BITS Pilani Presentation

BITS Pilani
Pilani Campus

Jagdish Prasad
WILP

innovate achieve lead

BITS Pilani
Pilani Campus

SSZG681: Cyber Security
Lecture No: 03

OS Hardening, Encryption & Rootkits

Agenda

- Hardening OS
- Encryption Technologies
- Rootkit
  - Rootkit & Types
  - How rootkit evades detection
  - Sony XCP rootkit
  - TDSS rootkit
  - Other rootkits
  - Defense against rootkits

Hardening OS

What is Hardening of OS?

- Hardening of the OS is the act of configuring an OS securely, updating it, creating rules and policies to help govern the system in a secure manner, and removing unnecessary applications and services. This is done to minimize a computer OS's exposure to threats and to mitigate possible risk.

- OS hardening refers to adding extra security measures to operating system in order to strengthen it against the risk of cyberattack.

OS Hardening Actions

innovate achieve lead

- **Disable unnecessary features:** Remove unnecessary programs
  - Linux server runs a graphical interface by default but you will only be accessing the system through an SSH client, you should disable (or, better, uninstall completely) the graphical interface.
  - Windows workstation has Skype installed by default but the users will actually be running Skype, disable or uninstall the program.
- **Use of Service Packs:** Keep up-to-date and install the latest version. No one thing ensures protection, especially from zero-day attacks, but this is an easy rule to follow.
- **Patches and patch management:** Planning, testing, implementing and auditing patch management should be part of regular security regimen. Make sure the OS is patched regularly, as well as the individual programs on the client's computer.

OS Hardening Actions

innovate achieve lead

- **Group policies:** Define what groups can or can not access and maintain these rules. Sometimes, it's simply user error that leads to a successful cyber attack. Establish or update user policies and ensure all users are aware and comply with these procedures. For example, everyone should be implementing strong passwords, securing their credentials and changing them regularly.

- **Security templates:** Group of policies which can be loaded into one procedures – commonly used in corporate environment.

- **Configuration baseline:** Baseline the process of measuring changes in networking, hardware, software etc. To create a baseline, select something to measure and measure it consistently for a period of time.

OS Hardening Actions

- **Data and Workload Isolation**
  - Isolate data and workloads from one another as much as possible. Isolation can be achieved by hosting different databases or applications inside different virtual machines or containers, or restricting network access between different workloads. That way, if an attacker is able to gain control of one workload, he won't necessarily be able to access others as well.
- **Hardening Frameworks**
  - Some operating systems provide frameworks that are designed for the specific purpose of adding extra access control and anti-buffer-overflow features to the system and the applications it hosts. AppArmor and SELinux are examples of this type of software on Linux. In general, installing or enabling these tools is a good system hardening best practice.
- **Antivirus:** Install and configure anti-virus software to to detect and remediate malware software

MAC Hardening Actions

innovate achieve lead

- Create a standard account (non-admin) for everyday operations
- Disable automatics login
- Uninstall standalone flash players
- Use password manager to cope with phishing attacks
- Run a two way firewall
- Enable full disk encryption
- Disable spotlight suggestions
- Audit your security and privacy settings
- Check for software updates
- Don’t leave computer unattended and unlocked
- Use VPN software
- Avoid illegal file sharing
- Have a backup solution

Encryption Technologies

What is data encryption?

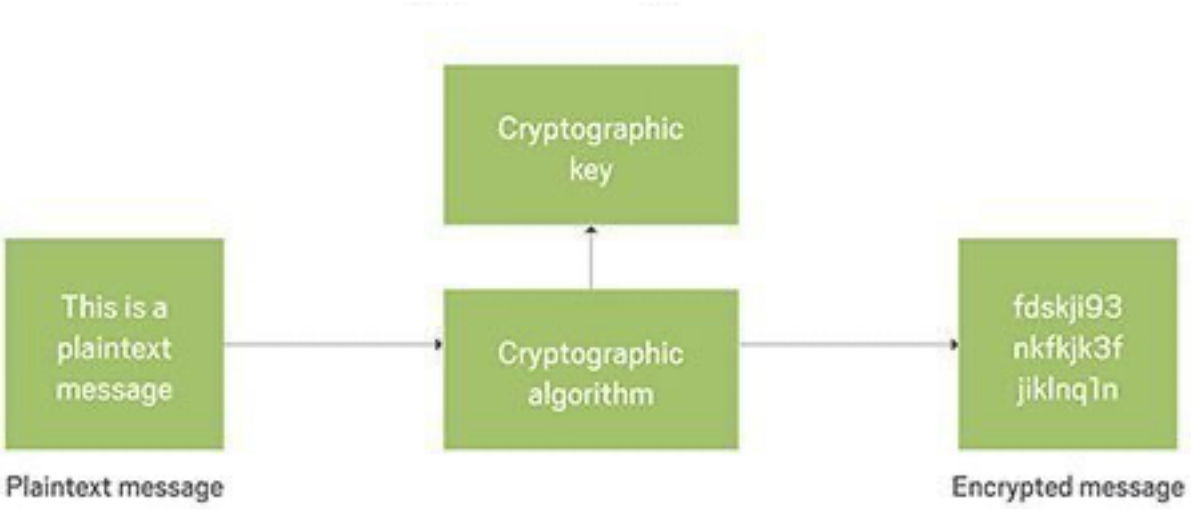
- The conversion of data from a readable format into an encoded format that can only be read or processed after it's been decrypted

