extraction from 64-bit Windows versions, and networks monitoring and decoding




http://www.l0phtcrack.com

## ophcrack

ophcrack is a Windows password cracker based on **rainbow tables**. It comes with a Graphical User Interface and runs on multiple platforms

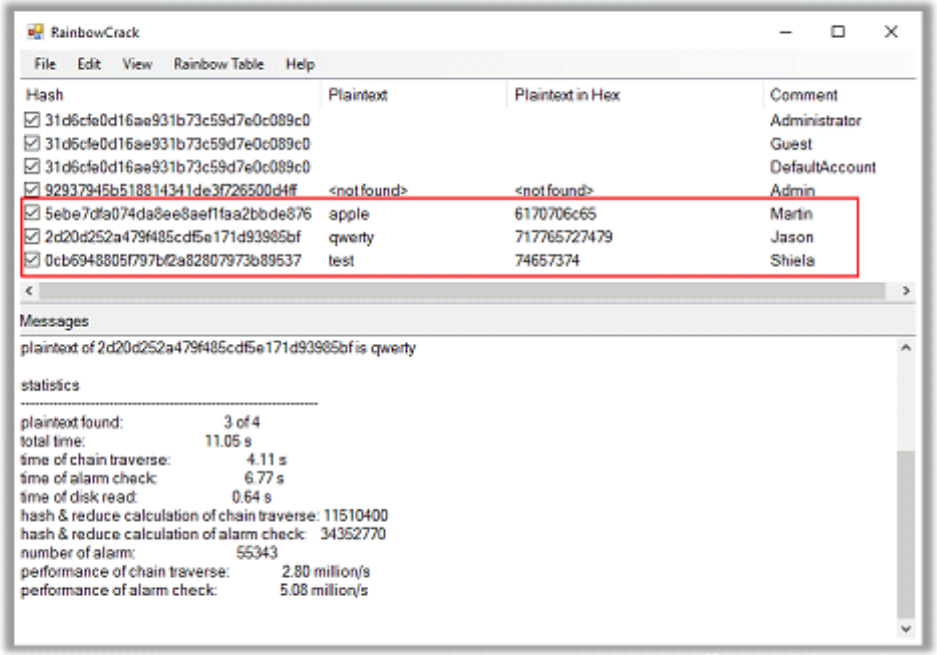



http://ophcrack.sourceforge.net

# Password Cracking Tools

## RainbowCrack

RainbowCrack cracks hashes with **rainbow tables**. It uses **time-memory tradeoff** algorithm to crack hashes




http://project-rainbowcrack.com

- **Cain & Abel**
  *http://www.oxid.it*
- **Windows Password Recovery Tool**
  *https://www.windowspasswordsrecovery.com*
- **Windows Password Key**
  *https://www.lostwindowspassword.com*
- **hashcat**
  *https://hashcat.net*
- **Passware Kit Forensic**
  *https://www.passware.com*

# How to Defend against Password Cracking



1. Enable **information security audit** to monitor and track password attacks
2. Do not use the **same password** during password change
3. Do not **share** passwords
4. Do not use passwords that can be found in a **dictionary**
5. Do not use **cleartext** protocols and protocols with **weak encryption**
6. Set the **password change policy** to 30 days
7. Avoid **storing passwords** in an unsecured location
8. Do not use any system's **default passwords**

# How to Defend against Password Cracking (Cont'd)

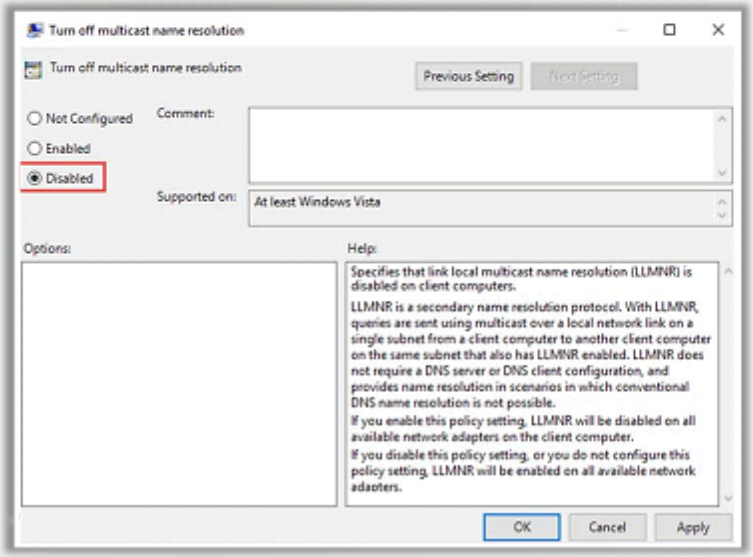


| | |
|---|---|
| 9 | Make passwords hard to guess by using **8-12 alphanumeric** characters in combination of uppercase and lowercase letters, numbers, and symbols |
| 10 | Ensure that applications **neither store** passwords to memory **nor write** them to disk in clear text |
| 11 | Use a **random string** (salt) as prefix or suffix with the password before encrypting |
| 12 | Enable **SYSKEY** with strong password to encrypt and protect the SAM database |
| 13 | Never use passwords such as **date of birth**, spouse, or child's or pet's name |
| 14 | Monitor the **server's logs** for brute force attacks on the users accounts |
| 15 | Lock out an account subjected to too many **incorrect password** guesses |

# How to Defend against LLMNR/NBT-NS Poisoning

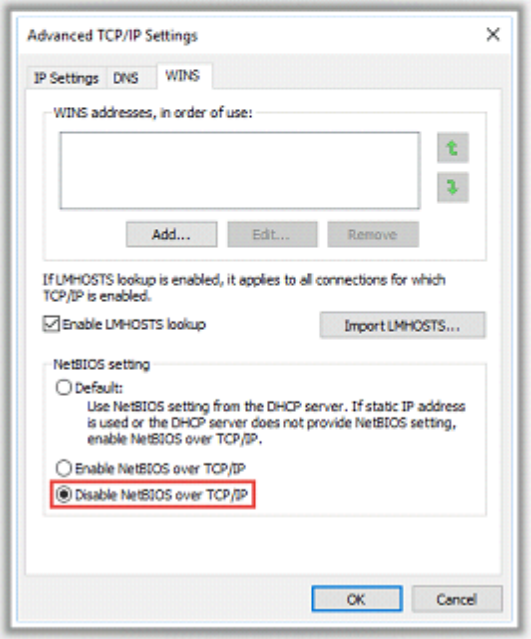
## Disable LMBNR

- Open **Local Group Policy Editor** and navigate to **Local Computer Policy → Computer Configuration → Administrative Templates → Network → DNS Client**
- In DNS client, double-click on **Turn off multicast name resolution**
- Select the **Disabled** radio button and then click **OK**




## Disable NBT-NS

- Open **Control Panel** and navigate to **Network and Internet → Network and Sharing Center** and click on **Change adapter settings** option present on the right side
- Right-click on the network adapter and click **Properties**, select **TCP/IPv4** and then click **Properties**
- Under **General** tab, go to **Advanced → WINS**
- From the NetBIOS options, check "**Disable NetBIOS over TCP/IP**" radio button and click **OK**



# Module Flow

1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# Privilege Escalation

- An attacker can gain access to the network using a **non-admin user account** and the next step would be to gain administrative privileges
- Attacker performs privilege escalation attack which takes advantage of **design flaws**, **programming errors**, **bugs**, and **configuration oversights** in the OS and software application to gain administrative access to the network and its associated applications
- These privileges allows attacker to **view critical/sensitive information**, delete files, or install malicious programs such as viruses, Trojans, worms, etc.

## Types of Privilege Escalation

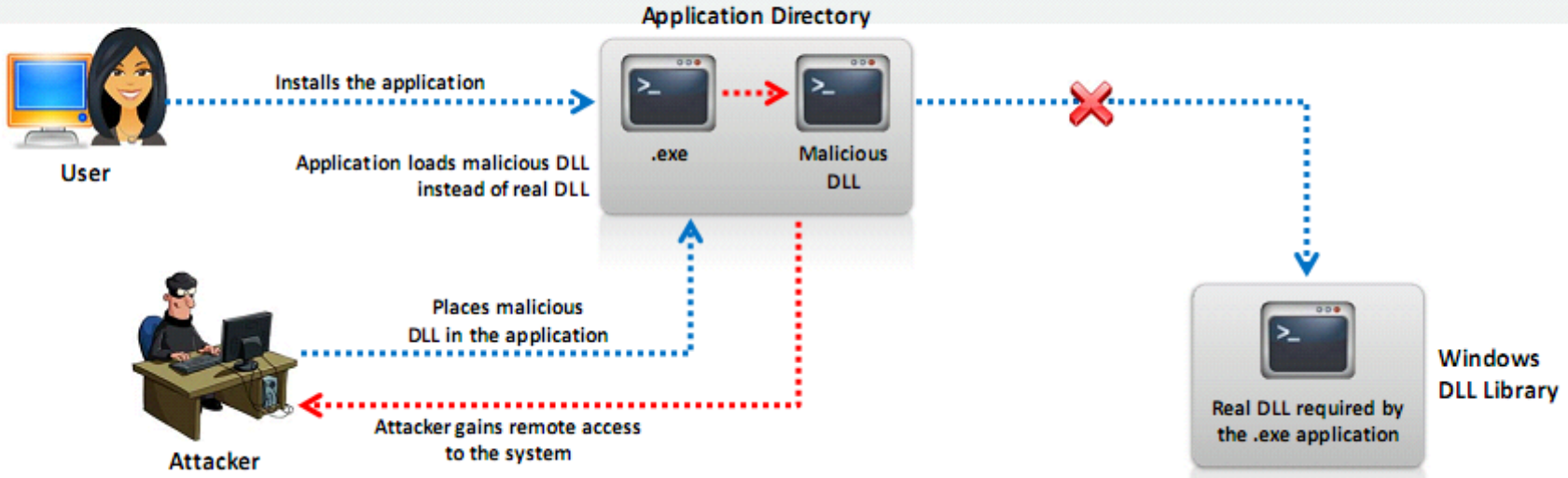
2. **Horizontal Privilege Escalation**
   - Refers to acquiring the same level of privileges that already has been granted but assuming the identity of another user with the similar privileges
1. **Vertical Privilege Escalation**
   - Refers to gaining higher privileges than the existing

# Privilege Escalation Using DLL Hijacking



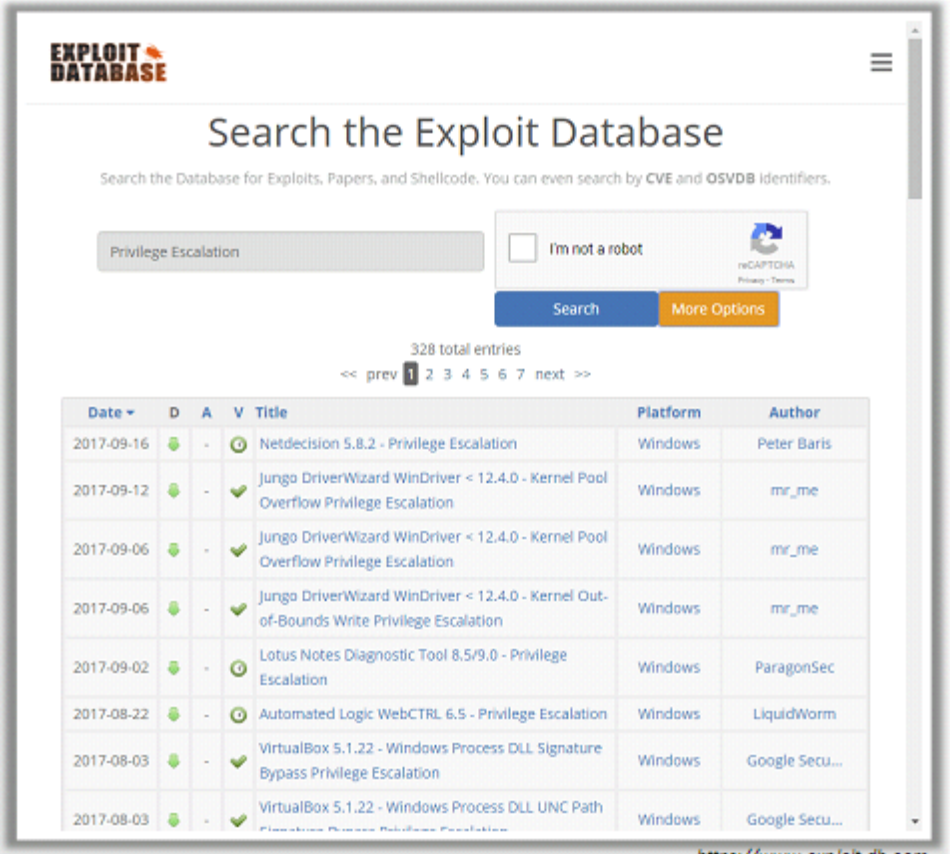
Most Windows applications do not use the **fully qualified path** when loading an external DLL library instead they search directory from which they have been loaded first

If attackers can place a **malicious DLL in the application directory**, it will be executed in place of the real DLL



# Privilege Escalation by Exploiting Vulnerabilities

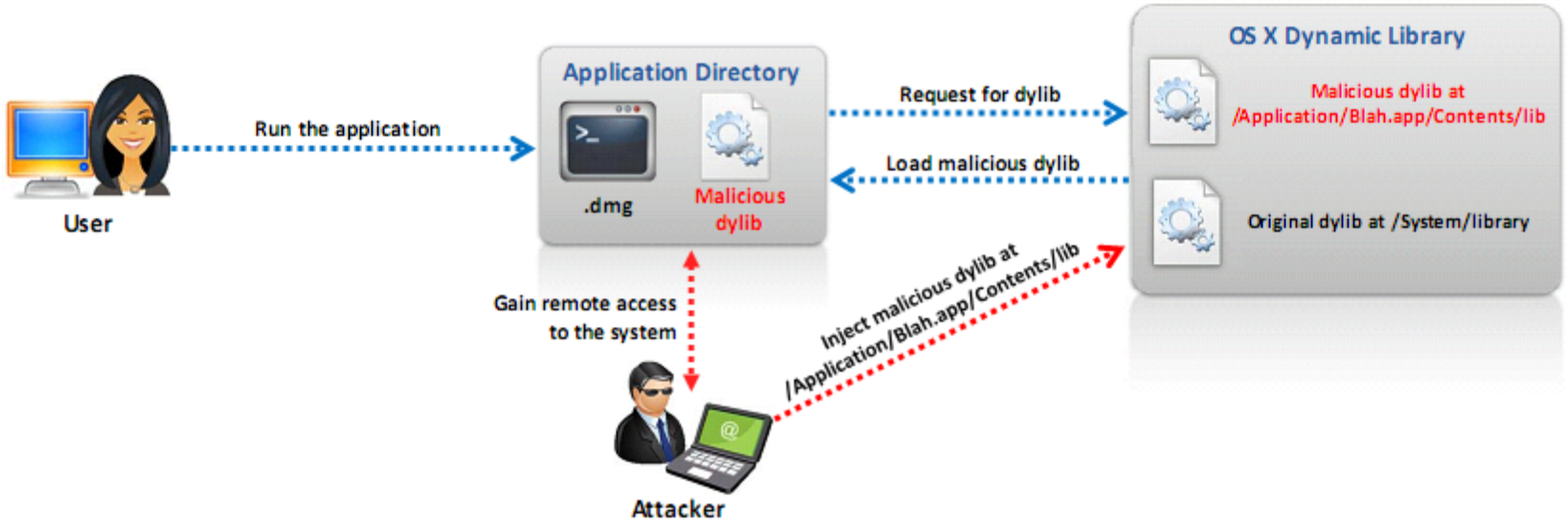
- Attackers **exploit software vulnerabilities** by taking advantage of programming flaws in a program, service, or within the operating system software or kernel to **execute malicious code**
- Exploiting software vulnerabilities allows attacker to execute a command or binary on a target machine to **gain higher privileges** than the existing or **bypass security mechanisms**
- Attackers using these exploits can access **privileged user accounts** and credentials
- Attackers search for an exploit based on the OS and software application on exploit sites such as **SecurityFocus** (http://www.securityfocus.com), **Exploit Database** (https://www.exploit-db.com), etc.




https://www.exploit-db.com

# Privilege Escalation Using Dylib Hijacking

- In OS X, applications while **loading an external dylib** (dynamic library), the loader searches for dylib in multiple directories
- If attackers can **inject a malicious dylib** in one of the primary directories, it will be executed in place of the original dylib



# Privilege Escalation Using Spectre and Meltdown Vulnerabilities



- Spectre and Meltdown are vulnerabilities found in **the design of the modern processor chips** from AMD, ARM, and Intel
- The **performance and CPU optimizations** in the processors such as branch prediction, out of order execution, caching, and speculative execution lead to these vulnerabilities
- Attackers exploit these vulnerabilities to gain unauthorized access and **steal critical system information such as credentials**, **secret keys**, etc. stored in the application's memory to escalate privileges

## Spectre Vulnerability

- Attackers may take advantage of this vulnerability to **read adjacent memory locations of a process** and access information for which he/she is not authorized
- Using this vulnerability an attacker can even **read the kernel memory** or perform a web based attack using JavaScript

## Meltdown Vulnerability

- Attackers may take advantage of this vulnerability to **escalate privileges by forcing an unprivileged process** to read other adjacent memory locations such as kernel memory and physical memory
- This leads to revealing of critical system information such as **credentials**, **private keys**, etc.

