System Hacking

System Hacking: Goals

Hacking-Stage	Goal	Technique/Exploit Used
Gaining Access	Bypass access controls to gain access to the system	Password Cracking. Social Engineering
Escalating Privileges	Acquire the rights of another user or admin	Exploiting known system vulnerabilities
Executing Application	Create & maintain remote access to the system	Trojans, Spywares, Backdoors, Keyloggers
Hiding Files	Hide attackers malicious activities & data theft	Rootkits, Steganography
Covering Tracks	Hide the evidence of compromise	Clearing logs

Password Cracking

- Used to **recover passwords** from computer system
- **Gain unauthorized access** to vulnerable system
- Successful due to weak or easily **guessable password**

Types of Password Attacks

- **Non-Electronic Attacks**
  - Shoulder Surfing
  - Social Engineering
  - Dumpster Diving
- **Active Online Attacks**
  - Dictionary & Brute Forcing Attack
  - Hash Injection & Phishing
  - Trojan/Spyware/Keyloggers
  - Password Guessing
- **Passive Online Attacks**
  - Wire Sniffing
  - Man-in-the-Middle
  - Replay
- **Offline Attacks**
  - Pre-Computed Hashes (Rainbow Table)
  - Distributed Network

Active Online Attack

Dictionary Attack

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

Brute Forcing Attack

Program tries **every combination of characters** until the password is broken

Rule-based Attack

Attack is used when the attacker get some **information about the password**

Password Guessing

Creates list of possible passwords through **social engineering** & tries them to crack **manually**

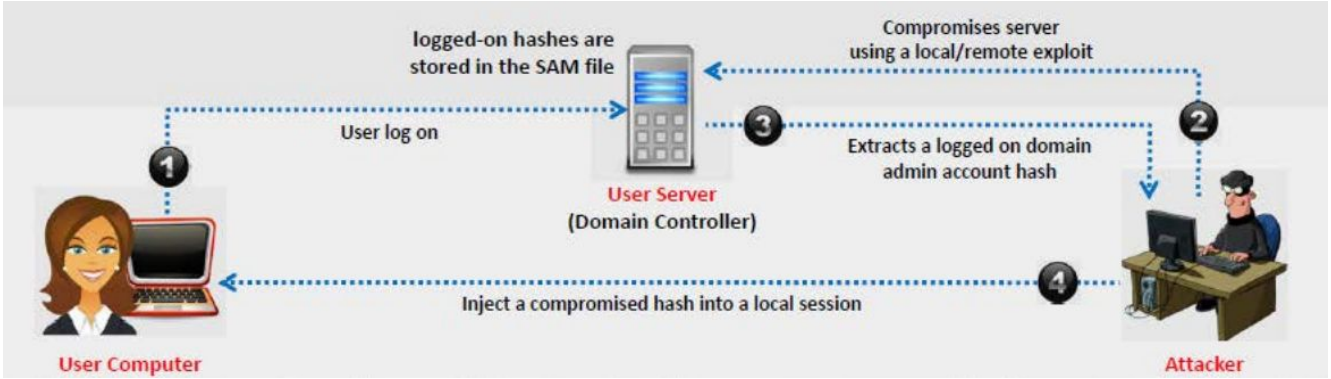
Trojan/Spyware/Keylogger s

Attacker installs Trojan/Spyware/Keyloggers on

Active Online Attack

Hash Injection Attack

- **Inject a compromised hash** into a local session & use the hash to validate network resources
- Finds & extracts a logged on **domain admin account hash**
- Uses the extracted hash to log on the **domain controller**



Wire Sniffing

- Attacker runs **packet sniffer tools** on the LAN to access & record the raw network traffic
- Captured data includes **sensitive information** like passwords & emails
- Sniffed credentials are used to **gain unauthorized access** to the target system

Wire Sniffing > Computationally Complex > Available Tools > Hard to Perpetrate

Active Online Attack

Man In the Middle

Attacker acquires **access** to the communication channels between victim & the server to extract information

Replay Attack

Packets & authentication tokens are captured using **sniffer**. After extracting information, token 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 precomputed table which contains word lists like **dictionary files** & **brute force lists** & their h**ash values**

Compare the Hashes

Capture the **hash of a password** & compare it with the precomputed hash table. If matched then the password is cracked