- Example:

  Plain Text: Lets meet for coffee at 4:00pm at Barista

  Encrypted Text:
  fUfDPzlyJu5LOnkBAf4vxSpQgQZltcz7LWwEtrughon5kSQIkQlZtfxtSTstutq6gVX4SmlC3
  A6RDAhhL2FfhfoeimC7sDv9G1Z7pCNzFLp0lgAWWA9ACm8r44RZOBiO5skw9cBZjZVfg
  mQ9VpFzSwzLLODhCU7/2THg2iDrW3NGQZfz3SSWviwCe7GmNIvp5jEkGPCGcla4Fgdp
  /xuyewPk6NDlBewftLtHJVf

How does encryption Work?



- The conversion of data from a readable format into an encoded format involves use of an encryption key and algorithm

Why do we need encryption?

- **Authentication:** To ensure that the participants in a network transactions are legitimately what they claim to be. An SSL certificate can ensure that your are talking to right website.
- **Privacy:** To guarantee confidentiality of the messages or data except the legitimate recipient or data owner. Encryption prevents cybercriminals, hackers, ISPs etc from accessing and reading personal data.
- **Regulatory Compliance:** Many industries and governments have rules in place that require organizations that work with users' personal information to keep that data encrypted. A sampling of regulatory and compliance standards that enforce encryption include HIPAA, PCI-DSS, and the GDPR.
- **Security:** Encryption helps protect information from data breaches, whether the data is at rest or in transit. For example:
  - If a corporate-owned device is misplaced or stolen, the data stored on it will most likely be secure if the hard drive is properly encrypted.
  - Encryption protect data against malicious activities like man-in-the-middle attacks, and lets parties communicate without the fear of data leaks.