# Other Privilege Escalation Techniques



| | |
|---|---|
| **Access Token Manipulation** | Windows operating system uses access tokens to **determine the security context** of a process or thread<br>Attackers can obtain access tokens of other users or generate **spoofed tokens** to escalate privileges and perform malicious activities by evading detection |
| **Application Shimming** | Windows Application Compatibility Framework, shim is used to **provide compatibility** between the older and newer versions of Windows operating system<br>Shims like **RedirectEXE**, **injectDLL**, and **GetProcAddress** can be used by attackers to escalate privileges, install backdoors, disable Windows defender, etc. |
| **File System Permissions Weakness** | If the file system permissions of binaries are not properly set, an attacker can **replace the target binary** with a malicious file<br>If the process that is executing this binary is having **higher level permissions** then the malicious binary also executes under higher level permissions |
| **Path Interception** | Applications include many **weaknesses** and **misconfigurations** like unquoted paths, path environment variable misconfiguration, and search order hijacking that lead to path interception<br>Path interception helps an attacker to **maintain persistence** on a system and **escalate privileges** |
| **Scheduled Task** | **Windows Task Scheduler** along with utilities such as 'at' and 'schtasks' can be used to schedule programs that can be executed at a specific date and time<br>Attacker can use this technique to **execute malicious programs** at system startup, maintain persistence, perform remote execution, escalate privileges, etc. |

# Other Privilege Escalation Techniques (Cont'd)



**Launch Daemon**

- Launchd is used in MacOS and OS X boot up to complete the system initialization process by loading parameters for each launch-on-demand system-level daemon
- Daemons have plists that are linked to executables that run at start up
- Attacker can alter the launch daemon's executable to maintain persistence or to escalate privileges

**Plist Modification**

- Plist files in MacOS and OS X describe when programs should execute, executable file path, program parameters, required OS permissions, etc.
- Attackers alter plist files to execute malicious code on behalf of a legitimate user to escalate privileges

**Setuid and Setgid**

- In Linux and MacOS, if an application uses setuid or setgid then the application will execute with the privileges of the owning user or group
- An attacker can exploit the applications with the setuid or setgid flags to execute malicious code with elevated privileges

**Web Shell**

- A Web shell is a web-based script that allows access to a web server
- Attackers create web shells to inject malicious script on a web server to maintain persistent access and escalate privileges

# How to Defend Against Privilege Escalation (Cont'd)



1. Restrict the **interactive logon privileges**
2. Use **encryption technique** to protect sensitive data
3. Run users and applications on the **least privileges**
4. Reduce the **amount of code** that runs with particular privilege
5. Implement **multi-factor authentication** and **authorization**
6. Perform **debugging** using bounds checkers and stress tests
7. Run services as **unprivileged accounts**
8. Test operating system and **application coding errors** and **bugs** thoroughly
9. Implement a **privilege separation methodology** to limit the scope of programming errors and bugs
10. **Patch** and **update** the kernel regularly

# How to Defend Against Privilege Escalation (Cont'd)



11. Change User Account Control settings to "**Always Notify**"
12. Restrict users from writing files to the **search paths** for applications
13. Continuously **monitor file system permissions** using auditing tools
14. **Reduce the privileges** of users and groups so that only legitimate administrators can make service changes
15. Use **whitelisting tools** to identify and block malicious software
16. Use **fully qualified paths** in all the Windows applications
17. Ensure that all executables are placed in **write-protected directories**
18. In Mac operating systems, **make plist files read-only**
19. **Block unwanted system utilities** or software that may be used to schedule tasks
20. Patch and update the **web servers** regularly

# Module Flow

1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# Executing Applications



- Attackers execute malicious applications in this stage. This is called "**owning**" the system
- Attacker executes malicious programs **remotely in the victim's machine** to gather information that leads to exploitation or loss of privacy, **gain unauthorized access** to system resources, **crack the password**, capture the screenshots, install backdoor to maintain easy access, etc.

**Keyloggers**

**Backdoors**

**Malicious Programs Attackers Execute on Target Systems**

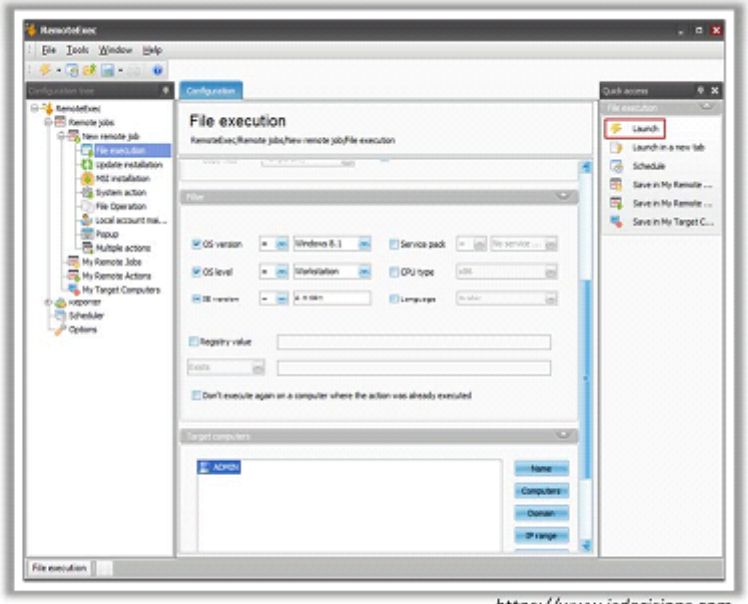
**Spyware**

**Crackers**

# Tools for Executing Applications

## RemoteExec

- RemoteExec **remotely installs applications**, **executes programs/scripts**, and updates files and folders on Windows systems throughout the network
- It allows attacker to **modify the registry**, **change local admin passwords**, **disable local accounts**, and copy/ update/delete files and folders



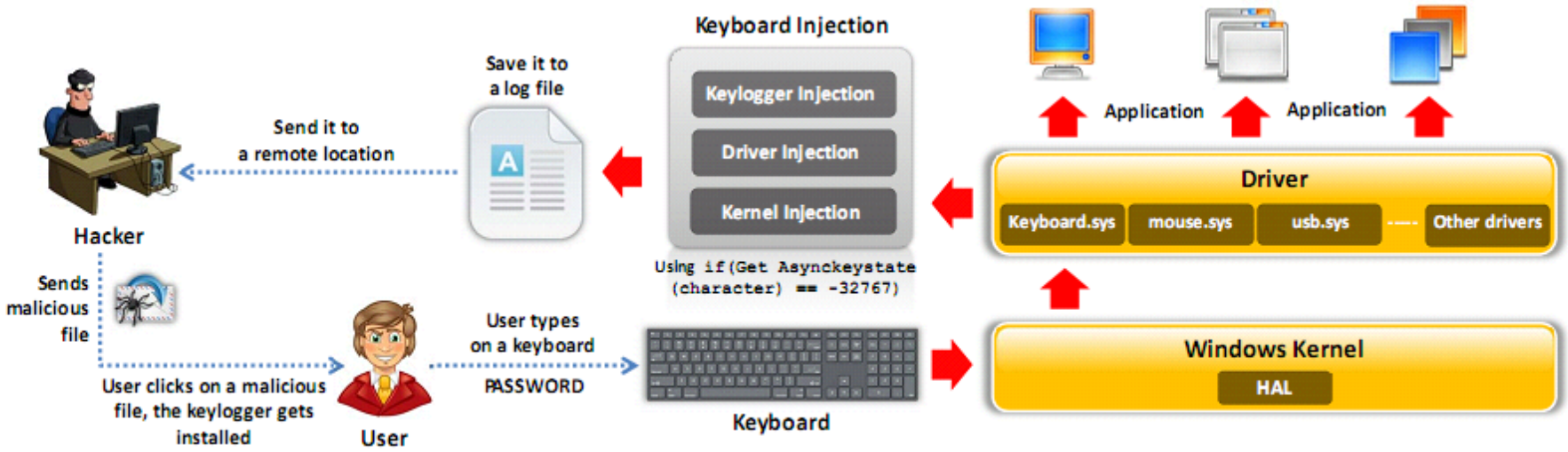

https://www.isdecisions.com

## Tools for Executing Applications

- PDQ Deploy (*https://www.pdq.com*)
- Dameware Remote Support (*https://www.dameware.com*)
- ManageEngine Desktop Central (*https://www.manageengine.com*)
- PsExec (*https://docs.microsoft.com*)
- TheFatRat (*https://github.com*)

# Keylogger

- Keystroke loggers are programs or hardware devices that **monitor each keystroke** as user types on a keyboard, logs onto a file, or transmits them to a remote location
- Legitimate applications for keyloggers include in office and industrial settings to monitor **employees' computer activities** and in home environments where parents can monitor and spy on **children's activity**
- It allows attacker to **gather confidential information** about victim such as email ID, passwords, banking details, chat room activity, IRC, instant messages, etc.
- Physical keyloggers are placed between the **keyboard hardware** and the **operating system**



# Types of Keystroke Loggers

# Hardware Keyloggers


https://www.keydemon.com

## Hardware Keyloggers Vendors

**KeyCarbon**
*http://www.keycarbon.com*

**Keyllama Keylogger**
*https://Keyllama.com*

**Keyboard logger**
*https://www.detective-store.com*

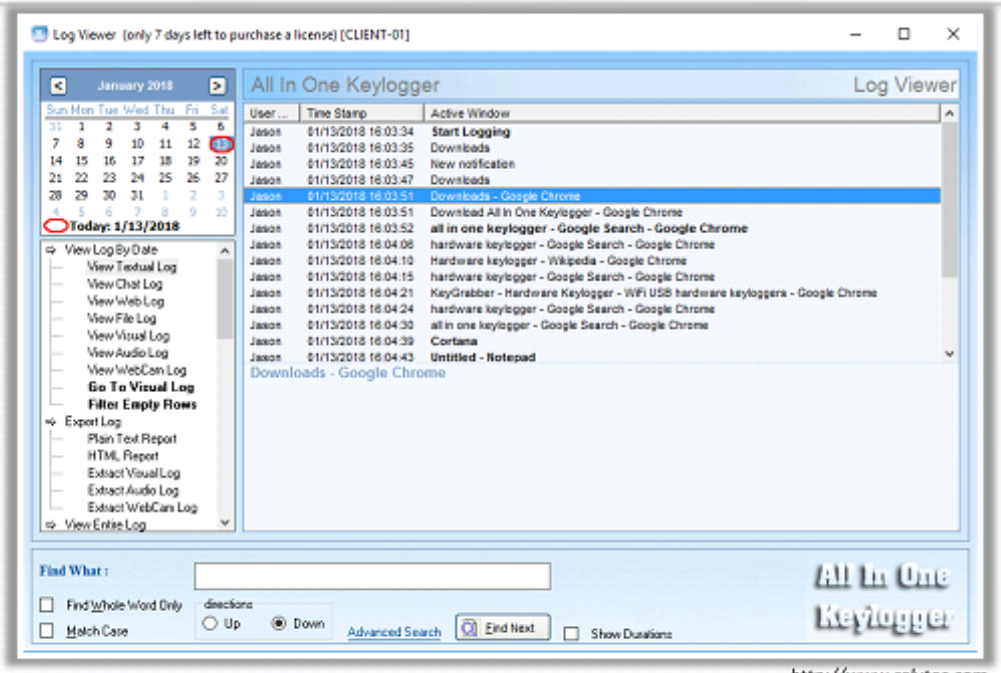
**KeyGhost**
*http://www.keyghost.com*

**KeyCobra**
*http://www.keycobra.com*

# Keyloggers for Windows

## All In One Keylogger

- It allows you to **secretly track** all activities from all computer users and **automatically receive logs** to a desired email/FTP/LAN accounting



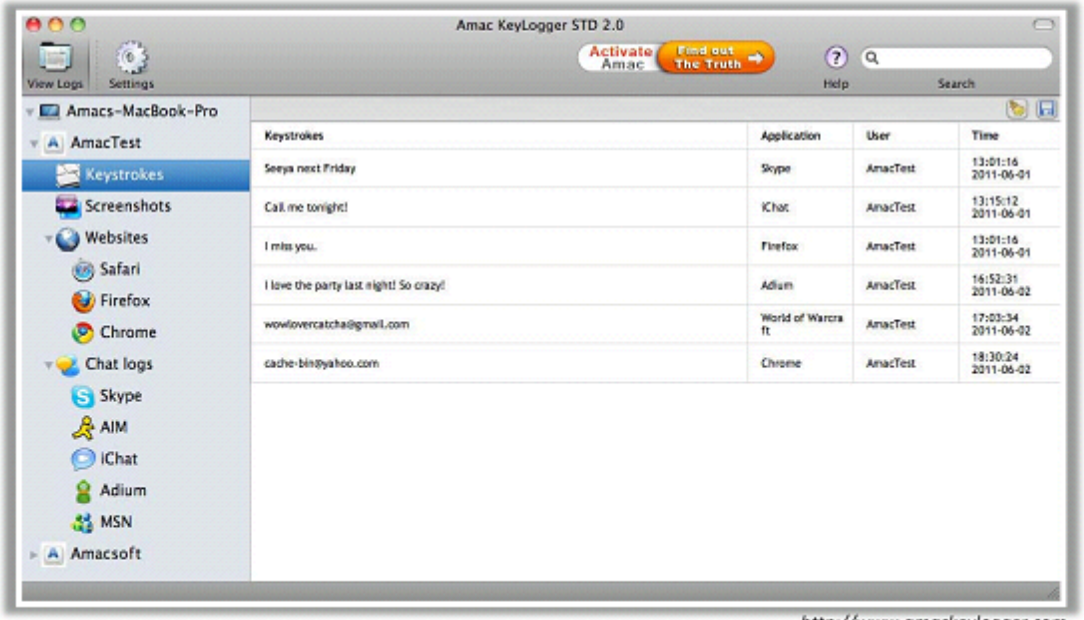

http://www.relytec.com

- **Spyrix Personal Monitor**
  *http://www.spyrix.com*
- **SoftActivity Activity Monitor**
  *https://www.softactivity.com*
- **Elite Keylogger**
  *https://www.elitekeyloggers.com*
- **Keylogger Spy Monitor**
  *http://ematrixsoft.com*
- **Micro Keylogger**
  *https://www.microkeylogger.com*

# Keyloggers for Mac

## Amac Keylogger

Amac Keylogger for Mac invisibly **records all keystrokes typed**, **IM chats**, **websites visited**, and takes screenshots and also sends all reports to the attacker by email, or upload everything to attacker's website




http://www.amackeylogger.com

**Elite Keylogger**
*https://www.elite-keylogger.net*

**Aobo Mac OS X KeyLogger**
*https://www.keylogger-mac.com*

**KidLogger for MAC**
*http://kidlogger.net*

**Perfect Keylogger for Mac**
*http://www.blazingtools.com*

**MAC Log Manager**
*http://www.keylogger.in*

# Spyware

- Spyware is a stealthy program that **records user's interaction** with the computer and Internet without the user's knowledge and sends them to the remote attackers
- Spyware **hides its process**, files, and other objects in order to avoid detection and removal
- It is similar to Trojan horse, which is usually bundled as a **hidden component of freeware programs** that can be available on the Internet for download
- It allows attacker to **gather information about a victim or organization** such as email addresses, user logins, passwords, credit card numbers, banking credentials, etc.