Easy to Recover

Easy to recover passwords by comparing captured password hashes to the **precomputed tables**

Distributed Network Attack (DNA)

A technique used for **recovering password from hashes or password protected files** using the unused processing power of the machine across the network to decrypt passwords

- 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 & **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 the processing capabilities of all the clients connected to network & uses it to **crack the password**

Microsoft Authentication

Security Accounts Manager(SAM) Database

- Windows stores user passwords in SAM or **Active Directory Database** in domains
- Passwords are hashed & the results are stored in the SAM

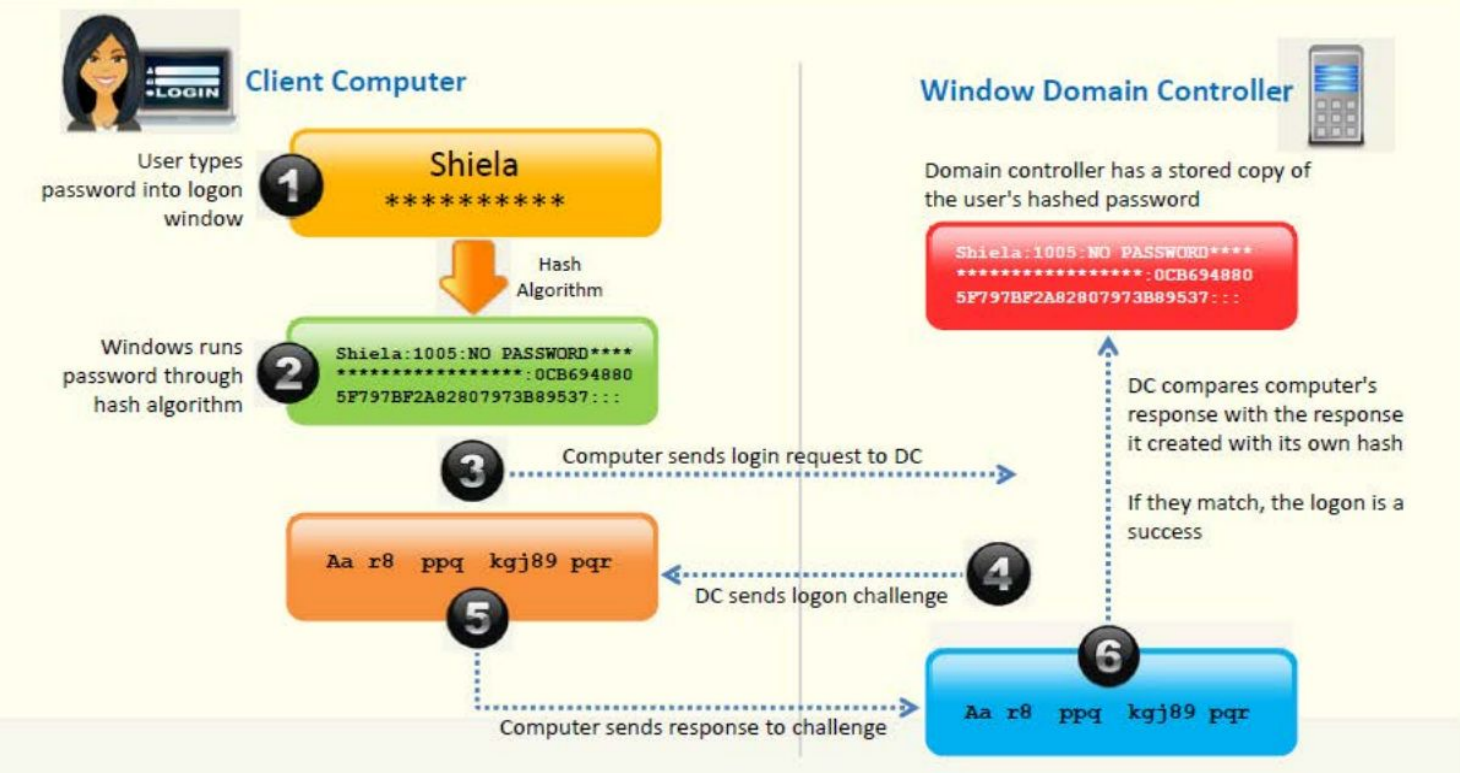
NTLM Authentication

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

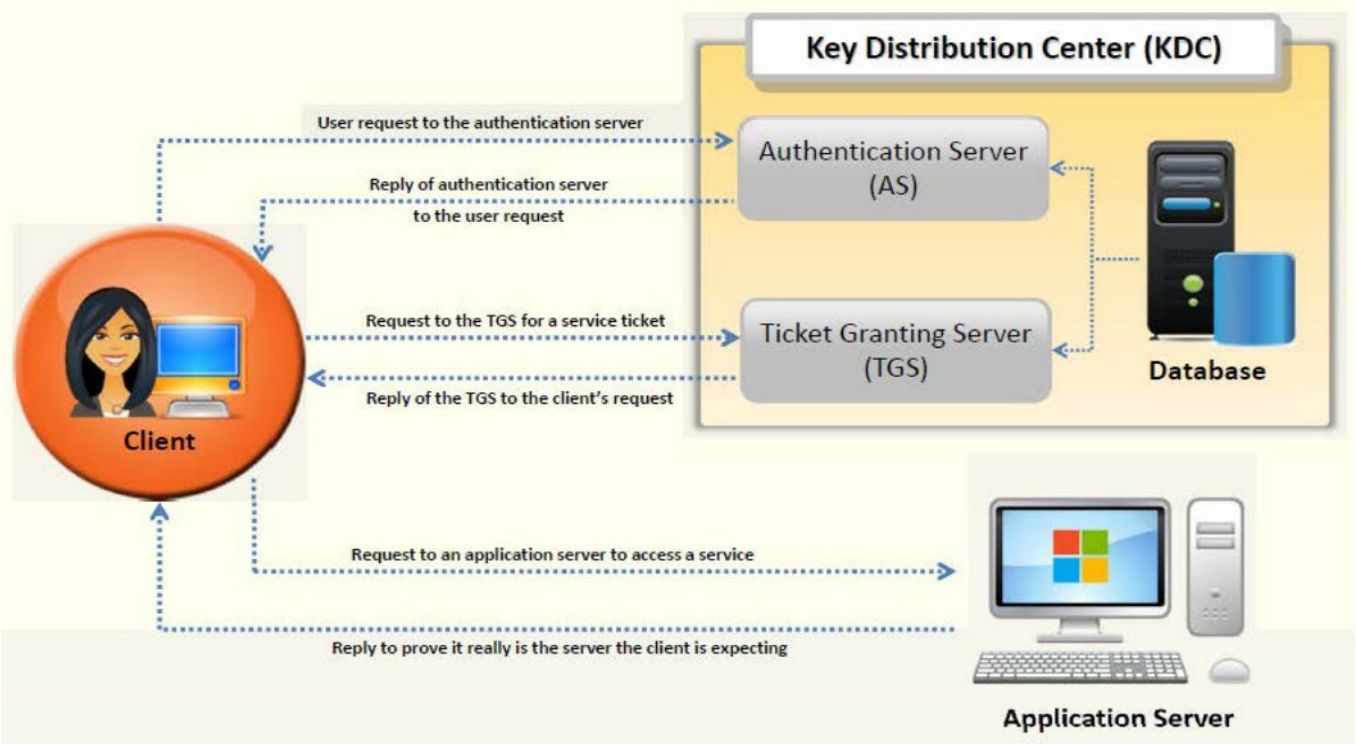
Kerberos Authentication

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

NTLM Authentication Process



Kerberos Authentication



Password Salting

- A technique where **random string of characters are added** to the password before calculating their hashes
- Salting makes it more **difficult to reverse** the hashes & **defeats** pre-computed hash attacks

Note: Windows password hashes are not salted

How to Defend against Password Cracking

- Enable **information security audit** to monitor & track password attacks
- Don’t use the **same password** during password change
- Don’t **share** passwords
- Don’t use password that can be found in **dictionary**
- Don’t use **cleartext** protocols & protocols with **weak encryption**
- Set the **password change policy** to 30 days
- Avoid **storing password** in an unsecured location
- Don’t use any system’s **default passwords**
- Make passwords **hard to guess** by using **alphanumeric** characters
- Ensure that applications **neither store** passwords to memory **nor write** them to disk
- Us a **random string**(salt) as prefix or suffix with the password before encrypting
- Enable **SYSKEY** with strong password to encrypt & protect SAM database
- Never use passwords such as **date of birth**, spouse or child or pet’s name
- Monitor the **server’s logs** for brute force attacks
- Lock out an account subjected to too many **incorrect password** guesses