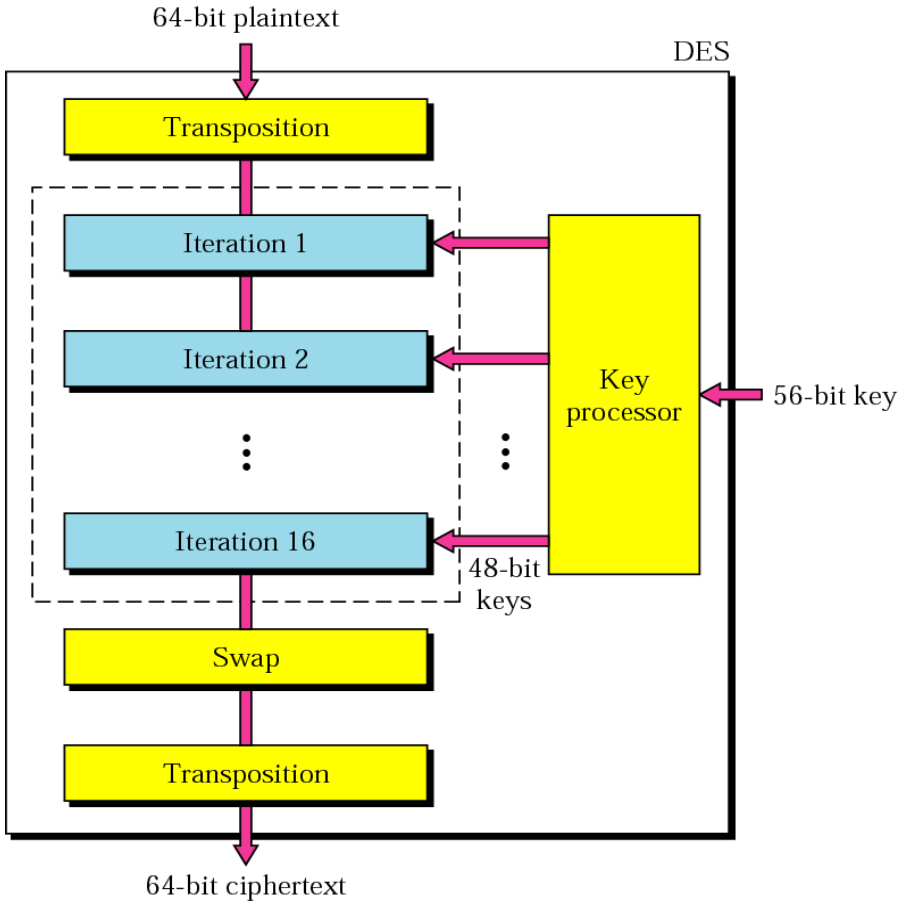
Type of encryption

- **Symmetric:** Sender and the receiver share the same key. The recipient needs to have the key before the message is decrypted. This method works best for closed systems, which have less risk of a third-party intrusion.
  - It is faster than asymmetric encryption. However, both parties need to make sure the key is stored securely and available only to them only.
- **Asymmetric:** Uses two keys for the encryption process, a public and a private key, which are mathematically linked. The user employs one key for encryption and the other for decryption. Public key is freely available to anyone, whereas the private key remains with the owner only.
- **Hashing:** Generates a unique signature of fixed length for a data set or message. Each specific message has its unique hash, making minor changes to the information easily trackable. Data encrypted with hashing cannot be deciphered or reversed back into its original form. That's why hashing is used only as a method of verifying data.
  - Its an effective way of showing that no one has tampered with the information.

Encryption algorithms

- **AES:** The Advanced Encryption Standard is the trusted standard algorithm used across world. It uses keys of 128, 192 and 256 bits for encryption. AES is widely considered invulnerable to all attacks except for brute force.
- **Triple DES:** Triple DES is created in response to hackers who figured out how to breach DES. TripleDES applies the DES algorithm three times to every data block and is commonly used to encrypt UNIX passwords and ATM PINs.
- **RSA:** RSA is a public-key encryption asymmetric algorithm that works off the factorization of the product of two large prime numbers. Only a user with knowledge of these two numbers can decode the message successfully. RSA creates a massive bunch of gibberish that frustrates hackers, causing them to expend a lot of time and energy to crack into systems. Digital signatures use RSA, but the algorithm slows down when it encrypts large volumes of data.
- **Blowfish:** This symmetric tool breaks messages into 64-bit blocks and encrypts them individually. Blowfish is fast, flexible, and unbreakable. Blowfish is used for e-commerce platforms, securing payments, and password management tools.
- **Twofish:** It's a symmetric encryption that deciphers 128-bit data blocks. It is perfect for both software and hardware environments and is considered one of the fastest of its type. Many of today's file and folder encryption software solutions use this method.