## Spyware Propagation

1. Drive-by download
2. Masquerading as anti-spyware
3. Web browser vulnerability exploits
4. Piggybacked software installation
5. Browser add-ons
6. Cookies

# Spyware: Spytech SpyAgent and Power Spy

## Spytech SpyAgent

- Spytech SpyAgent allows you to monitor everything users do on your computer


https://www.spytech-web.com

## Power Spy

- Power Spy secretly monitors and records all activities on your computer


http://emicrisoft.com

# Spyware

## Spyware

**ACTIVTrak**
*https://activtrak.com*

**Veriato 360**
*http://www.veriato.com*

**NetVizor**
*https://www.netvizor.net*

**Activity Monitor**
*https://www.softactivity.com*

**SoftActivity TS Monitor**
*https://www.softactivity.com*

## USB Spyware

**USB Analyzer**
*https://www.eltima.com*

**USB Monitor**
*https://www.hhdsoftware.com*

**USBDeview**
*http://www.nirsoft.net*

**Advanced USB Port Monitor**
*https://www.aggsoft.com*

**USB Monitor Pro**
*http://www.usb-monitor.com*

## Audio Spyware


**Spy Voice Recorder**
*http://www.mysuperspy.com*

**Spy Audio Listening Device**
*http://www.securityplanet.co*

**Spy USB Voice Recorder**
*http://www.securityplanet.co*

**Audio Spy**
*http://www.topsecretsoftware.com*

**Voice Activated Flash Drive Voice Recorder**
*http://www.spytecinc.com*

# Spyware (Cont'd)



## Video Spyware

**Movavi Video Editor**
*https://www.movavi.com*

**Free2X Webcam Recorder**
*http://www.free2x.com*

**iSpy**
*https://www.ispyconnect.com*

**NET Video Spy**
*https://www.sarbash.com*

**Eyeline Video Surveillance Software**
*http://www.nchsoftware.com*

## Telephone/Cellphone Spyware

**Phone Spy**
*http://www.phonespysoftware.com*

**XNSPY**
*https://xnspy.com*

**iKeyMonitor**
*https://ikeymonitor.com*

**OneSpy**
*https://www.onespy.in*

**TheTruthSpy**
*http://thetruthspy.com*

## GPS Spyware

**Spyera**
*https://spyera.com*

**mSpy**
*https://www.mspy.com*

**MOBILE SPY**
*http://www.mobile-spy.com*

**MobiStealth**
*http://www.mobistealth.com*

**FlexiSPY**
*https://www.flexispy.com*

# How to Defend Against Keyloggers



1. Use pop-up blocker and avoid opening junk emails
2. Install anti-spyware/antivirus programs and keeps the signatures up to date
3. Install professional firewall software and anti-keylogging software
4. Recognize phishing emails and delete them
5. Update and patch system software regularly
6. Do not click on links in unwanted or doubtful emails that may point to malicious sites
7. Use keystroke interference software, which inserts randomized characters into every keystroke
8. Scan the files before installing and use registry editor or process explorer to check for the keystroke loggers
9. Use Windows on-screen keyboard accessibility utility to enter the password or any other confidential information
10. Install a host-based IDS, which can monitor your system and disable the installation of keyloggers
11. Use automatic form-filling password manager or virtual keyboard to enter user name and password
12. Use software that frequently scans and monitors the changes in the system or network

# How to Defend Against Keyloggers (Cont'd)



## Hardware Keylogger Countermeasures

01 Restrict physical access to sensitive computer systems

02 Periodically check all the computers and check whether there is any hardware device connected to the computer

03 Use encryption between the keyboard and its driver

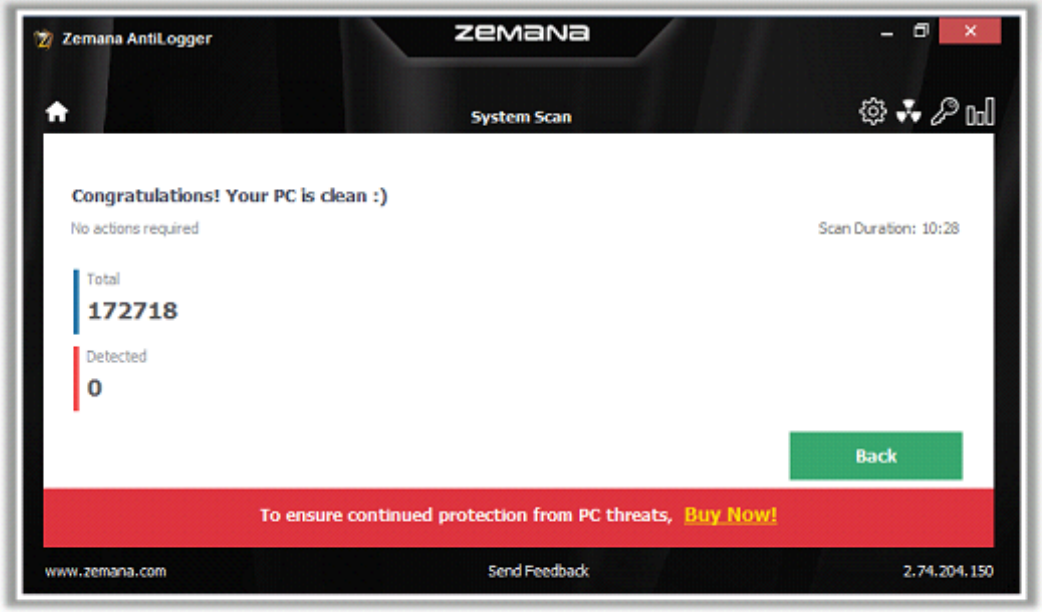
04 Use an anti-keylogger that detects the presence of a hardware keylogger such as Oxynger KeyShield

05 Use an on-screen keyboard and click on it by using a mouse

# Anti-Keylogger

## Zemana AntiLogger

It keeps track of who is doing what on your PC. It monitors your PC against the bad guys and prevents any kind of attempts to record or steal your private data and blocks any kind of suspicious activity




https://www.zemana.com

- **GuardedID** — https://www.strikeforcecpg.com
- **KeyScrambler** — https://www.qfxsoftware.com
- **SpyShelter Free Anti-Keylogger** — https://www.spyshelter.com
- **DefenseWall HIPS** — http://www.softsphere.com
- **Elite Anti Keylogger** — http://www.elite-antikeylogger.com

# How to Defend Against Spyware

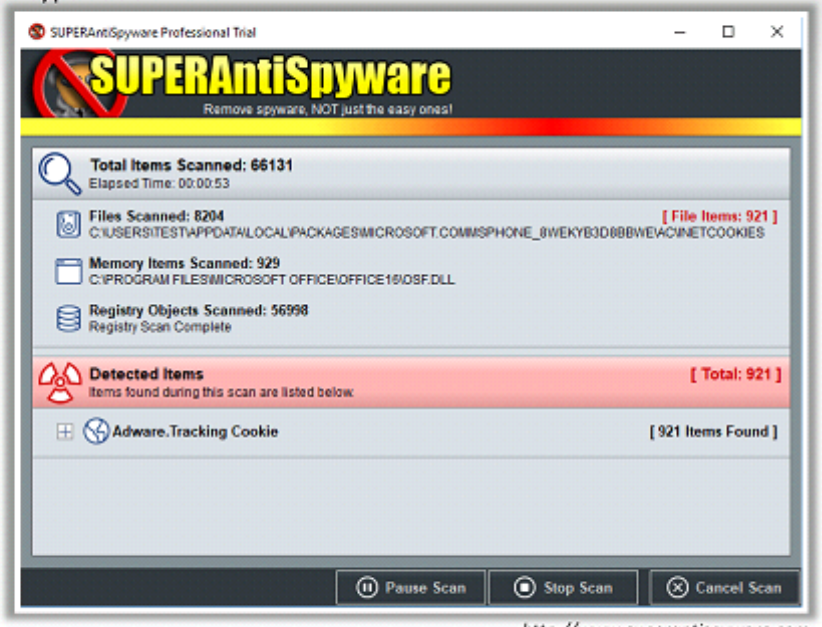


1. Try to avoid using any computer system which is not totally under your control
2. Adjust browser security settings to medium or higher for Internet zone
3. Be cautious about suspicious emails and sites
4. Enable firewall to enhance the security level of the computer
5. Update the software regularly and use a firewall with outbound protection
6. Regularly check task manager report and MS configuration manager report
7. Update virus definition files and scan the system for spyware regularly
8. Install and use anti-spyware software
9. Perform web surfing safely and download cautiously
10. Do not use administrative mode unless it is necessary
11. Keep your operating system up to date
12. Do not download free music files, screensavers, or smiley faces from Internet
13. Beware of pop-up windows or web pages. Never click anywhere on these windows
14. Carefully read all disclosures, including the license agreement and privacy statement before installing any application

# Anti-Spyware

## SUPERAntiSpyware

- Identify potentially unwanted programs and securely removes them
- Detect and remove Spyware, Adware, Malware, Trojans, Dialers, Worms, Keyloggers, Hijackers, Parasites, Rootkits, Rogue security products, and many other types of threats




http://www.superantispyware.com

- **Kaspersky Internet Security 2018**
  *https://www.kaspersky.com*
- **SecureAnywhere Internet Security Complete**
  *https://www.webroot.com*
- **adaware antivirus free**
  *https://www.adaware.com*
- **MacScan**
  *https://www.securemac.com*
- **Norton AntiVirus Basic**
  *https://in.norton.com*

# Module Flow

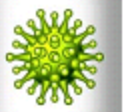
1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# Rootkits

- Rootkits are programs that **hide their presence** as well as attacker's malicious activities, granting them full access to the server or host at that time and also in future
- Rootkits replace certain operating system calls and utilities with its own **modified versions** of those routines that in turn undermine the security of the target system causing **malicious functions** to be executed
- A typical rootkit comprises of backdoor programs, DDoS programs, packet sniffers, log-wiping utilities, IRC bots, etc.

### Attacker places a rootkit by:

- Scanning for **vulnerable** computers and servers on the web
- **Wrapping** it in a special package like games
- Installing it on the public computers or corporate computers through **social engineering**
- Launching **zero day attack** (privilege escalation, buffer overflow, Windows kernel exploitation, etc.)



### Objectives of Rootkit

- To **root** the host system and **gain remote backdoor** access
- To mask **attacker tracks** and presence of malicious applications or processes
- To gather **sensitive data**, **network traffic**, etc. from the system to which attackers might be restricted or possess no access
- To store other **malicious programs** on the system and act as a server resource for bot updates

# Types of Rootkits



### Hypervisor Level Rootkit

Acts as a hypervisor and modifies the boot sequence of the computer system to load the host operating system as a **virtual machine**

### Boot Loader Level Rootkit

Replaces the original **boot loader** with one controlled by a remote attacker

### Hardware/Firmware Rootkit

Hides in hardware devices or platform firmware which is not inspected for **code integrity**

### Application Level Rootkit

Replaces regular **application binaries** with fake Trojan or modifies the behavior of existing applications by injecting malicious code

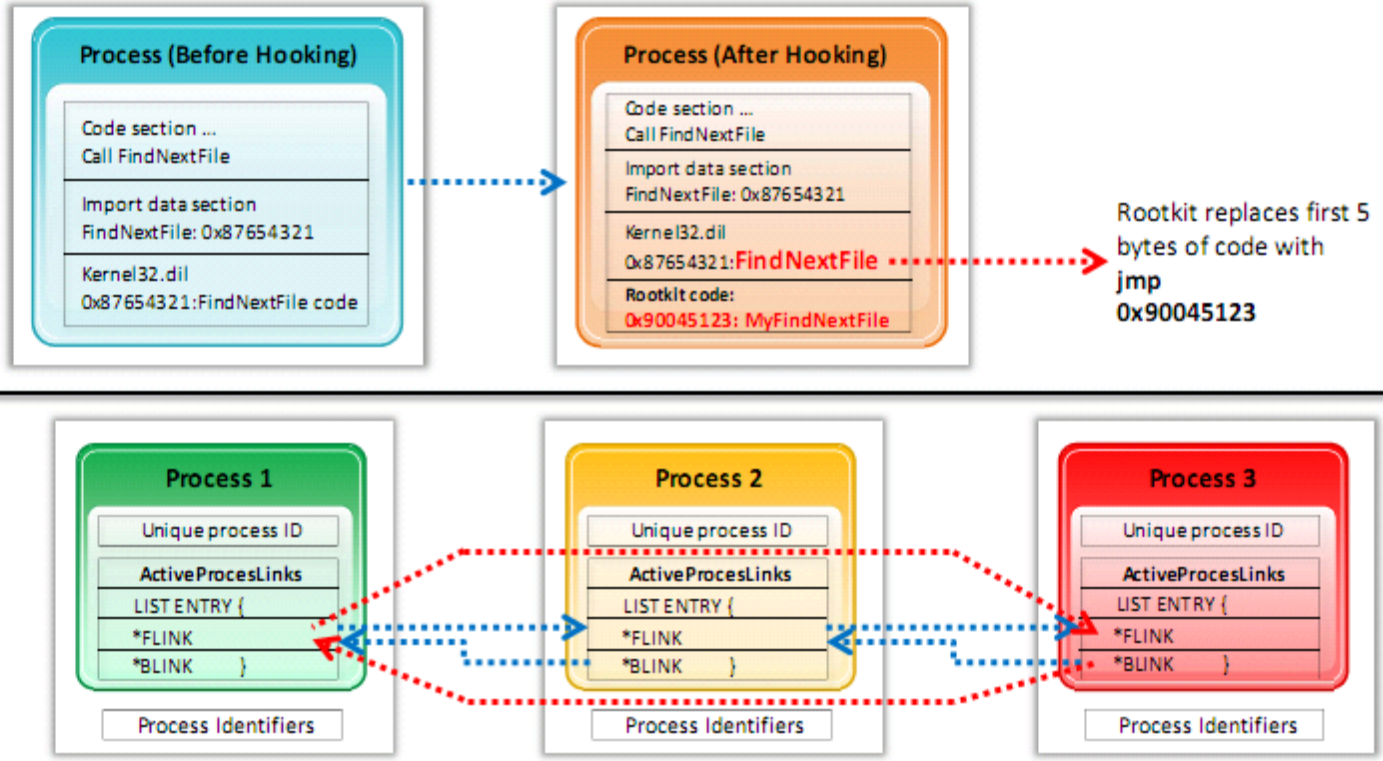
### Kernel Level Rootkit

Adds malicious code or replaces original **OS kernel** and **device driver codes**

### Library Level Rootkits

Replaces original system calls with fake ones to **hide information** about the attacker

# How Rootkit Works




DKOM rootkits hide a process by unlinking it from the process list

# Rootkits: Horse Pill and GrayFish



## Horse Pill

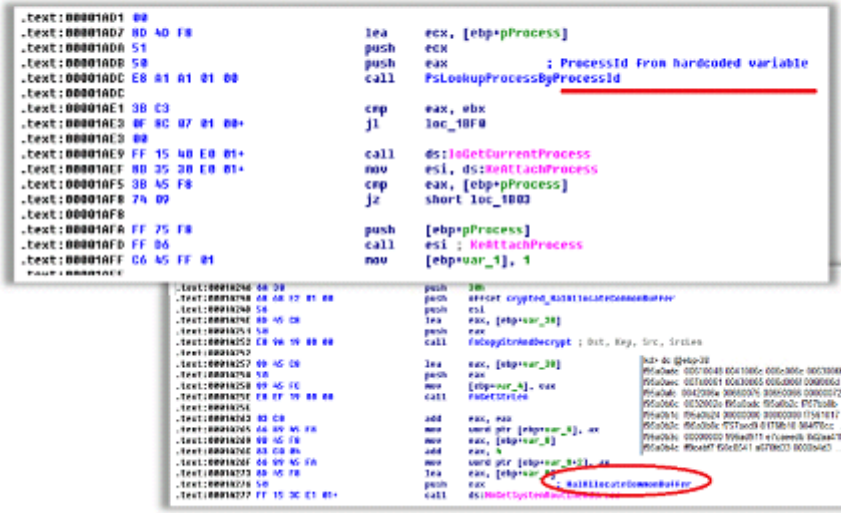
- Horse Pill is Linux kernel rootkit that resides inside the "**initrd**" using which it infects the system and deceives the system owner with the use of **container primitives**
- It has three important parts; **klibc-horsepill.patch, horsepill_setopt,** and **horsepill_infect**

```
root@gtfo:~# ls -l /proc/1/ns
total 0
lrwxrwxrwx 1 root root 0 Jul  8 16:47 ipc -> ipc:[4026531839]
lrwxrwxrwx 1 root root 0 Jul  8 16:47 mnt -> mnt:[4026531840]
lrwxrwxrwx 1 root root 0 Jul  8 16:47 net -> net:[4026531969]
lrwxrwxrwx 1 root root 0 Jul  8 16:47 pid -> pid:[4026531836]
lrwxrwxrwx 1 root root 0 Jul  8 16:47 user -> user:[4026531837]
lrwxrwxrwx 1 root root 0 Jul  8 16:47 uts -> uts:[4026531838]
```

```
root@gtfo:/usr/src/linux# cat -n include/linux/proc_ns.h | grep -A2 -B8
PROC_PID_INIT_INO
    31    /*
    32     * We always define these enumerators
    33     */
    34    enum {
    35            PROC_ROOT_INO        = 1,
    36            PROC_IPC_INIT_INO    = 0xEFFFFFFFU,
    37            PROC_UTS_INIT_INO    = 0xEFFFFFFEU,
    38            PROC_USER_INIT_INO   = 0xEFFFFFFDU,
    39            PROC_PID_INIT_INO    = 0xEFFFFFFCU,
    40            PROC_CGROUP_INIT_INO= 0xEFFFFFFBU,
    41    };
```

## GrayFish

- GrayFish is a Windows kernel rootkit that runs inside the Windows operating system and provides an effective mechanism, **hidden storage**, and malicious command execution while remaining invisible
- It injects its malicious code into the **boot record** which handles the launching of Windows at each step