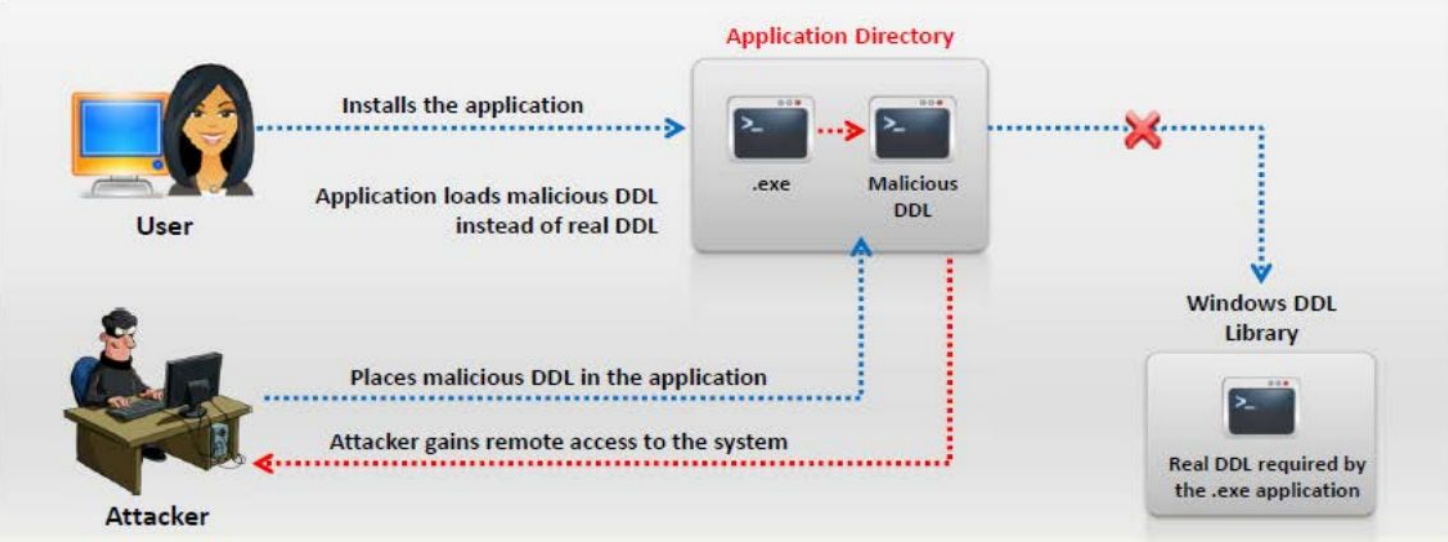
Privilege Escalation

- Gain access to the network using a **non-admin user account**
- Performs privilege escalation attack which takes advantage of **design flaws**, **programming errors**, **bugs** & **configuration oversights** on the OS & application to gain administrative access
- These privileges allows attacker to **view critical/sensitive information**, delete files or install malicious programs
- Types of Privilege Escalation:
  - **Vertical Privilege Escalation**
    - Refers to gaining higher privileges than the existing
  - **Horizontal Privilege Escalation**
    - Refers to acquiring the same level of privileges assuming the identity of another user

Privileges Escalation

DLL Hijacking

- Most windows application don’t use **fully qualified path** when loading an external DLL library
- They search directory from which they have been loaded first
- Attacker places **malicious DLL in the application directory**, instead of real one it will be executed



How to Defend against Privileges Escalation

- Restrict the **interactive logon privileges**
- Use **encryption technique** to protect sensitive data
- Run users & application on the **least privileges**
- Reduce the **amount of code** that run on particular privilege
- Implement **multi-factor authentication** & **authorization**
- Perform **debugging** using bounds checkers & stress tests
- Run services as **unprivileged accounts**
- Test operating system & **application coding errors** & **bugs** thoroughly
- Implement a **privilege separation methodology** to limit the scope of programming errors & bugs
- **Patch the system** regularly

THE END