General Scheme of DES

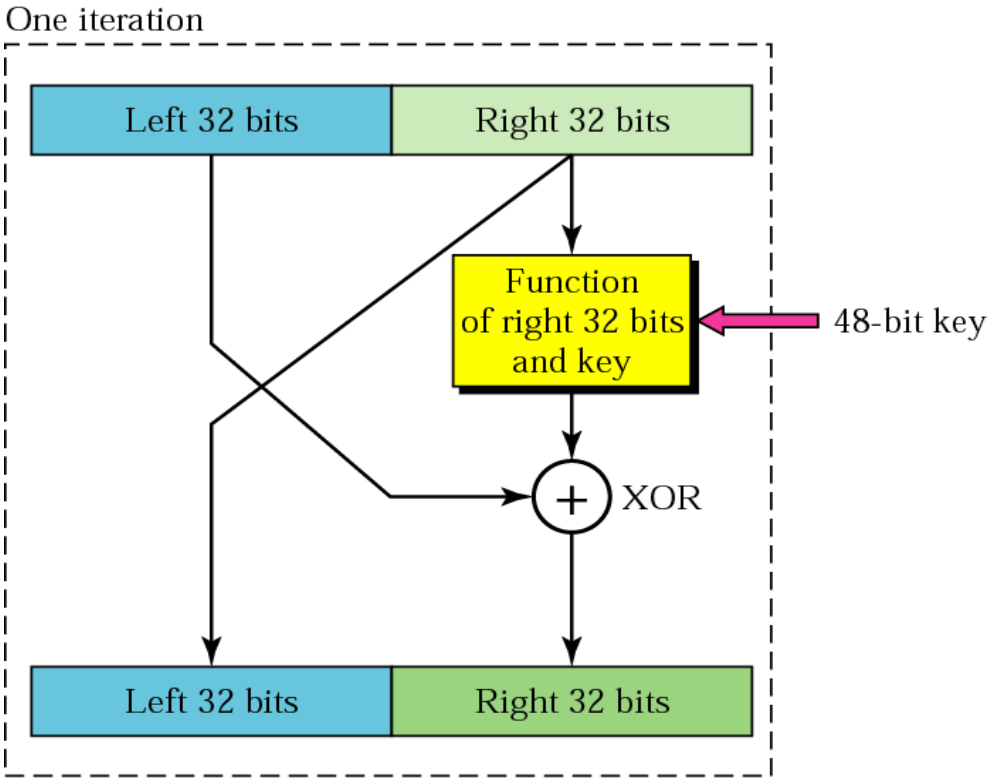


- **16 iterations (rounds):** Each round contain
  - Substitution (Confusion)
  - Transpositions (diffusion)
- **Avalanche Effects:** A small change in either plaintext or key must result in a significant change in ciphertext

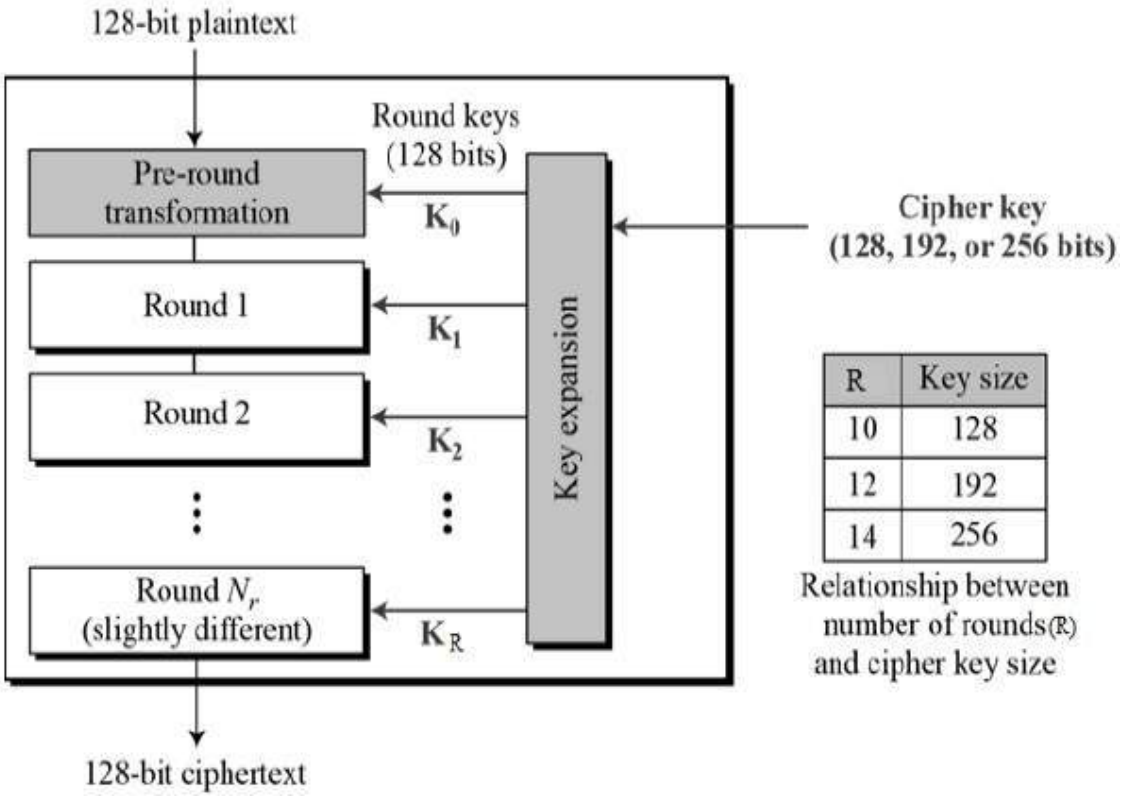
Strength of DES:

- **Use of 56-bit key:** 2^{56} or 7.2×10^{16} keys hence brute-force attach will be difficult. High speed computing @ 10^{13} encryptions per second would require about 1 hour to break
- **Nature of DES algorithm:** DES usage 8 substitution tables (S-Boxes) for iterations. These are secret and may have some weakness though so far no one claimed to break these tables
- **Timing attack:** key principle being that an encryption algorithm will take different amount of time for different inputs. By providing varying inputs and measuring time difference one can try guessing the key.