# Rootkits: Sirefef and Necurs

## Sirefef

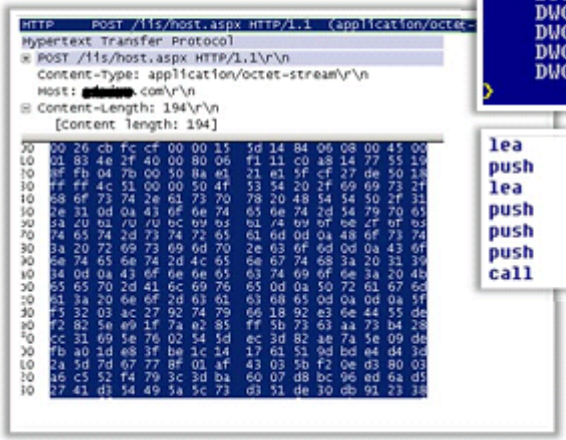
- Sirefef Rootkit or ZeroAccess gives attackers full access to your system while using stealth techniques in order to hide its presence from the affected device
- It hides itself by altering the internal processes of an operating system so that your antivirus and anti-spyware can't detect it

```
cmd.exe                          2956 Console                    0
wuauclt.exe                      3400 Console                    0
explorer.exe                     2952 Console                    0
2383950902:3385583473.exe        3012 Console                    0
taskmgr.exe                       856 Console                    0
ntvdm.exe                        1984 Console                    0
notepad.exe                      3148 Console                    0
tasklist.exe                     3188 Console                    0
wmiprvse.exe                     3204 Console                    0
```

```
C:\NU>cacls c:\BIN\prochack.exe
c:\BIN\prochack.exe Everyone:(NP)(special access:)
                                    DELETE
                                    READ_CONTROL
                                    WRITE_DAC
                                    WRITE_OWNER
                                    STANDARD_RIGHTS_REQUIRED
                                    FILE_READ_DATA
                                    FILE_WRITE_DATA
                                    FILE_APPEND_DATA
                                    FILE_READ_EA
                                    FILE_WRITE_EA
                                    FILE_EXECUTE
                                    FILE_DELETE_CHILD
                                    FILE_READ_ATTRIBUTES
                                    FILE_WRITE_ATTRIBUTES
```

## Necurs

- Necurs contains backdoor functionality, allowing remote access and control of the infected computer
- It monitors and filters network activity and has been observed to send spam and install rogue security software




```
Typedef struct NecursCnd {
    BYTE Reserved;
    DWORD CmdLength;
    DWORD Key1; //Prebuild key1
    DWORD Key2; //Prebuild key2
    DWORD CmdBuffer;
}
```

```
lea     eax, [ebp+CmdBufferLength]
push    eax             ; OUT_BufLen
lea     eax, [ebp+CmdBuffer]
push    eax             ; OUT_Buf
push    9CA1E108h       ; Skey2
push    0AFE8991Bh      ; Skey1
call    bNecurs_CmdSearchA
```

# Detecting Rootkits

| | |
|---|---|
| **Integrity-Based Detection** | It compares a snapshot of the **file system**, **boot records**, or **memory** with a known trusted baseline |
| **Signature-Based Detection** | This technique compares characteristics of all **system processes** and **executable files** with a database of known rootkit fingerprints |
| **Heuristic/Behavior - Based Detection** | Any **deviations in the system's normal activity** or behavior may indicate the presence of rootkit |
| **Runtime Execution Path Profiling** | This technique compares **runtime execution paths** of all system processes and executable files before and after the rootkit infection |
| **Cross View-Based Detection** | Enumerates key elements in the computer system such as **system files**, **processes**, and **registry keys** and compares them to an **algorithm** used to generate a similar data set that does not rely on the common APIs. Any discrepancies between these two data sets indicate the presence of rootkit |

# Steps for Detecting Rootkits



**Step 1**

Run "*dir /s /b /ah*" and "*dir /s /b /a-h*" inside the potentially infected OS and save the results


**Step 2**

Boot into a clean CD, run "*dir /s /b /ah*" and "*dir /s /b /a-h*" on the same drive and save the results

**Step 3**

Run a clean version of WinDiff on the two sets of results to detect file-hiding ghostware (i.e., invisible inside, but visible from outside)

**Note:** There will be some false positives. Also, this does not detect stealth software that hides in BIOS, video card, EEPROM, bad disk sectors, Alternate Data Streams, etc.

# How to Defend against Rootkits

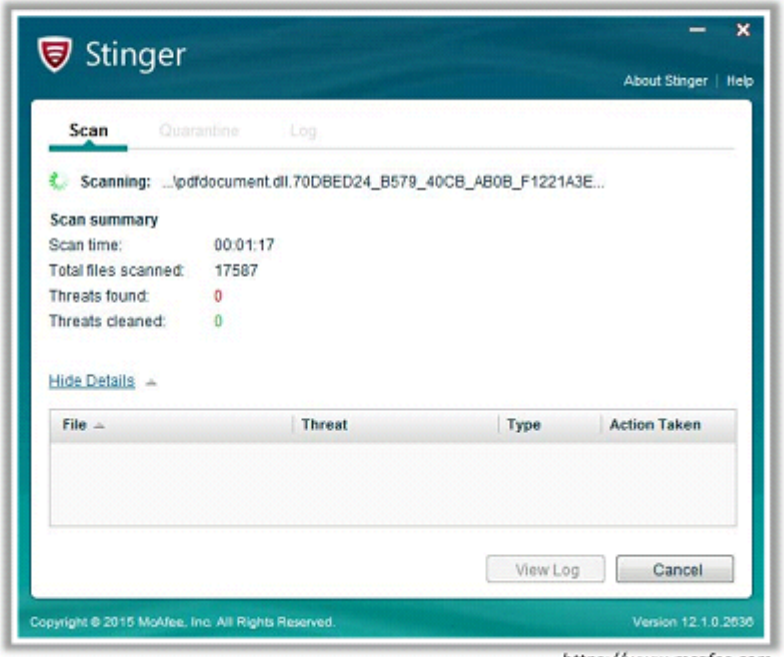


1. Reinstall OS/applications from a trusted source after backing up the critical data
2. Well-documented automated installation procedures need to be kept
3. Perform kernel memory dump analysis to determine the presence of rootkits
4. Harden the workstation or server against the attack
5. Educate staff not to download any files/programs from untrusted sources
6. Install network and host-based firewalls
7. Ensure the availability of trusted restoration media
8. Update and patch operating systems and applications
9. Verify the integrity of system files regularly using cryptographically strong digital fingerprint technologies
10. Update antivirus and anti-spyware software regularly
11. Avoid logging in an account with administrative privileges
12. Adhere to the least privilege principle
13. Ensure the chosen antivirus software posses rootkit protection
14. Do not install unnecessary applications and also disable the features and services not in use

# Anti-Rootkits

## Stinger

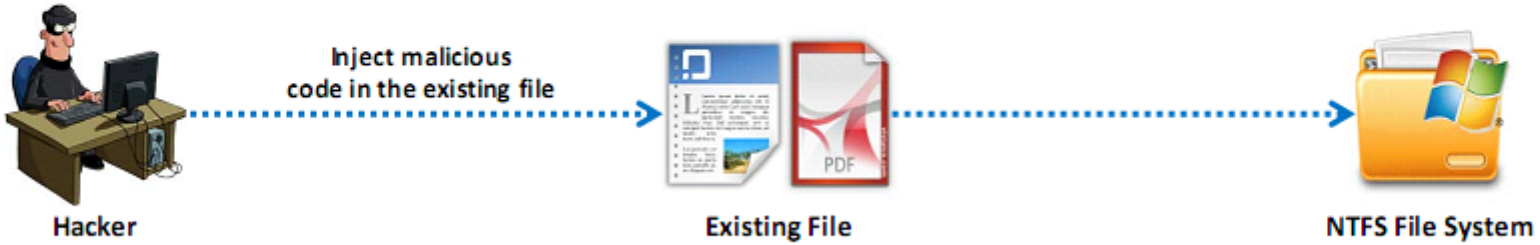
- Stinger scans rootkits, running processes, loaded modules, registry, and directory locations known to be used by **malware** on the machine




https://www.mcafee.com

- **Avast Free Antivirus**
  *https://www.avast.com*
- **TDSSKiller**
  *https://usa.kaspersky.com*
- **Malwarebytes Anti-Rootkit**
  *https://www.malwarebytes.com*
- **Rootkit Buster**
  *http://www.trendmicro.co.in*
- **UnHackMe**
  *http://www.greatis.com*

# NTFS Data Stream




NTFS Alternate Data Stream (ADS) is a **Windows hidden stream** which contains metadata for the file such as attributes, word count, author name and access, and modification time of the files

ADS is the ability to **fork data into existing files** without changing or altering their functionality, size, or display to file browsing utilities

ADS allows an attacker to **inject malicious code** in files on an accessible system and execute them without being detected by the user

# How to Create NTFS Streams



**Notepad is stream compliant application**

**Step 1**

- Launch `c:\>notepad myfile.txt:lion.txt`
- Click 'Yes' to create the new file, enter some data and Save the file

**Step 2**

- Launch `c:\>notepad myfile.txt:tiger.txt`
- Click 'Yes' to create the new file, enter some data and Save the file

**Step 3**

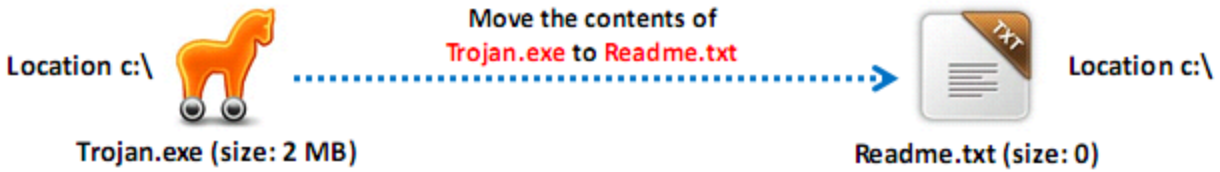
- View the file size of `myfile.txt` (It should be zero)

**Step 4**

- To view or modify the stream data hidden in step 1 and 2, use the following commands respectively:

```
notepad myfile.txt:lion.txt
notepad myfile.txt:tiger.txt
```

# NTFS Stream Manipulation




01 To move the contents of Trojan.exe to Readme.txt (stream):

```
C:\>type c:\Trojan.exe > c:\Readme.txt:Trojan.exe
```

02 To create a link to the Trojan.exe stream inside the Readme.txt file:

```
C:\>mklink backdoor.exe Readme.txt:Trojan.exe
```

03 To execute the Trojan.exe inside the Readme.txt (stream), type:

```
C:\>backdoor
```

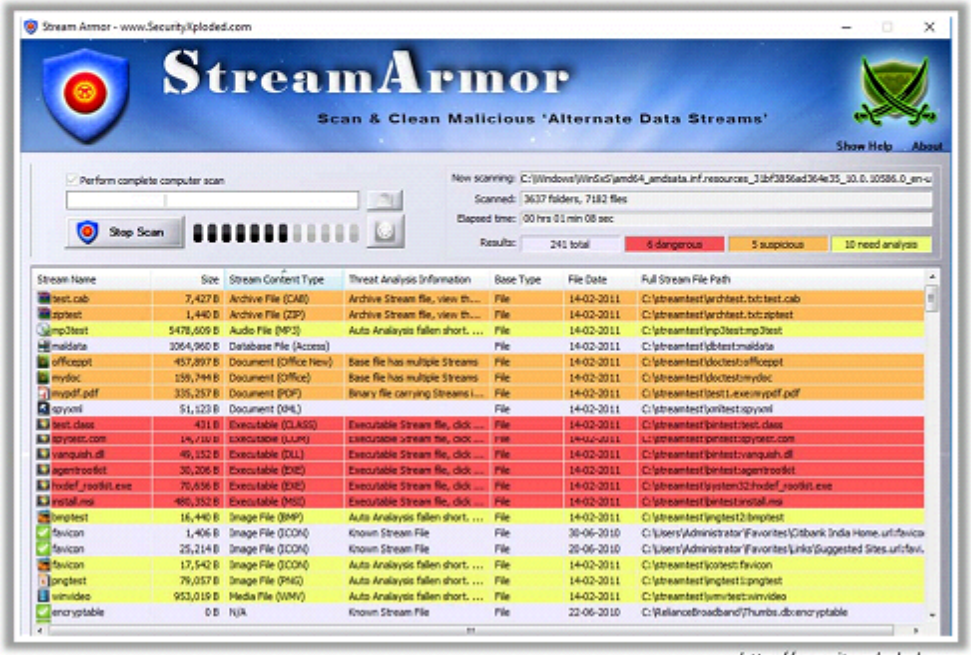
# How to Defend against NTFS Streams

- To delete NTFS streams, move the **suspected files** to FAT partition
- Use third-party **file integrity checker** such as Tripwire File Integrity Monitoring to maintain integrity of an NTFS partition files
- Use programs such as **Stream Detector**, **LADS**, **ADS Detector**, etc. to detect streams
- **Enable real-time antivirus scanning** to protect against execution of malicious streams in your system
- Use **up-to-date antivirus software** on your system

# NTFS Stream Detectors

## Stream Armor

- Stream Armor **discovers hidden Alternate Data Streams (ADS)** and cleans them completely from the system




http://securityxploded.com

- **Stream Detector**
  *http://www.novirusthanks.org*
- **Forensic Toolkit**
  *https://www.mcafee.com*
- **ADS Manager**
  *https://dmitrybrant.com*
- **ADS Scanner**
  *https://www.pointstone.com*
- **ADS Spy**
  *http://www.merijn.nu*

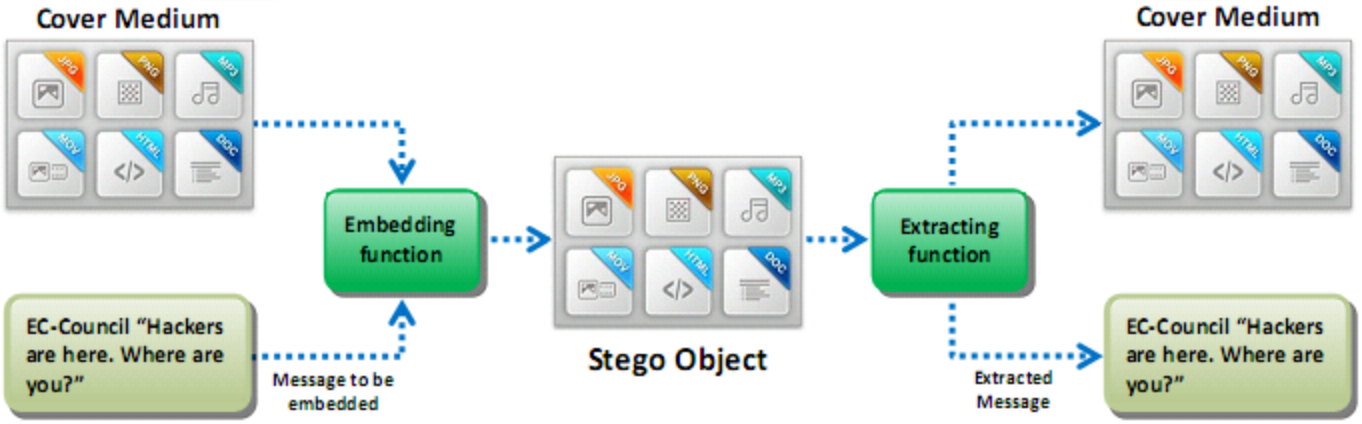
# What is Steganography?



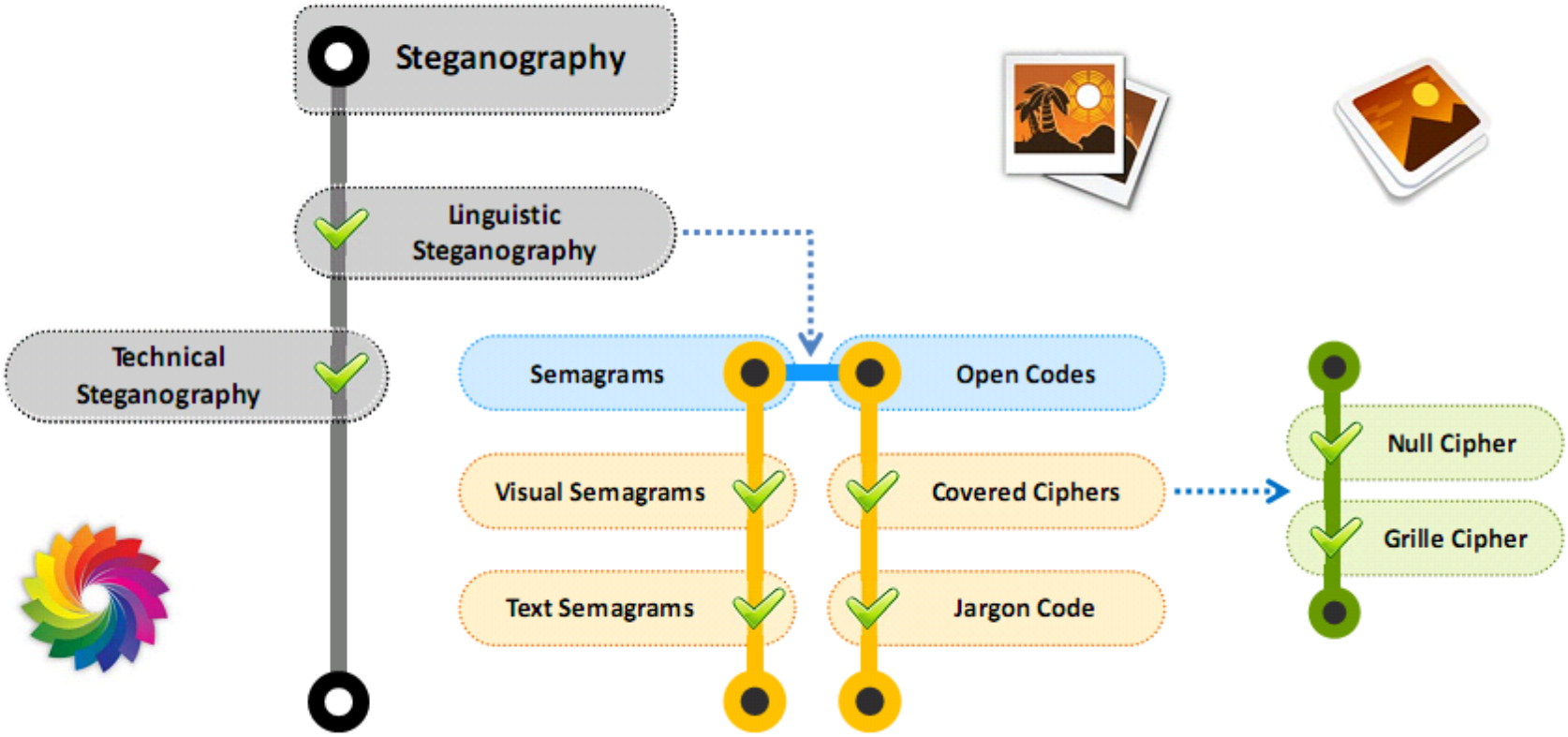
01 Steganography is a technique of **hiding a secret message** within an ordinary message and **extracting it at the destination** to maintain confidentiality of data

02 **Utilizing a graphic image as a cover** is the most popular method to conceal the data in files

03 Attacker can use steganography to hide messages such as **list of the compromised servers**, source code for the hacking tool, plans for future attacks, etc.



