# Privilege Escalation

In the privilege escalation phase, attackers **elevate their permissions within a compromised system, gaining higher-level access** that allows them to control more resources and execute more damaging actions. Privilege escalation (often shortened to PrivEsc) is essential for attackers to **bypass access restrictions**, reach sensitive data, or maintain control over a network. Some privilege escalation methods overlap with persistence techniques, as attackers can use them to ensure long-term access at higher privilege levels.

Here are some common privilege escalation techniques, including **Sudo exploits, token/key theft, IAM/group policy modifications, and the use of persistence exploits** as PrivEsc methods.

## 1. Sudo Exploits

- Definition: On Unix-based systems, sudo allows users to execute commands with elevated privileges (typically as the root user). Attackers may attempt to **exploit misconfigurations in sudo or vulnerabilities to gain root access**.
- Common Sudo Exploits:
  - **Misconfigured Sudoers File**: Sometimes, the sudoers file (which controls sudo permissions) grants more privileges than necessary. Attackers can exploit such configurations to execute commands as root.
  - **Unrestricted Sudo Permissions**: In cases where users are allowed to execute all commands via sudo without restrictions, attackers can quickly escalate their privileges.
- Security Implications: If attackers can exploit sudo, they gain root privileges, allowing them to modify critical system files, install software, and control almost every aspect of the system.

## 2. Token/Key Theft

- Definition: Many operating systems and cloud environments **use tokens, keys, and credentials to authenticate and authorize users and processes**. Attackers may steal these tokens or keys to gain elevated privileges.
- Methods of Token/Key Theft:
  - **Access Tokens (Windows)**: Attackers can steal tokens that represent user sessions, impersonating users with higher privileges.
  - **Cloud Access Keys**: In cloud environments (e.g., AWS, Azure), attackers may attempt to steal access keys or tokens tied to privileged accounts to control cloud resources.
  - **Session Cookies**: Attackers may steal session cookies from web browsers or applications to impersonate users without needing their passwords.
- Security Implications: Stolen tokens and keys grant attackers higher access without requiring further exploitation, allowing them to bypass access controls, manipulate data, and control resources directly.

## 3. IAM/Group Policy Modifications

- Definition: In enterprise and cloud environments, Identity and Access Management (IAM) systems and group policies control user permissions and access rights. Attackers may **modify IAM policies or group memberships to escalate their privileges**.

- Techniques:
  - **IAM Policy Modification**: Attackers with sufficient access may modify IAM policies to grant themselves broader permissions, such as administrator rights in cloud environments.
  - **Adding to Privileged Groups**: Attackers with local admin rights on Windows, for example, can add their account to higher-privilege groups (e.g., Domain Admins or Local Administrators).
- Security Implications: By altering IAM policies or group memberships, attackers gain access to additional resources or permissions, making it easier for them to control the environment and access sensitive data.

## 4. Persistence Exploits as Privilege Escalation Methods

- Definition: Many persistence techniques double as privilege escalation methods, as they help attackers gain or retain higher privileges.
- Examples:
  - **Scheduled Tasks**: If attackers create scheduled tasks under system or administrator accounts, they gain elevated access upon task execution.
  - **DLL Side-Loading**: Attackers can plant malicious DLLs in directories used by privileged applications. When the application loads the DLL, it runs with elevated privileges, effectively escalating the attacker's access.
  - **WMI Persistence**: Attackers who use WMI scripts for persistence can configure them to execute with system-level privileges, providing elevated access.
- Security Implications: By using persistence techniques that allow elevated permissions, attackers not only gain continued access but also ensure they can operate with high-level privileges, increasing the potential damage they can cause.

## Summary

- Sudo Exploits target privilege misconfigurations in Unix-based systems to gain root access.
- Token/Key Theft involves stealing access tokens, session cookies, or cloud keys, allowing attackers to impersonate privileged users.
- IAM/Group Policy Modifications enable attackers to expand their permissions by altering user roles and group memberships.
- Persistence Exploits as PrivEsc Methods use scheduled tasks, DLL side-loading, and WMI to establish persistent elevated access, often doubling as privilege escalation.

These privilege escalation methods are critical in allowing attackers to deepen their control over a compromised environment, enabling them to perform more sophisticated actions and access sensitive resources. **Regular auditing of user privileges, monitoring group memberships, and securing token storage are effective ways to mitigate these threats**.