Iteration Block



AES Outline

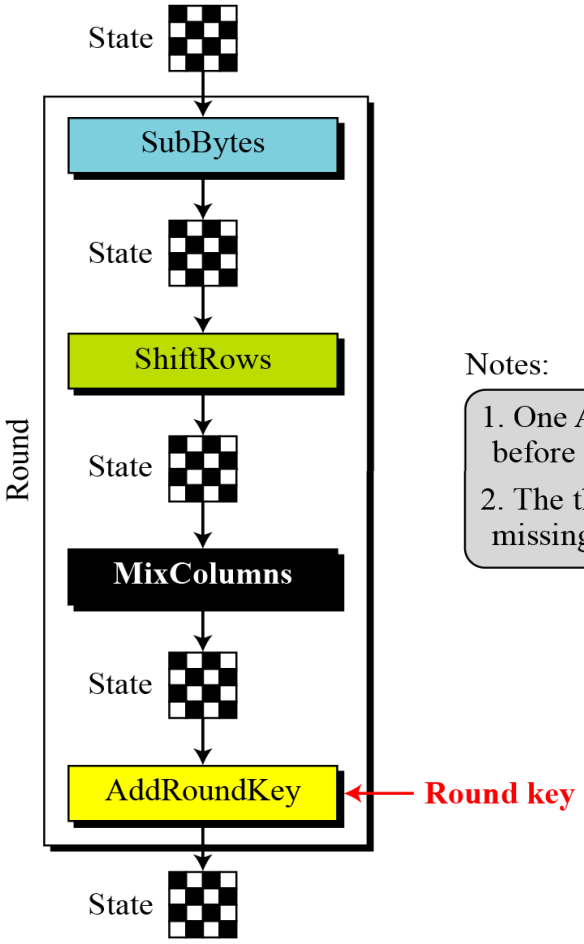


R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

- Encrypts and decrypts using 128 bits of data block
- Usage 10, 12 or 14 rounds
- Usage a 128, 192 or 256 bit key
- Pre-round and Last round are different in structure
- Rest rounds have same structure (step)

Structure of a round



Notes:

1. One AddRoundKey is applied before the first round.
2. The third transformation is missing in the last round.

- Encrypts and decrypts using 128 bits of data block
- Usage 10, 12 or 14 rounds
- Usage a 128, 192 or 256 bit key

- Pre-round has only AddRoundKey step
- Last round doesn't have mix column step

ROOTKIT

Rootkit

- A piece of code that sits in between operating system and hardware
- Rootkit can circumvent, disable or alter the working of operating system
- In Unix/Linux/Windows Root/Admin is the most privileged subject – getting Root/Admin access is the ultimate goal of a hacker
- **Rootkit:** A piece of code which attains Root privileges
- The term rootkit is combination of two words: root + kit. "Root" refers to the administrator account with full privileges and unrestricted access. ”Kit" refers to the programs that allow a threat actor to obtain unauthorized root/admin access to the computer.
- The rootkit enables the threat actor to perform all these actions surreptitiously without the user's consent or knowledge.