# Classification of Steganography



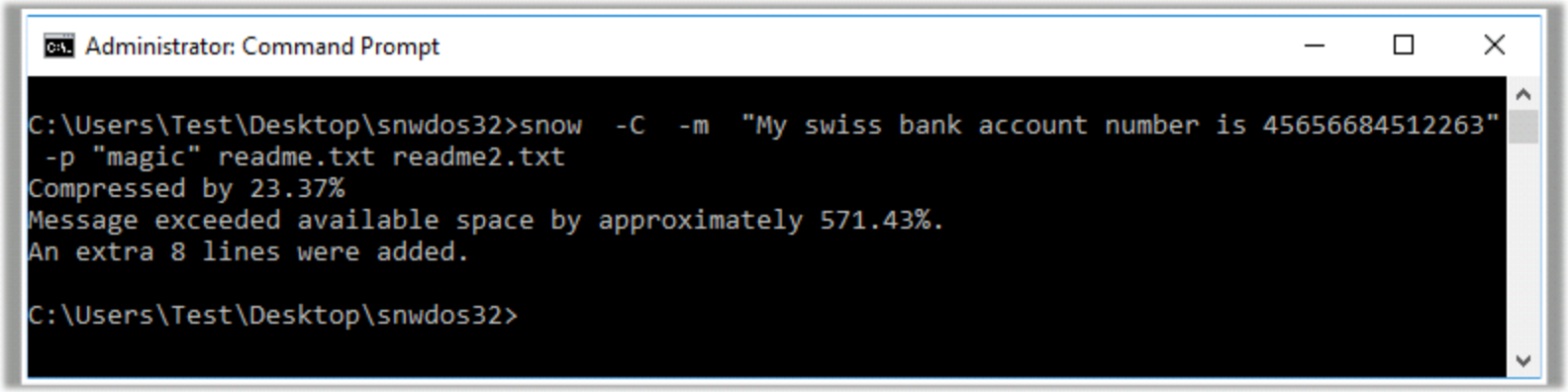
# Types of Steganography based on Cover Medium



| | | | |
|---|---|---|---|
| 01 | Image Steganography | 07 | Web Steganography |
| 02 | Document Steganography | 08 | Spam/Email Steganography |
| 03 | Folder Steganography | 09 | DVD-ROM Steganography |
| 04 | Video Steganography | 10 | Natural Text Steganography |
| 05 | Audio Steganography | 11 | Hidden OS Steganography |
| 06 | White Space Steganography | 12 | Source Code Steganography |

# Whitespace Steganography

- In white space steganography, user **hides the messages in ASCII text** by adding white spaces to the end of the lines
- Because spaces and tabs are generally not visible in **text viewers**, therefore the message is effectively hidden from casual observers
- Use of **built-in encryption** makes the message unreadable even if it is detected
- Use **SNOW** tool to hide the message




http://www.darkside.com.au

# Image Steganography

- In image steganography, the **information is hidden in image** files of different formats such as .PNG, .JPG, .BMP, etc.
- Image steganography tools **replace redundant bits of image** data with the message in such a way that the effect cannot be detected by human eyes

## Image File Steganography Techniques

### Least Significant Bit Insertion

- The binary data of the message is broken and inserted into the **LSB of each pixel** in the image file in a deterministic sequence

### Masking and Filtering

- Masking and filtering techniques **hide data using a method similar to watermarks on actual paper** and it can be done by modifying the luminance of parts of the image

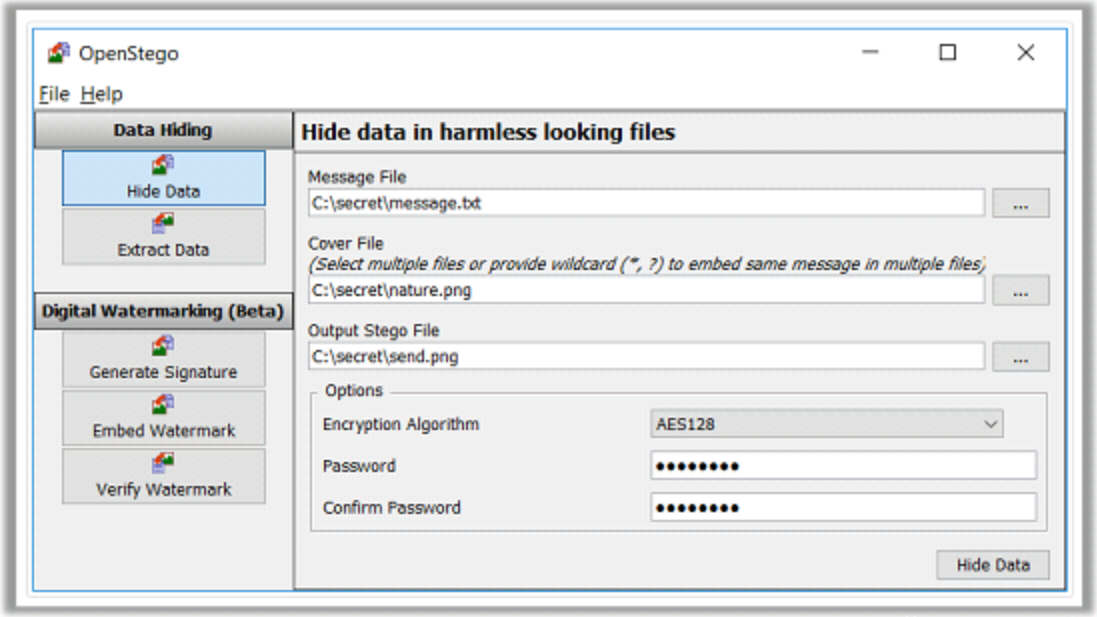
### Algorithms and Transformation

- Hide data in **mathematical functions** used in compression algorithms
- The data is embedded in the cover image by **changing the coefficients of a transform of an image**

# Image Steganography Tools

## OpenStego

- **Data Hiding**: It can hide any data within a cover file (e.g. images)
- **Watermarking**: Watermarking files (e.g. images) with an invisible signature. It can be used to detect unauthorized file copying




https://www.openstego.info

**QuickStego**
*http://quickcrypto.com*

**CryptaPix**
*https://www.briggsoft.com*

**Hide In Picture**
*https://sourceforge.net*

**gifshuffle**
*http://www.darkside.com.au*

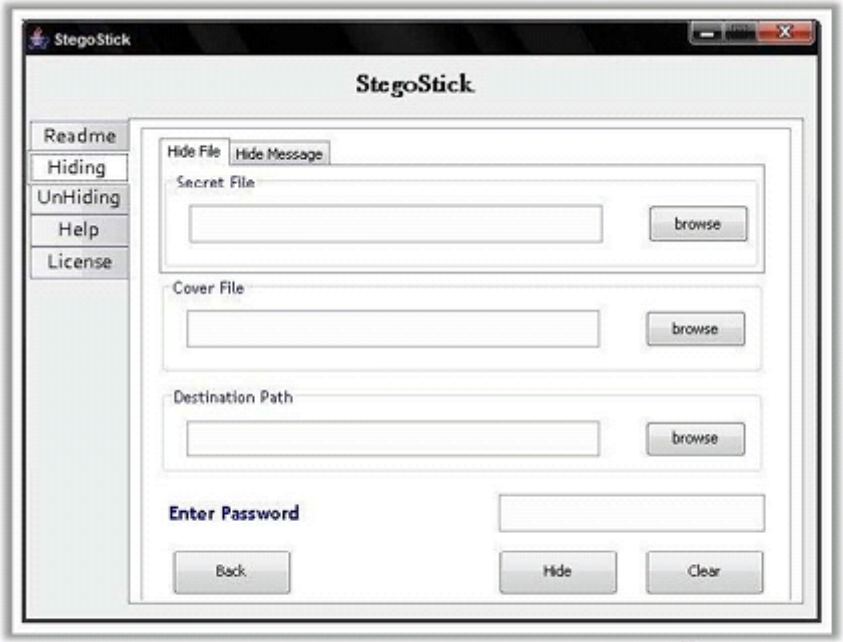
**PHP-Class Stream Steganography**
*https://www.phpclasses.org*

# Document Steganography

- Document steganography is the technique of **hiding secret messages** transferred in the **form of documents**
- It includes **addition of white spaces and tabs** at the end of the lines

## StegoStick

It hides any file or message into an image (BMP,JPG,GIF), Audio/Video (MPG, WAV, etc.) or any other file format (PDF,EXE,CHM, etc.)


https://sourceforge.net

## Document Steganography Tools

- StegJ (*http://stegj.sourceforge.net*)
- Office XML (*https://www.irongeek.com*)
- SNOW (*http://www.darkside.com.au*)
- Data Stash (*http://www.skyjuicesoftware.com*)
- Hydan (*http://www.crazyboy.com*)

# Video Steganography

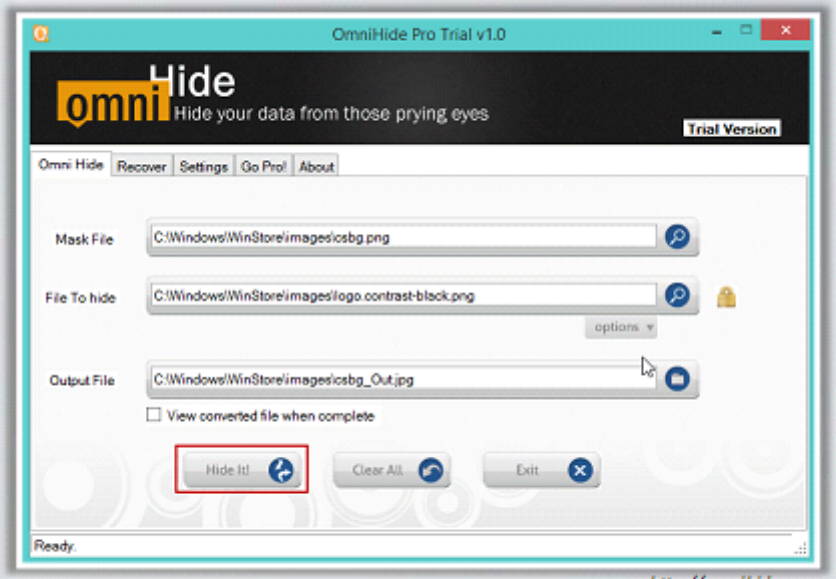
- Video steganography refers to **hiding secret information** into a carrier video file
- In video steganography, the information is hidden in **video files** of different formats such as .AVI, .MPG4, .WMV, etc.
- **Discrete Cosine Transform (DCT)** manipulation is used to add secret data at the time of the transformation process of video

## Video Steganography Tools

- RT Steganography (*https://rtstegvideo.sourceforge.net*)
- StegoStick (*https://sourceforge.net*)
- OpenPuff (*http://embeddedsw.net*)
- MSU StegoVideo (*http://www.compression.ru*)

## OmniHide Pro

OmniHide Pro **hides a file within another file**. Any file can be hidden within common image/music/ video/document formats. The output file would work just as the original source file




*http://omnihide.com*

# Audio Steganography

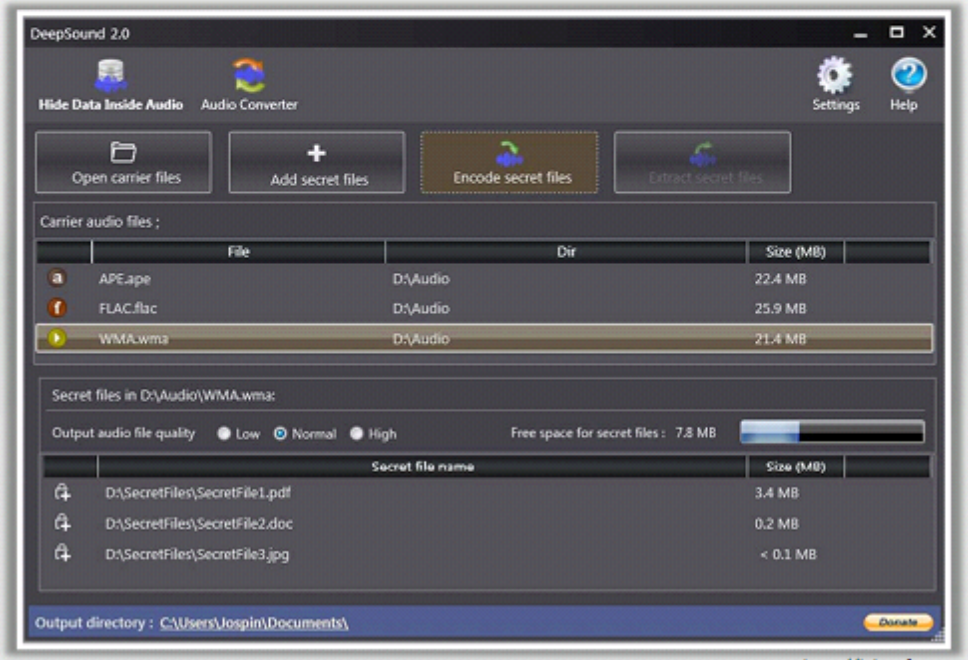
- Audio steganography refers to **hiding secret information in audio files** such as .MP3, .RM, .WAV, etc.
- Information can be hidden in an audio file by using **LSB** or by using **frequencies** that are inaudible to the human ear (>20,000 Hz)
- Some of the audio steganography methods are **echo data hiding**, **spread spectrum method**, **LSB coding**, **tone insertion**, **phase encoding**, etc.

## Audio Steganography Tools

- BitCrypt (*http://bitcrypt.moshe-szweizer.com*)
- SilentEye (*http://silenteye.v1kings.io*)
- CHAOS Universal (*http://safechaos.com*)
- StegoStick (*https://sourceforge.net*)
- MP3Stego (*http://www.petitcolas.net*)

## DeepSound

- DeepSound hides secret data into **audio files** - **wave and flac**
- It enables extracting secret files directly from **audio CD tracks**




*http://jpinsoft.net*

# Folder Steganography

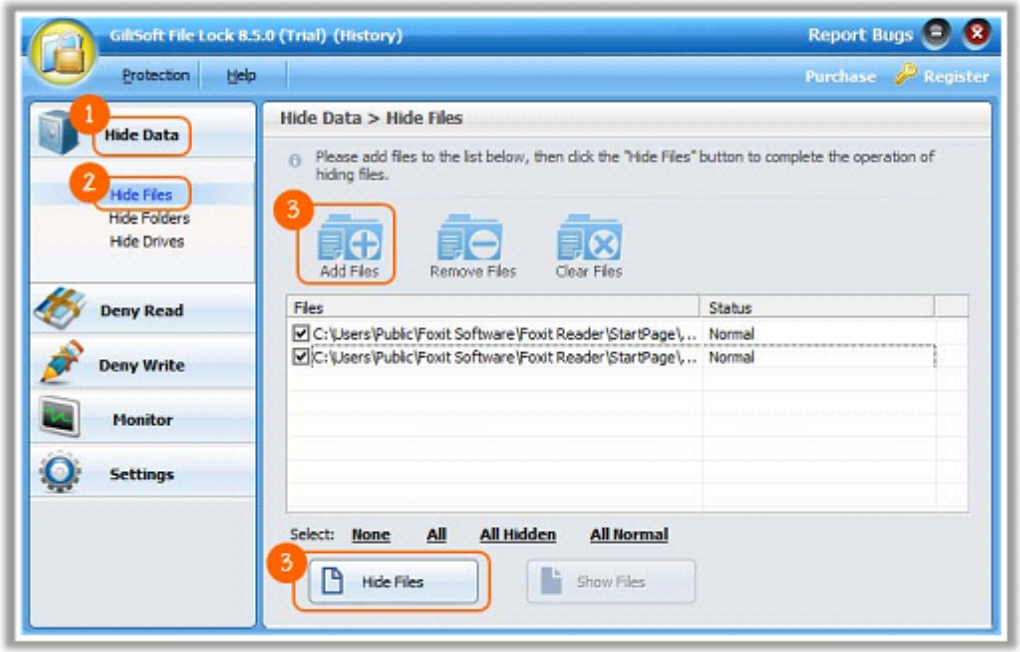
- In Folder steganography, **Files are hidden and encrypted** within a folder and do not appear to normal Windows applications, including Windows Explorer

## Folder Steganography Tools

- Folder Lock *(http://www.newsoftwares.net)*
- Hide Folders 5 *(https://fspro.net)*
- WinMend Folder Hidden *(http://www.winmend.com)*
- Invisible Secrets 4 *(http://www.invisiblesecrets.com)*
- Max Folder Secure *(http://maxpcsecure.com)*

## GiliSoft File Lock Pro

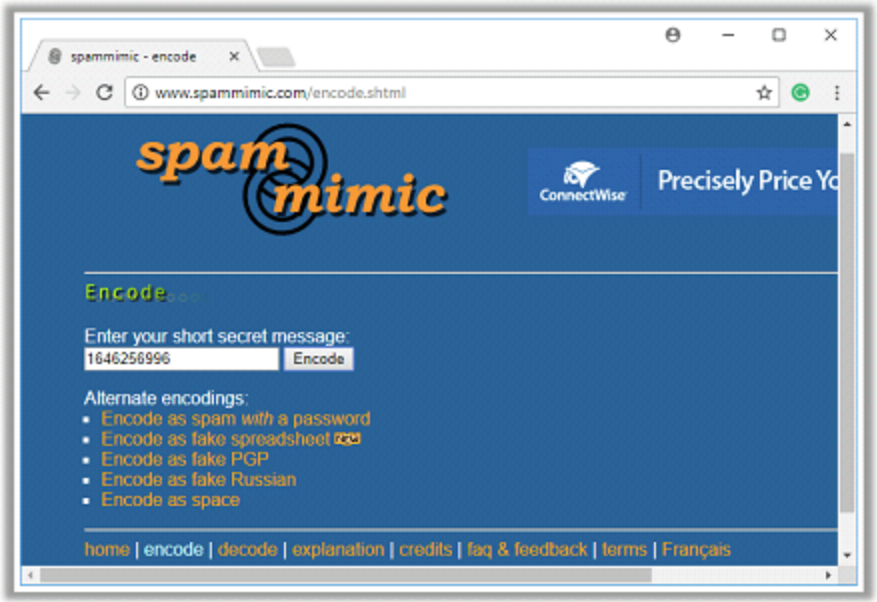
It lock files, folders, and drives; hide files, folders, and drives to make them invisible; or password protects files folders, and drives




*http://www.gilisoft.com*

# Spam/Email Steganography

- Spam/email steganography refers to the technique of **sending secret messages by hiding them in spam/email messages**
- Spam emails help to **communicate secretly** by embedding the secret messages in some way and hiding the embedded data in the spam emails
- **Spam Mimic** is a spam/email steganography tool that encodes the secret message into an innocent looking spam emails

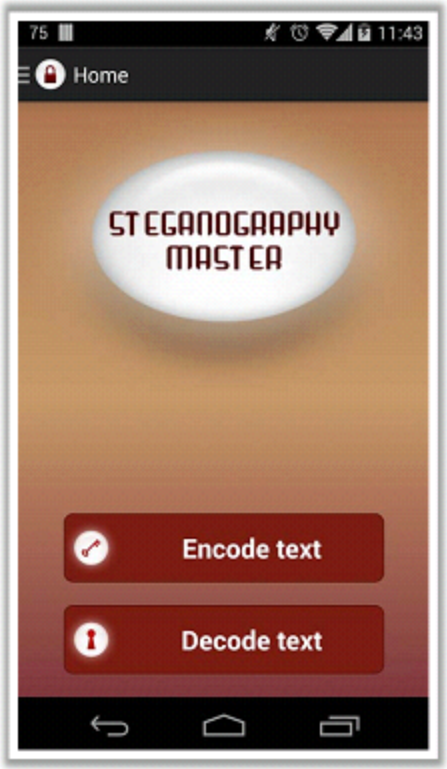




*http://www.spammimic.com*

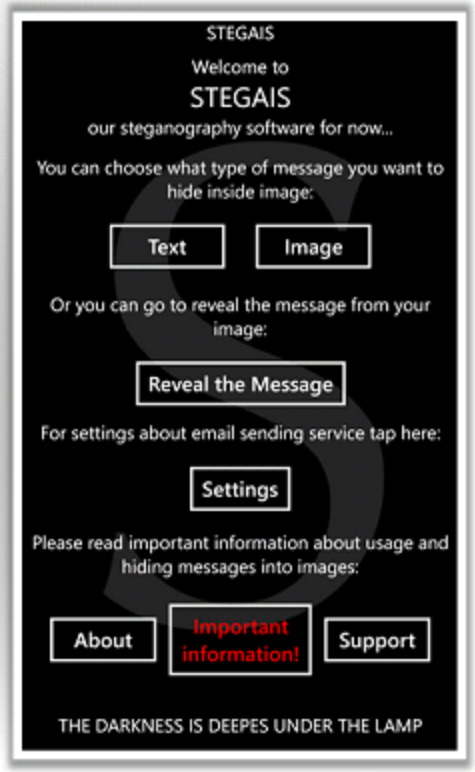
# Steganography Tools for Mobile Phones

## Steganography Master




https://play.google.com

## Stegais




http://stegais.com

**SPY PIX**
*https://www.juicybitssoftware.com*

**Pixelknot: Hidden Messages**
*https://guardianproject.info*

**Pocket Stego**
*http://www.talixa.com*

**Steganography Image**
*https://play.google.com*

**StegoSec**
*http://csocks.altervista.org*

# Steganalysis



## Reverse Process of Steganography

- Steganalysis is the art of **discovering** and **rendering covert messages** using steganography
- It **detects the hidden messages** embedded in images, text, audio, and video carrier mediums using steganography

## Challenge of Steganalysis

- Suspect information stream may or may not have encoded hidden data
- Efficient and accurate detection of hidden content within digital images is difficult
- The message might have been encrypted before inserting into a file or signal
- Some of the suspect signals or files may have irrelevant data or noise encoded into them

# Steganalysis Methods/Attacks on Steganography

| | |
|---|---|
| **Stego-only** | Only the stego object is available for analysis |
| **Known-stego** | Attacker has access to the stego algorithm and both the cover medium and the stego-object |
| **Known-message** | Attacker has access to the hidden message and the stego object |
| **Known-cover** | Attacker compares the stego-object and the cover medium to identify the hidden message |
| **Chosen-message** | This attack generates stego objects from a known message using specific steganography tools in order to identify the steganography algorithms |
| **Chosen-stego** | Attacker has access to the stego-object and stego algorithm |

# Detecting Steganography
## (Text, Image, Audio, and Video Files)



**Text File**

- For the text files, the alterations are made to the **character positions** for hiding the data
- The alterations are detected by looking for **text patterns** or disturbances, language used, and an unusual amount of blank spaces

**Image File**

- The hidden data in an image can be detected by **determining changes** in size, file format, the last modified timestamp, and the color palette pointing to the existence of the hidden data
- **Statistical analysis** method is used for image scanning

**Audio File**

- Statistical analysis method can be used for detecting audio steganography as it involves **LSB modifications**
- The **inaudible frequencies** can be scanned for hidden information
- The **odd distortions and patterns** show the existence of the secret data

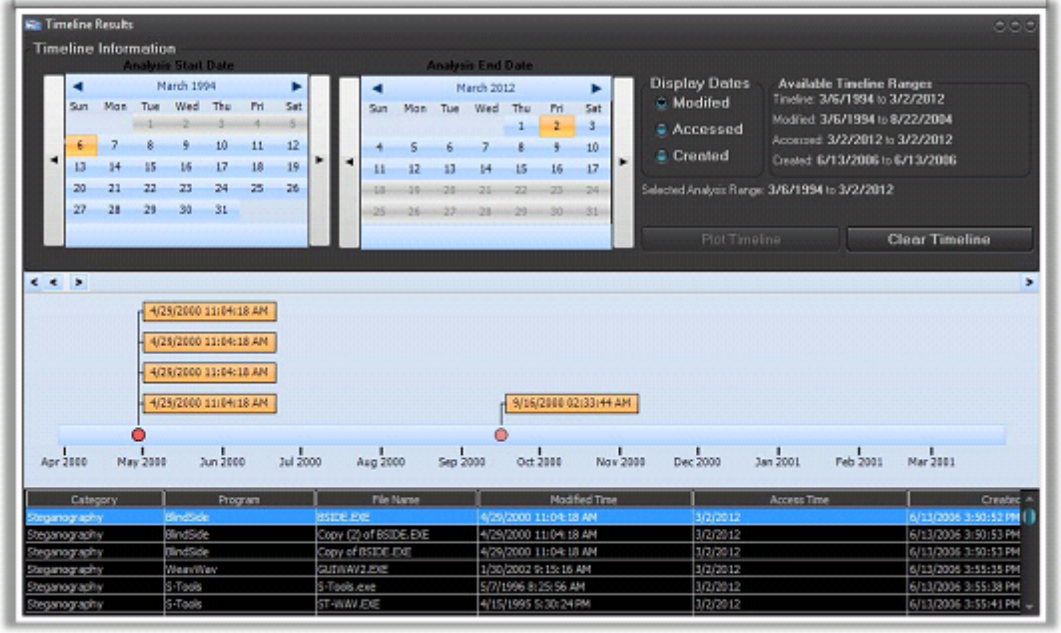
**Video File**

- Detection of the secret data in video files includes a **combination of methods** used in image and audio files

# Steganography Detection Tools

**Gargoyle Investigator™ Forensic Pro**

- Gargoyle Investigator™ Forensic Pro provides inspectors with the ability to conduct a quick search on a given computer or machine for known contraband and hostile programs




https://www.wetstonetech.com

**StegAlyzerSS**
*http://www.sarc-wv.com*

**Steganography Studio**
*http://stegstudio.sourceforge.net*

**StegAlyzerAS**
*http://www.sarc-wv.com*

**StegAlyzerRTS**
*http://www.sarc-wv.com*

**Virtual Steganographic Laboratory (VSL)**
*http://vsl.sourceforge.net*

# Module Flow

1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks
7. Penetration Testing

# Covering Tracks



- Once intruders have successfully gained administrator access on a system, they will try to **cover the tracks to avoid their detection**


**Attacker uses the following techniques to cover tracks on the target system**

1. **Disable Auditing**: Disables auditing features of the target system
2. **Clearing Logs**: Clear/delete the system log entries corresponding to his/her activities
3. **Manipulating Logs**: Manipulates logs in such a way that he/she will not be caught in legal actions

# Disabling Auditing: Auditpol

- Intruders will **disable auditing** immediately after gaining administrator privileges
- At the end of their stay, the intruders will just turn on auditing again using **auditpol.exe**

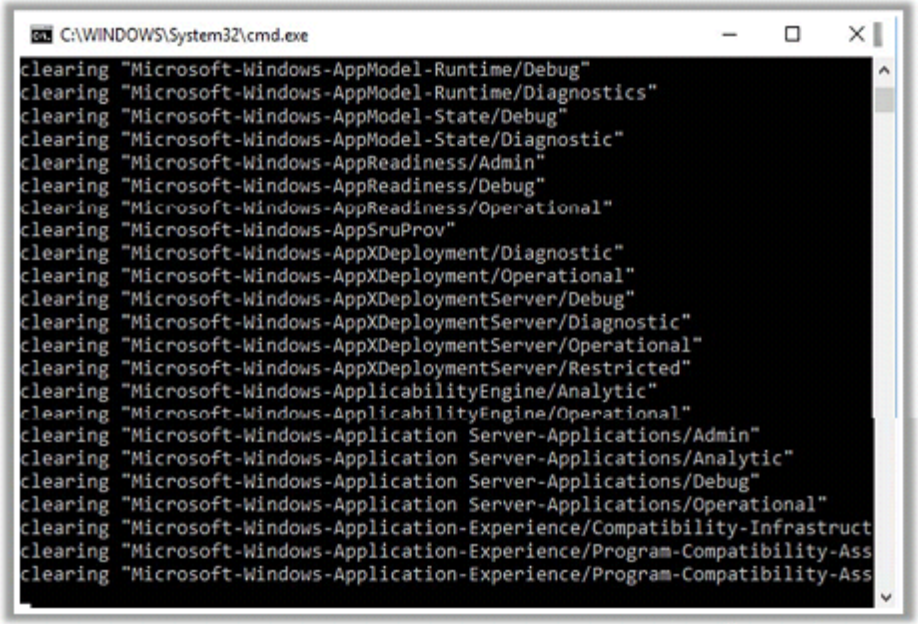
Administrator: Command Prompt

```
C:\Users\Administrator>auditpol /set /category:"system","account logon" /success:
enable /failure:enable
The command was successfully executed.
C:\Users\Administrator>auditpol /get /category:*
System audit policy
Category/Subcategory                    Setting
System
  Security System Extension             Success and Failure
  System Integrity                      Success and Failure
  IPsec Driver                          Success and Failure
  Other System Events                   Success and Failure
  Security State Change                 Success and Failure
Logon/Logoff
  Logon                                 Success and Failure
  Logoff                                Success
  Account Lockout                       Success
  IPsec Main Mode                       No Auditing
  IPsec Quick Mode                      No Auditing
  IPsec Extended Mode                   No Auditing
  Special Logon                         Success
  Other Logon/Logoff Events             No Auditing
  Network Policy Server                 Success and Failure
  User / Device Claims                  No Auditing
  Group Membership                      No Auditing
Object Access
  File System                           No Auditing
  Registry                              No Auditing
  Kernel Object                         No Auditing
  SAM                                   No Auditing
  Certification Services                No Auditing
  Application Generated                 No Auditing
  Handle Manipulation                   No Auditing
```

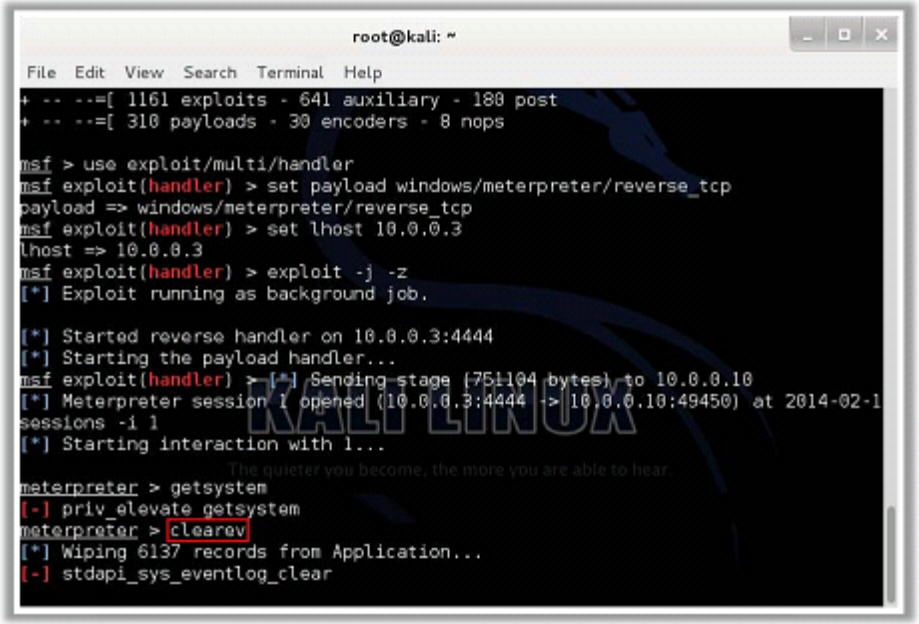
*https://technet.microsoft.com*

# Clearing Logs

Attacker uses **Clear_Event_Viewer_Logs.bat** or **clearlogs.exe** utility to clear the security, system, and application logs




If the system is exploited with the Metasploit, attacker uses **meterpreter shell** to wipe out all the logs from a Windows system