Rootkit Types

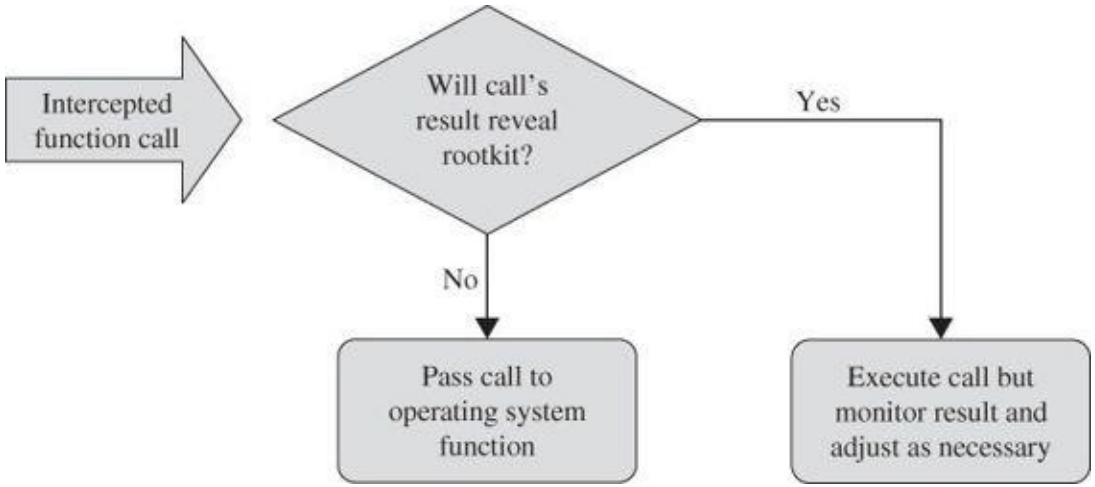
- **Hardware or firmware rootkit:** This type of malware could infect your computer's hard drive or its system BIOS, the software that is installed on a small memory chip in your computer's motherboard. It can even infect your router. Hackers can use these rootkits to intercept data written on the disk.
- **Bootloader rootkit:** It loads your computer's operating system when you turn the machine on. A bootloader toolkit, then, attacks this system, replacing your computer's legitimate bootloader with a hacked one.
- **Memory rootkit:** This type of rootkit hides in your computer's RAM, or Random Access Memory. These rootkits will carry out harmful activities in the background. They only live in your computer's RAM and will disappear once you reboot your system, though sometimes further work is required to get rid of them.
- **Application rootkit:** Application rootkits replace standard files in your computer with rootkit files. These rootkits might infect programs such as Word, Paint, or Notepad. Every time you run these programs, you will give hackers access to your computer.
- **Kernel mode rootkits:** These rootkits target the core of your computer's operating system. Cybercriminals can use these to change how your operating system functions.

Use of Rootkits: Malicious

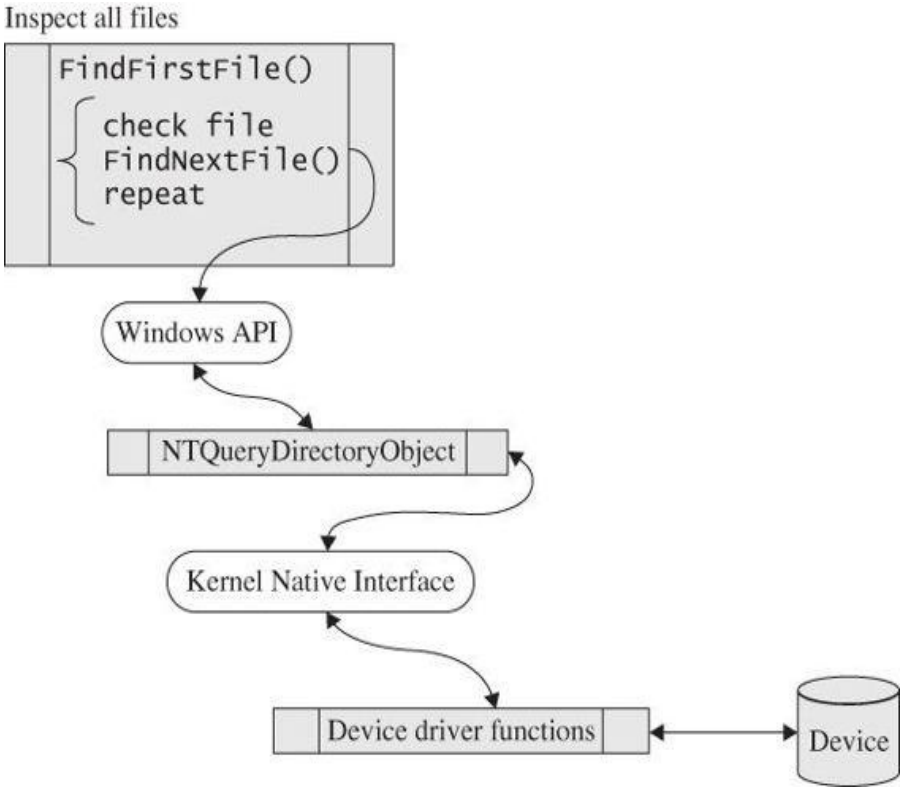
- **Stealth capabilities:** Modern rootkits add stealth capabilities to malicious software payloads (such as keyloggers and viruses) to make them undetectable.
- **Backdoor access:** Rootkits permit unauthorized access through backdoor malware.
  - Subverts the login mechanism and create a secret login access for the attacker.
  - Bypass authentication and authorization mechanisms to provide admin privileges to the attacker.
- **DDoS attacks:** Rootkits allow the compromised computer to be used as a bot for distributed-denial-of-service attacks.
  - The attack would now be traced to the compromised computer and not to the attacker's system.
  - These bots (zombie) can be used as part of bot networks to launch the DDoS attacks, and other malicious activities such as click fraud and spam email distribution.

How rootkit evades detection

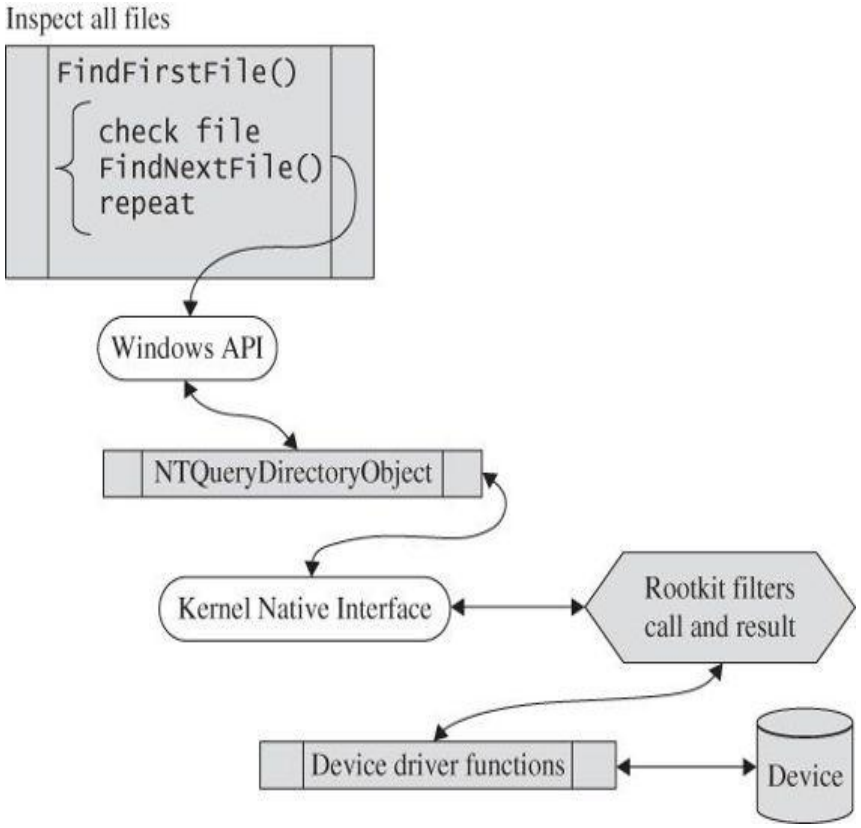
- Rootkits intercept the operating systems calls then alter results of the call if required. This allows rootkit to evade it's detection – antivirus tools or operating system tools



How rootkit evades detection…



Normal OS call execution



Rootkit controlled OS call execution

Rootkit operates unchecked

innovate achieve lead

- Rootkits are difficult to detect, and eradicate.
- 7% of malicious code is rootkits
- Rootkits interfere with normal OS functions to remain hidden
- Normal trick is to intercept the file directory enumeration call to conceal rootkit's presence
- Rootkit revealer is a program that directly interfaces with disk or file system and enumerates files and compares this with the OS function results

Use of Rootkits: Good Cause

- In a honeypot to detect attacks
- To enhance emulation software
- To enhance security software – it enables the software to secure itself from malicious actions
- Digital rights management enforcement
- Device anti-theft protection - BIOS-based rootkit software enables monitoring, disabling and wiping of data on mobile devices when they get lost or stolen

Example: Phone Rootkit

- Mobile phone operating system is simple and less secure
- Researchers at Rutgers University planted rootkit on phones
- Test 1: operate microphone of the phone without users knowledge. This allowed them to eavesdrop on the user conversation
- Test 2: obtain location of the phone thru it's GPS system thru a text message hence tracking whereabouts of a user
- Test 3: put on power hungry functions of a phone like Bluetooth radio and GPS receiver thus draining the battery faster and putting it out of use for emergencies
- All attacks were undetectable