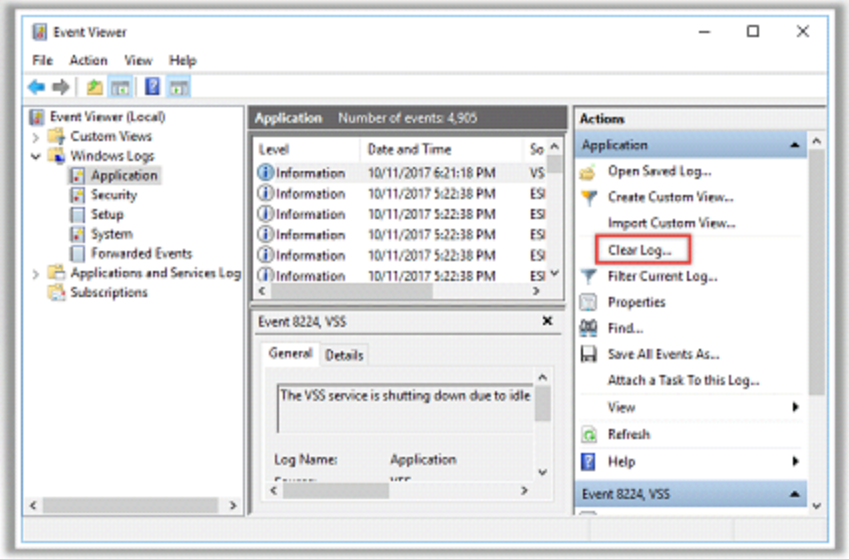



https://www.tenforums.com

# Manually Clearing Event Logs

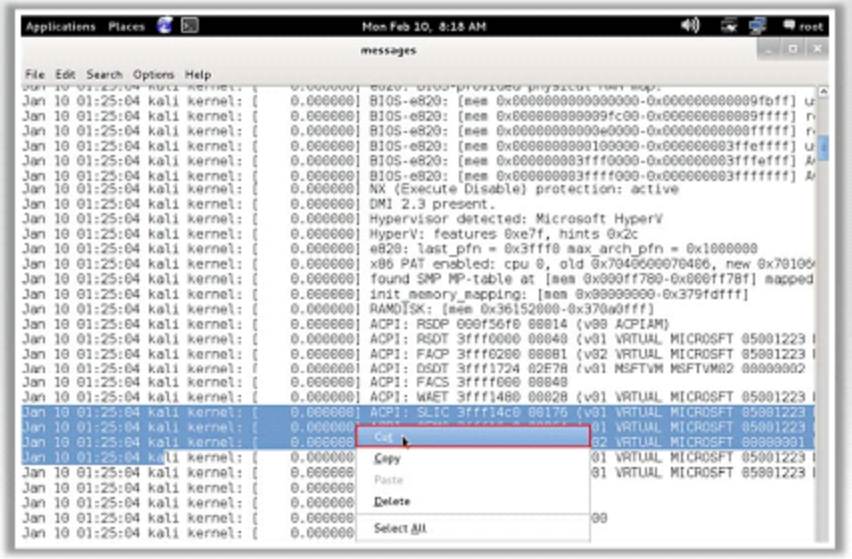
## Windows

- Navigate to **Start** → **Control Panel** → **System and Security** → **Administrative Tools** → double click **Event Viewer**
- Delete the all the log entries logged while compromising of the system




## Linux

- Navigates to `/var/log` directory on the Linux system
- Open plain text file containing log messages with text editor `/var/log/messages`
- Delete all the log entries logged while compromising of the system



# Ways to Clear Online Tracks



- Remove **Most Recently Used (MRU)**, delete cookies, clear cache, turn off AutoComplete, and clear Toolbar data from the browsers

### From the Privacy Settings in Windows 10

- Right-click on the **Start** button, choose **Settings**, and click on "**Personalization**"
- In Personalization, click **Start** from the left pane and Turn Off both "**Show most used apps**" and "**Show recently opened items in Jump Lists on Start or the taskbar**"

### From the Registry in Windows 10

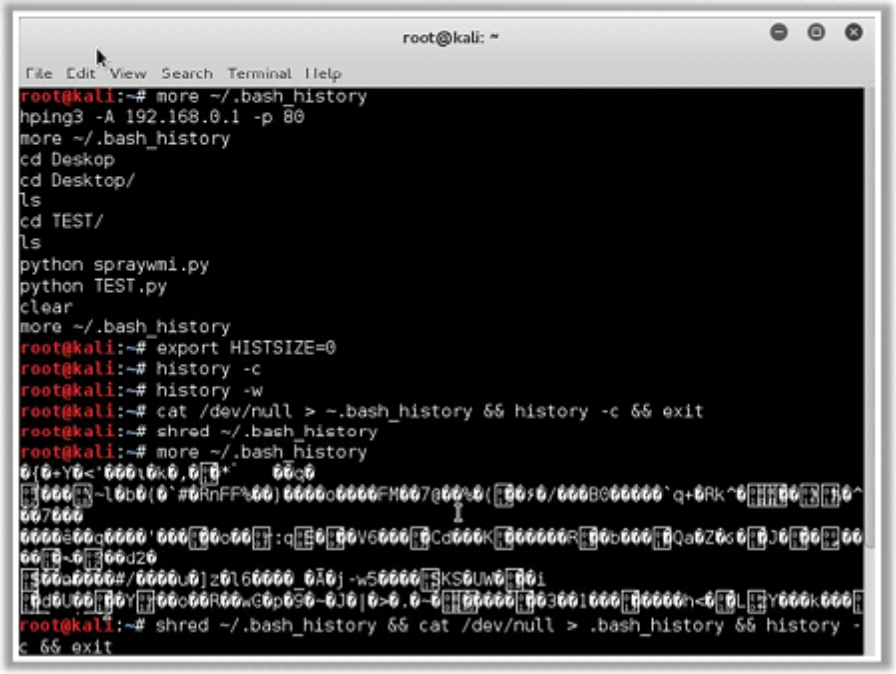
- Open the **Registry Editor** and navigate to **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer** and then remove the key for "Recent Docs"
- Delete all the values except "**(Default)**"

# Covering BASH Shell Tracks

- The BASH is an **sh-compatible shell** which stores command history in a file called **bash_history**
- You can view the saved command history using **more ~/.bash_history** command

**Attackers use following commands to clear the saved command history tracks:**

- **Disabling history**
  - `export HISTSIZE=0`
- **Clearing the history**
  - `history -c` (Clears the stored history)
  - `history -w` (Clears history of current shell)
- **Clearing the user's complete history**
  - `cat /dev/null > ~.bash_history && history -c && exit`
- **Shredding the history**
  - `shred ~/.bash_history` (Shreds the history file, making its content unreadable)
  - `shred ~/.bash_history && cat /dev/null > .bash_history && history -c && exit` (Shreds the history file and clear the evidence of the command)



# Covering Tracks on Network



## Using Reverse HTTP Shells

- Attacker **installs reverse HTTP shell** on victim's machine, which is programmed in such a way that it would ask for commands to an **external master** who controls the reverse HTTP shell
- Victim here will act as a web client who is executing **HTTP GET commands** whereas the attacker behaves like a web server and responds to the requests
- This type of traffic is considered as a **normal traffic** by an organization's network perimeter security like DMZ, firewall, etc.

## Using Reverse ICMP Tunnels

- Attacker uses ICMP tunneling technique to use **ICMP echo** and **ICMP reply** packets as a carrier of TCP payload, to access or control a system stealthily
- Victim's system is triggered to encapsulate **TCP payload** in an ICMP echo packet which is forwarded to the proxy server
- Organizations have security mechanisms that only check incoming ICMP packets but not outgoing ICMP packets, therefore attackers can easily **bypass firewall**

# Covering Tracks on Network (Cont'd)



## Using DNS Tunneling

- Attackers can use DNS tunneling to **encode malicious content** or data of other programs within DNS queries and replies
- DNS tunneling **creates a back channel** to access a remote server and applications
- Attackers can make use of this back channel to **exfiltrate stolen confidential** or sensitive information from the server

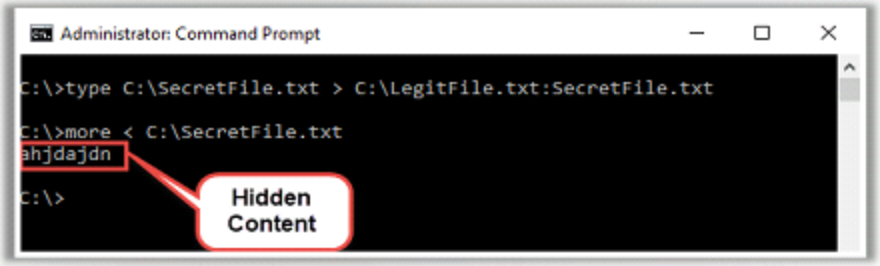
## Using TCP Parameters

- TCP parameters can be used by the attacker to **distribute the payload** and to create **covert channels**
- TCP fields where data can be hidden are as follow:
  - IP Identification field
  - TCP acknowledgement number
  - TCP initial sequence number

# Covering Tracks on OS

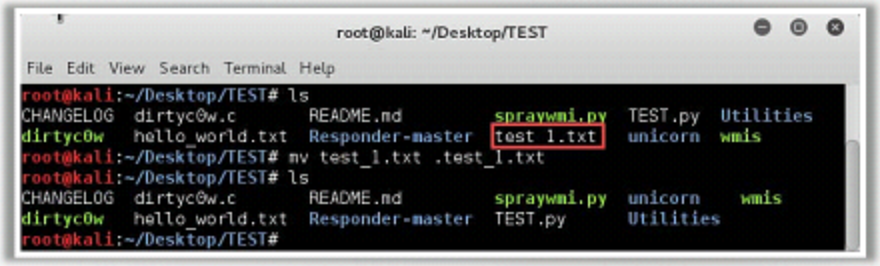
## Windows

- NTFS has a feature called as **Alternate Data Streams** that allows attackers to hide a file behind other normal files
- Given below are some steps in order to hide file using NTFS:
  - Open the command prompt with an elevated privilege
  - Type the command "**`type C:\SecretFile.txt > C:\LegitFile.txt:SecretFile.txt`**" (here, file is kept in C drive where SecretFile.txt file is hidden inside LegitFile.txt file)
  - To view the hidden file, type "**`more < C:\SecretFile.txt`**" (for this you need to know the hidden file name)




## UNIX

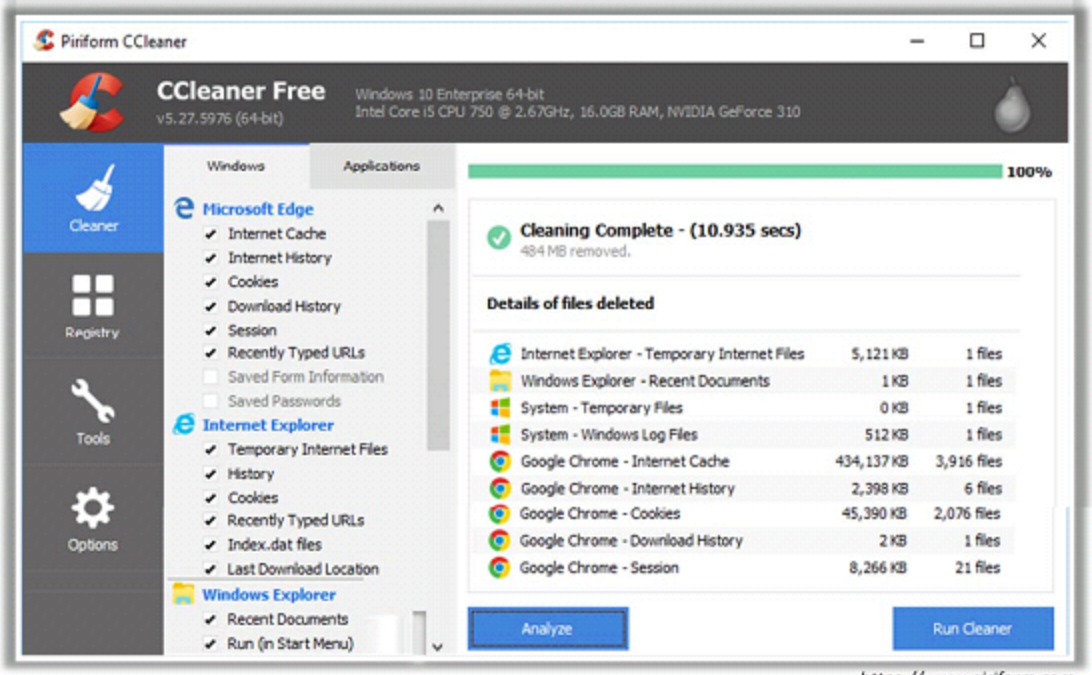
- Files in UNIX can be hidden just by **appending a dot** (.) in front of a file name
- Attackers can use this feature to edit the **log files** to cover their tracks
- Attackers can use "**`export HISTSIZE=0`**" command to delete the command history and the specific command they used to hide log files



# Covering Tracks Tools

## CCleaner

- CCleaner cleans traces of temporary files, log files, registry files, memory dumps, and also your online activities such as your Internet history




*https://www.piriform.com*

**DBAN**
*http://www.cybertronsoft.com*

**Privacy Eraser**
*http://www.cybertronsoft.com*

**Wipe**
*https://privacyroot.com*

**BleachBit**
*https://www.bleachbit.org*

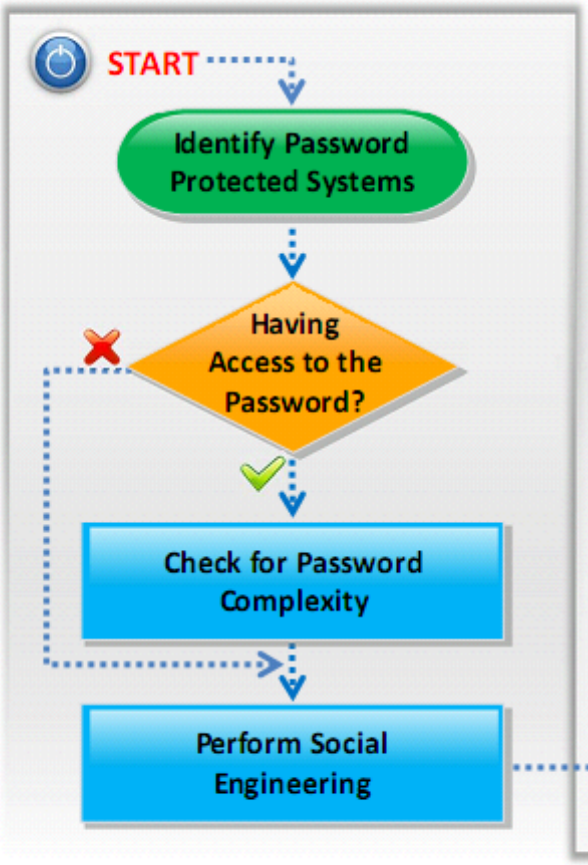
**ClearProg**
*http://www.clearprog.de*

# Module Flow

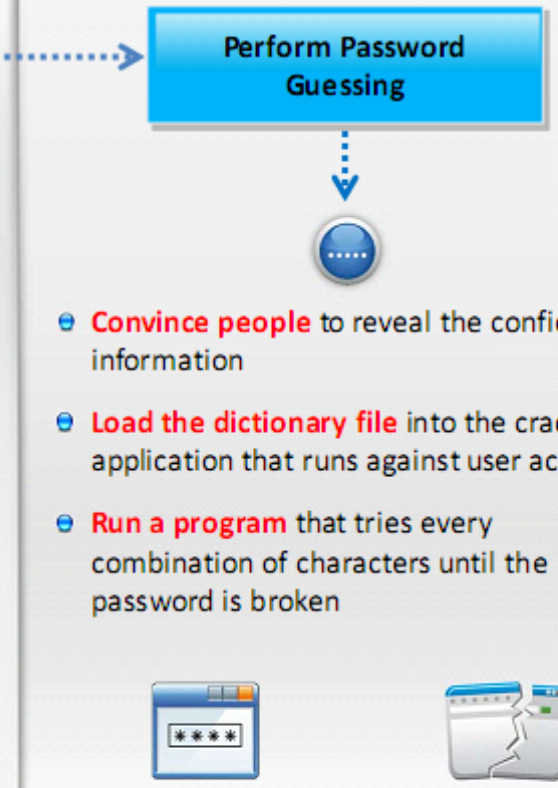
1. System Hacking Concepts
2. Cracking Passwords
3. Escalating Privileges
4. Executing Applications
5. Hiding Files
6. Covering Tracks