Rootkit Examples

- Stuxnet
- Sony BMG Copy Protection
- Lion
- NTRootkit

Sony XCP rootkit

- In 2005 Mark Russinovich detected a rootkit on his machine
- The rootkit was installed when a Sony music CD was inserted to be played on the computer
- Sony used an auto install program to install the rootkit when a CD was inserted first time in the CD drive.
- This would allow the Sony music player to play the CD songs but would prevent any other program to read the CD
- This was basically a copy protection mechanism (XCP – eXtended Copy Protection)
- Rootkit hid the display of any program starting with sys from any source – malicious or not
- It created a vulnerability in user systems – they were open to attack by virus writers who could name their virus as sysviru-1 and so on

Sony XCP rootkit…

- Sony released an uninstaller which was thought thru properly
- Shortsighted solution opened further security holes than to fix the issue
- Uninstaller was presented thru a web page to be downloaded and run
- Uninstaller didn’t check what code was being executed on the machine and the uninstaller remained on the machine even after XCP was uninstalled
- Count of this rootkit victims but it was upwards of 500000

TDSS rootkit

- Family of rootkits TDL-1 to TDL-4 based on Alureon rootkit
- TDL-1 originated in 2008 with basic objective of collecting and extracting personal data
- TDL-1 hides itself using:
  - Installed filter code in the stack of divers associated with access to each disk drives. Filters drop all files with ‘tdl’ as name prefix
  - TDL-1 could install as many files on as many disk drives anywhere
- Windows registry was loaded with entries to cause these malicious drivers to reload on every system startup
- TDL-1 hides these registry entries by modifying system function NTEnumerateKey used to list data items in registry
- Early 2009 TDL-2 was released. Functionality was same except that the code was scrambled, encrypted and padded with nonsense data
- Later in 2009, TDL-3 was released. It implemented it’s own file system thus becoming completely independent of Windows file system (FAT and NTFS)

TDSS rootkit…

- TDL-3 also introduced command and control structure with which rootkit communicate to receive work assignments and return collected data
- TDL-3 used encrypted communication streams effectively preventing security analysts to check it's growth
- In 2009, more than 3 million computers were infected by TDL-3 (more than half in US)
- TDL-4 appeared in autumn 2010 and circumvented Microsoft security techniques
- Microsoft implemented an cryptographic technique by which it encrypted part of it's each driver using a digital signature
- That helps it verify source integrity of kernel code each time it was loaded
- TDL-4 changed a system configuration value LoadIntegrityCheckPolicy to allow unsigned rootkit to be loaded
- TDL-4 also infects Master Boot Record (MBR) and replaces kernel debugger (kdcomm.dll) to always return safe values

Other rootkits

- Not all rootkits are malicious
- As a security in-charge of very sensitive information (medical records of high profile patients, intellectual property etc) one may want this information to be available internally but restrict it go out.
- Rootkits can control this.
- Products like eBlaster or Spector are rootkits which allow parents to monitor email, messaging, web searches etc on their kids computers
- Law enforcement agencies use rootkits on suspect machines to gather evidence
- Security tools like anti-virus of intrusion detection tools also sometime act in stealth and hard to detect manner like rootkits

Defense Against Rootkits: Preventive

- **Don’t ignore OS or standard application updates:** Keeping operating system, antivirus software, and other applications updated
- **Watch out for phishing emails:** Never click on any links on phishing email or a suspicious looking email from known sources (friends, companies etc)
- **Be careful of drive-by downloads:** Drive-by downloads happen automatically when you visit a website and it installs malware on your computer. Keep always most up-to-date protections in place for your computer system.
- **Don’t download files sent by people you don’t know:** Be careful when opening attachments sent by unknown people.
- Regularly run anti-virus and occasionally run anti-rootkit tools on sensitive machines.
- Use behavioural-based detection (analyse system behaviour) to discover suspicious patterns of API calls or CPU usage, which may indicate a rootkit.
- Closely examine network logs from packet analysers, firewalls, or other network tools to identify rootkits communicating with a remote control centre

Defense Against Rootkits: Scanners

- Scanners are programs designed to parse a system in order to weed out active rootkits.
- Scanners can help detect and remove application-layer rootkits
- Scanners are ineffective against rootkits operating at the kernel, boot or firmware level.
- No individual scanner can guarantee that a system is completely clean, hence use multiple scanners and rootkit removers.
- To fully secure system from rootkits operating at the boot, firmware or hypervisor level, the only remedy is to backup data, then wipe the device and perform a clean install.

Thank You