7. Penetration Testing

# Password Cracking

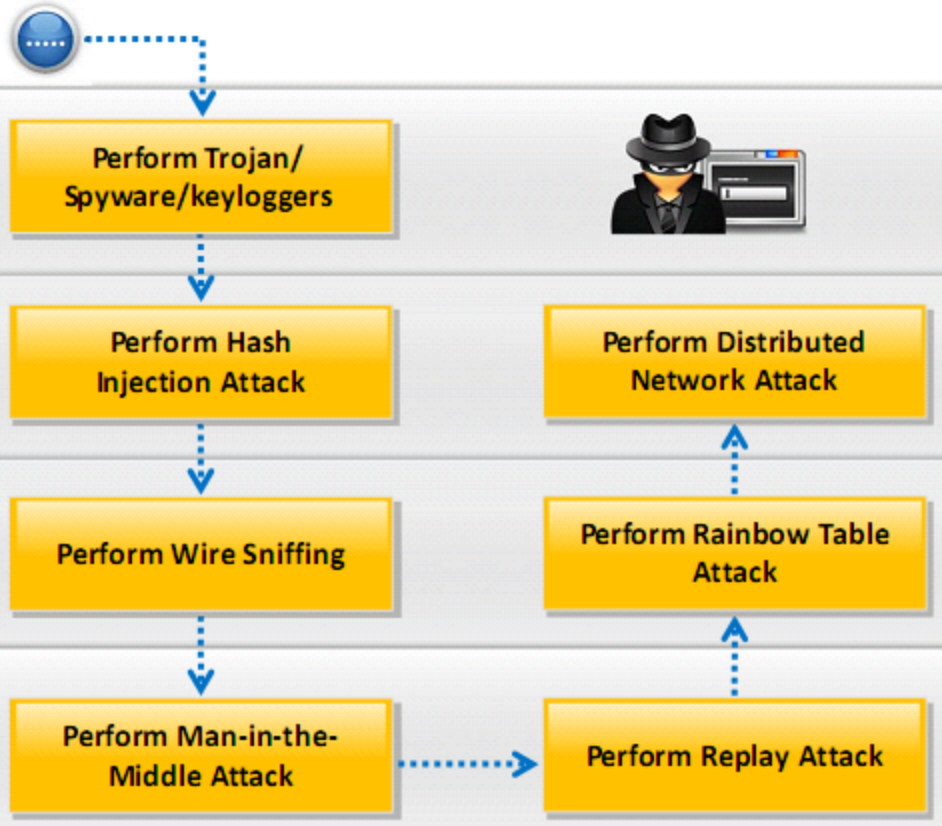






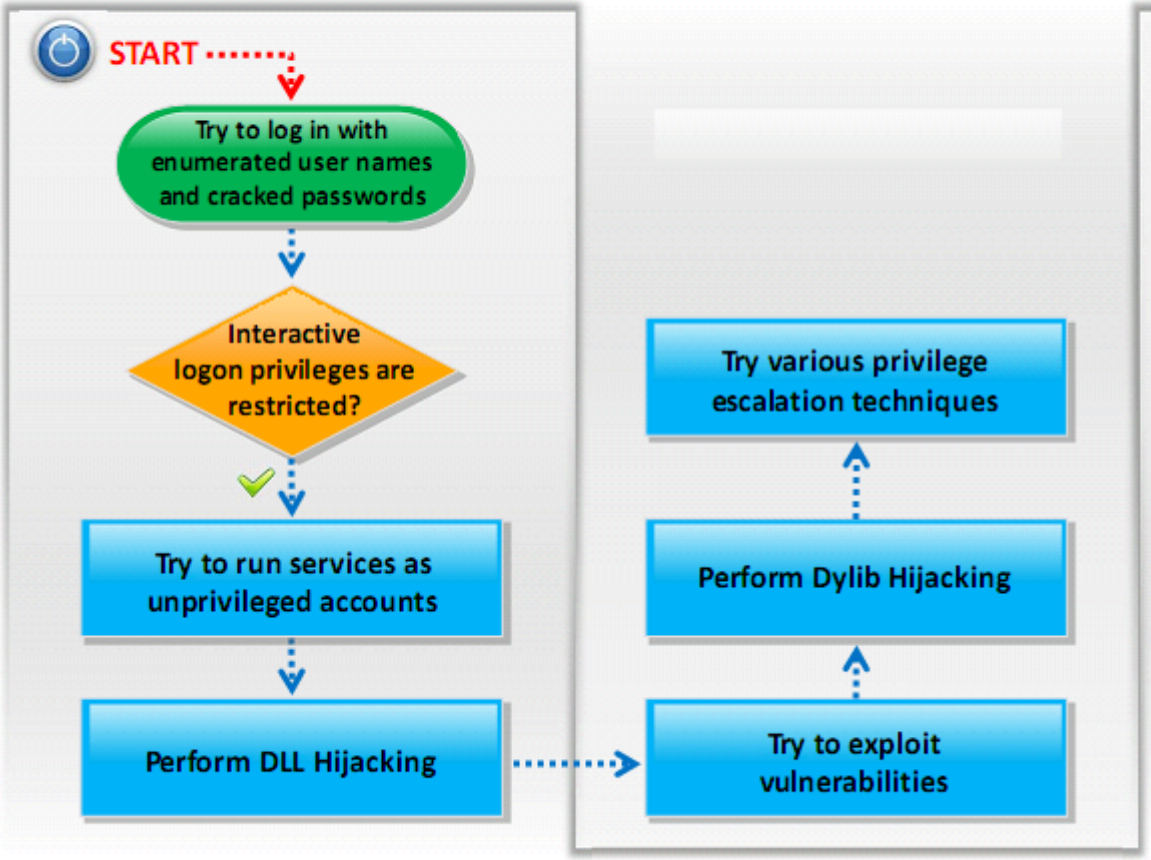
# Password Cracking (Cont'd)





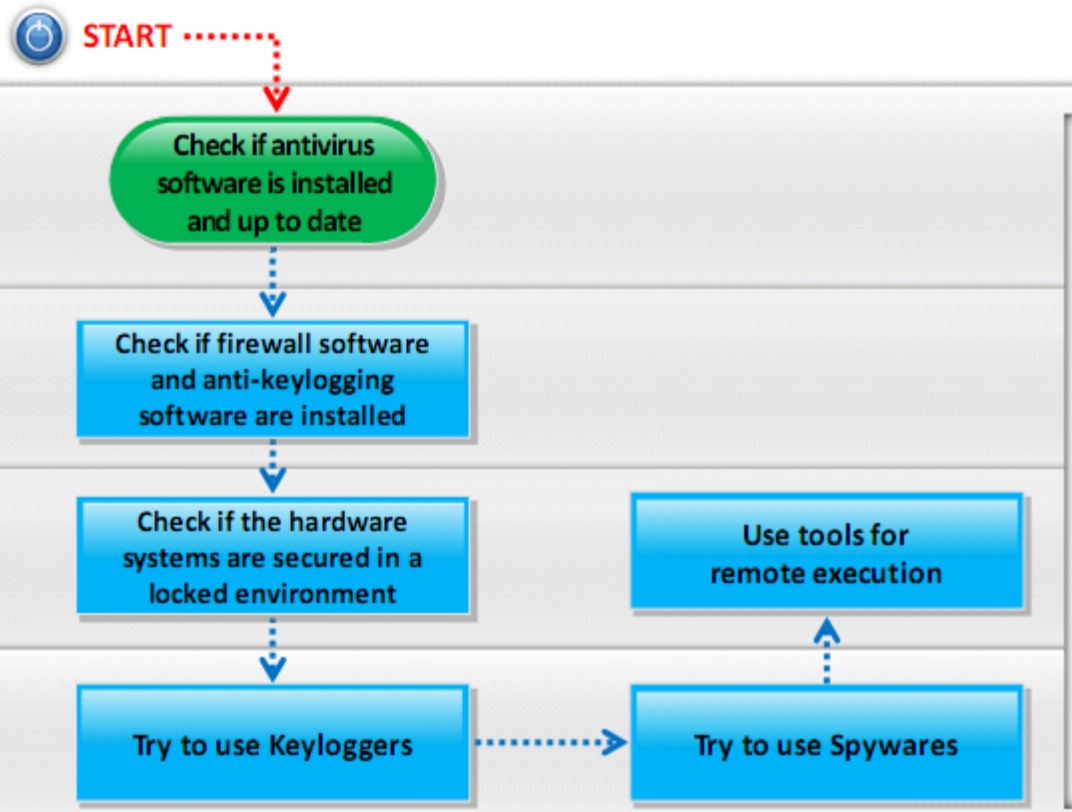

- **Record every keystroke** that an user types using keyloggers
- Secretly gather person or organization personal information using **spyware**
- With the help of a **Trojan**, get access to the stored passwords in the Trojaned computer
- **Inject a compromised hash** into a local session and use the hash to validate to network resources
- Run **packet sniffer tools** on the LAN to access and record the raw network traffic that may include passwords sent to remote systems
- Acquires access to the **communication channels between victim and server** to extract the information
- Use a **Sniffer** to capture packets and authentication tokens. After extracting relevant info, place back the tokens on the network to gain access
- Recover password-protected files using the unused processing power of **machines across the network** to decrypt password

# Privilege Escalation



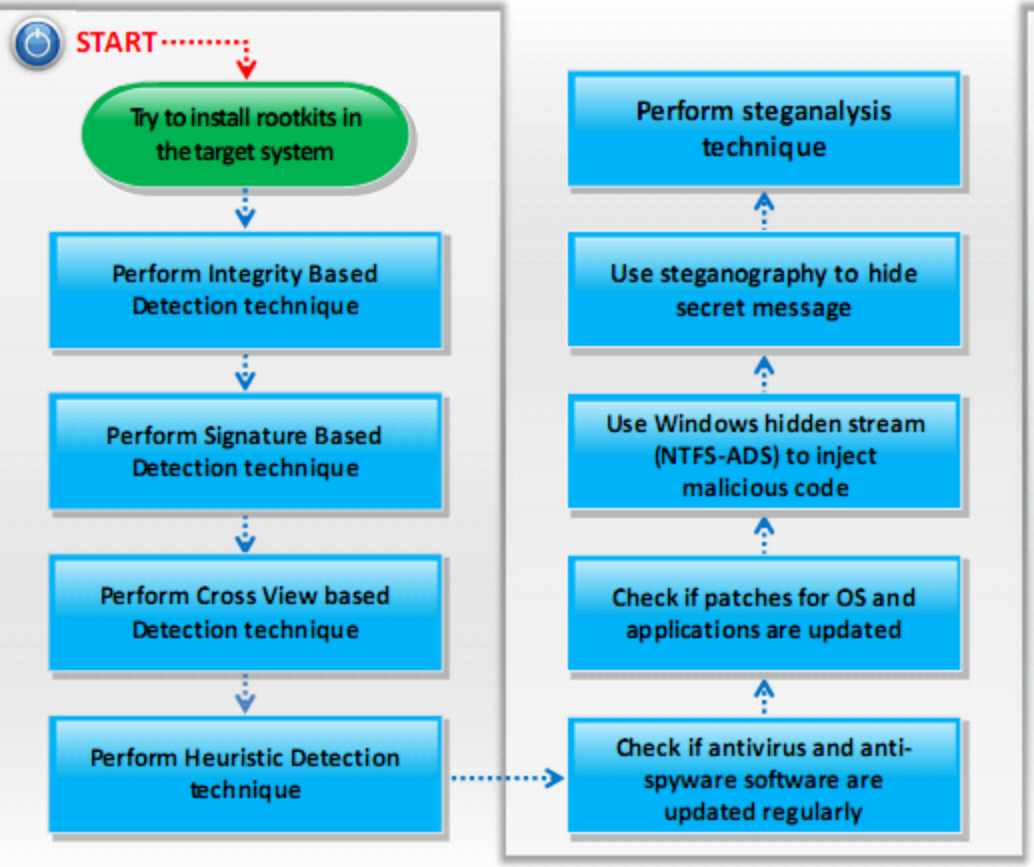

- Place a **malicious DLL in the application directory** to get executed in place of the real DLL
- Search for an exploit based on the OS and software application on exploit sites such as **SecurityFocus**, **Exploit Database**, etc.
- **Inject a malicious dylib** in one of the primary directories to get executed in place of the original dylib
- Try various privilege escalation techniques such as **access token manipulation**, **application shimming**, Path Interception, Plist Modification, Setuid and Setgid, Web Shell, etc.

# Executing Applications



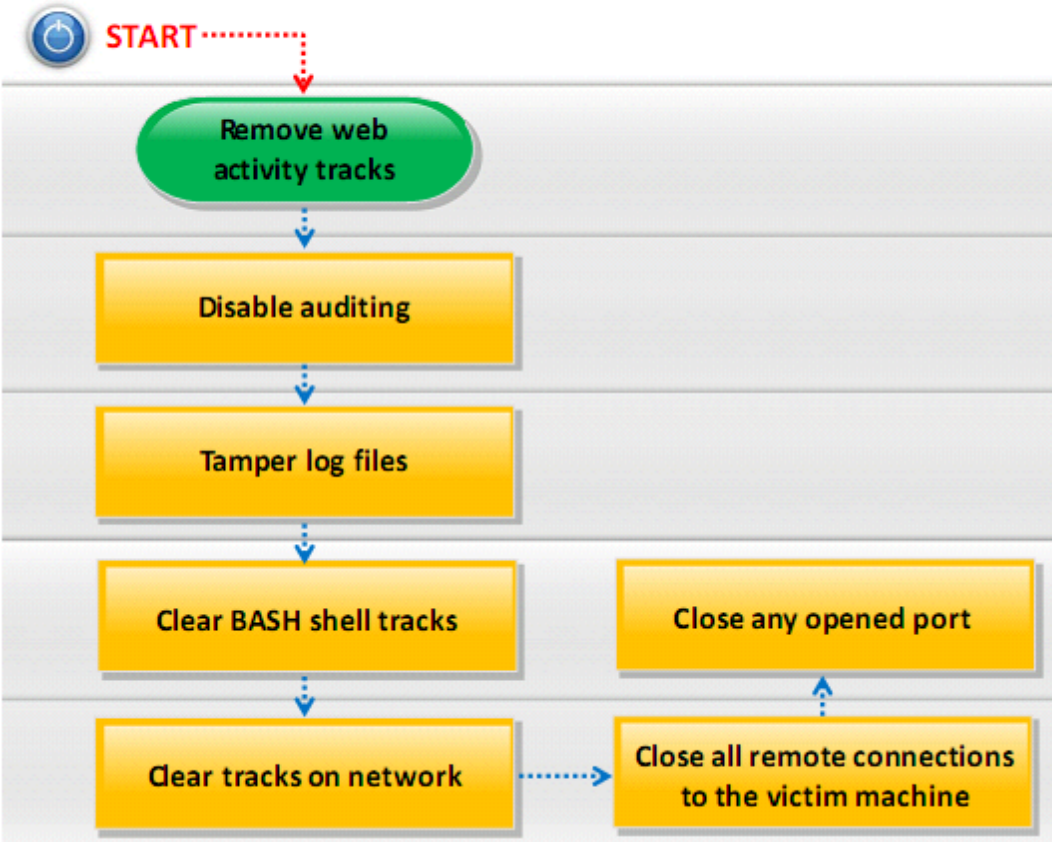

- Use **keyloggers** such as All In One Keylogger, Spyrix Personal Monitor, SoftActivity Activity Monitor, Elite Keylogger, etc.
- Use **spywares** such as Spytech SpyAgent, Power Spy, ACTIVTrak, Veriato 360, NetVizor, Activity Monitor, etc.
- Use remote execution tools such as RemoteExe, PDQ Deploy, Dameware Remote Support, etc. to install application remotely

# Hiding Files




- Try to install the rootkit in the target system to **maintain hidden access**
- Perform Integrity Based Detection, Signature Based Detection, Cross View Based Detection, and Heuristic Detection techniques to **detect rootkits**
- Use **anti-rootkits** such as Stinger, Avast Rootkit, TDSSKiller, Malwarebytes Anti-Rootkit, Rootkit Buster, etc. to detect rootkits
- Use NTFS Alternate Data Stream (ADS) to **inject malicious code** on a breached system and execute them without being detected by the user
- Use **NTFS stream detectors** such as StreamArmor, Stream Detector, Forensic Toolkit, ADS Manager, etc. to detect NTFS-ADS stream
- Use steganography technique **to hide secret message** within an ordinary message and extract it at the destination to maintain confidentiality of data
- Use **steganography detection tools** such as Gargoyle Investigator™ Forensic Pro, StegAlyzerSS, Steganography Studio, StegAlyzerAS, etc. to perform steganalysis

# Covering Tracks




- Remove **web activity tracks** such as MRU, cookies, cache, temporary files and history using **Clear_Event_Viewer_Logs.bat** utility and **meterpreter shell**
- Disable auditing using tool such as **Auditpol**
- Tamper log files such as event log files, server log files and proxy log files by **log poisoning or log flooding**
- Use command like `history -c`, `cat /dev/null > ~.bash_history && history -c && exit`, etc. to clear BASH shell tracks
- Cover tracks on network using Reverse HTTP Shells, Reverse ICMP Tunnels, and TCP Parameters
- Use **track covering tools** such as CCleaner, AVG TuneUp, Privacy Eraser, Wipe, etc.

# Module Summary

- Attackers use a variety of means to penetrate systems, such as:
  - Uses password cracking techniques to gain unauthorized access to the vulnerable system
  - Creates a list (dictionary) of all possible passwords from the information collected through social engineering and perform dictionary, brute force, and rule-based attack on the victim's machine to crack the passwords
  - Performs privilege escalation attack which takes advantage of design flaws, programming errors, bugs, and configuration oversights in the OS and software application to gain administrative access to the network and its associated applications
  - Executes malicious programs remotely in the victim's machine to gather information
  - Uses keystroke loggers and spywares to gather confidential information about victim such as email ID, passwords, banking details, chat room activity, IRC, instant messages, etc.
  - Uses rootkits to hide their presence as well as malicious activities, which grant them full access to the server or host at that time and also in future
  - Uses steganography techniques to hide messages such as list of the compromised servers, source code for the hacking tool, communication and coordination channel, plans for future attacks, etc.
- Once intruders have successfully gained administrator access on a system, they will try to cover the tracks to avoid their detection