Testing for vulnerabilities

Privilege escalation

INTRODUCTION

- Privilege escalation: the act of exploiting a bug, design flaw or configuration oversight in an operating system or software application to gain elevated access to resources that are normally protected from an application or user. The result is that an application with more privileges than intended by the application developer or system administrator can perform unauthorized actions.
- Any method that allows you to obtain increased privileges
- Horizontal privilege escalation
  - user1 -> user2
- Vertical privilege escalation
  - user -> root/system
  - vm -host >
  - user/kernel/vm -> firmware

INTRODUCTION

- Compromising a system is the most important step in pentesting
- To take advantage of it, however, we need increased privileges
- Motivation: A limited account does not allow us to perform post-enumeration
  - Hashes, credentials, confidential files
- Privilege escalation refers to any technique that allows for increased privileges to be obtained
  - Exploits can vary greatly

PRIVILEGE LEVELS: HARDWARE (X86)

- In hardware, there is a separation of privilege levels (starting with protected mode)

Ring 3	User-mode	User-mode
Ring 2		Supervisor mode
Ring 1		Supervisor mode
Ring 0	Kernel-mode	Supervisor mode

- This separation is enforced at processor level

PRIVILEGE LEVELS: HARDWARE (X86)

- Recent processors contain new privilege levels

VMX-NonRoot	VMX-Root	SMM	AMT
R3, R2, R1, R0	R3, R2, R1, R0	Firmware	AMT

- Overall, the hardware is well-made
- Problems occur in software: operating system code, hypervisor code, firmware code

PRIVILEGE LEVELS: HARDWARE (X86)

- ARM also contains a privilege-level separation

Non-secure state	Secure state
User Mode	Secure Mode
Supervisor Mode	Secure Mode
Hyp Mode	Secure Mode
Trust Zone	

- In principle it is similar to x86

PRIVILEGE LEVELS: HARDWARE

VM1		VM2		VMM	SMM	Hardware
App11	App12	App21	App22	App1	SMM	Hardware
OS		OS		OS	SMM	Hardware

PRIVILEGE LEVELS: SOFTWARE (WINDOWS)

- Windows: A privilege is the right of an account, such as a user or group account, to perform various system-related operations on the local computer, such as shutting down the system, loading device drivers, or changing the system time. (MSDN)
- Account DB contains privileges for users and groups
- When a user logs in, they are assigned an access token
- This token contains the privileges that the user has
- At each privileged operation, the OS verifies the token
- The most privileged account is named *SYSTEM*

PRIVILEGE LEVELS: SOFTWARE (LINUX)

- Permissions on Linux to spin around files
- Permissions are set for owner, group, and others
- File rights: read, write, execute
- Advanced rights: directory, link, suid (runs the executable with the rights of the file's owner, not with the rights of the current user)
- Certain files contain information about the user:
  - /etc/passwd
  - /etc/shadow
- The most privileged account is called *root*

PRIVILEGE ESCALATION

- To increased privileges, we need to exploit a vulnerability or configuration error
- In general, the vulnerabilities will be in the operating system itself
  - Kernel exploits
  - OS services
- Other times, a configuration error will be exploited
  - Executables with SUID bit set
- Sometimes, vulnerabilities in 3rd party applications will be exploited

PRIVILEGE ESCALATION

- Enumeration
  - Enumeration, enumeration, enumeration
- Data processing
  - Enumerated data, sorted, prioritized, filtered, etc.
- Search
  - Search for a vulnerability or configuration error
- Adaptation
  - You adapt the exploit for your purpose, write the exploit from 0, etc.
- Attack
  - Get root/SYSTEM

PRIVILEGE ESCALATION: LINUX

- Getting the current user
  - id, whoami
- Getting the Linux distribution and kernel version
  - cat /etc/issue, uname -a
- Listing mounted filesystems
  - mount -l
- Viewing Environment Variables
  - /etc/profile, /etc/bashrc, ~/.bash_profile, ~/.bashrc

PRIVILEGE ESCALATION: LINUX

- Get your network configuration
  - ifconfig –a, arp, cat /etc/hosts, cat /etc/resolv.conf, iptables, etc.
- Search for a compiler/interpreter
  - which/find gcc/g++/python/perl/ruby/etc.
- Listing installed packages/apps
  - dpkg –l, rpm –qa, ls -alh /var/cache/yum/, ls -alh /var/cache/apt/archives, etc.
- Enumeration of active network services
  - netstat –antup
- Enumeration of active processes
  - ps aux

PRIVILEGE ESCALATION: LINUX

- Listing scheduled jobs
  - crontab –l, ls -alh /var/spool/cron, ls -al /etc/ | grep cron, ls -al /etc/cron*, cat /etc/cron*, cat /etc/at.allow, cat /etc/at.deny, cat /etc/cron.allow, cat /etc/cron.deny, cat /etc/crontab, cat /etc/anacrontab, cat /var/spool/cron/crontabs/root, etc.
- Listing readable/writable files from sensitive paths (/etc, /home, etc.)
  - find /etc -user 'id -u' -perm -u=r -o -group 'id -g' -perm -g=r -o -perm -o=r –ls
- Search for SUID/GUID executables
  - find / -type f -perm -u=s -o -type f -perm -g=s -ls
- Listing configuration files
  - cat /etc/syslog.conf, cat /etc/chttp.conf, cat /etc/lighttpd.conf, cat /etc/cups/cupsd.conf, cat /etc/inetd.conf, cat /etc/apache2/apache2.conf, cat /etc/my.conf, cat /etc/httpd/conf/httpd.conf, cat /opt/lampp/etc/httpd.conf, ls -aRl /etc/ | awk '$1 ~ /^.*r.*/, etc.

PRIVILEGE ESCALATION: LINUX

- Search by plain-text passwords
  - grep -i user [filename], grep -i pass [filename]
- Listing sensitive files
  - cat /etc/passwd, cat /etc/group, cat /etc/shadow, ls -alh /var/mail/
  - cat /var/apache2/config.inc, cat /var/lib/mysql/mysql/user. MYD cat /root/anacondaks.cfg
  - how much ~/.bash_history, how much ~/.nano_history, how much ~/.atftp_history, how much ~/.mysql_history, how much ~/.php_history

PRIVILEGE ESCALATION: LINUX

- Some (older) CVE’s
  - CVE-2016-8655
  - CVE-2016-5195
  - CVE-2016-4557
  - CVE-2015-1328
  - CVE-2016-3643
- Exim 4 (Debian 8 / Ubuntu 16.04) - Spool Privilege Escalation
- runAV mod_security - Arbitrary Command Execution
- FireEye - Malware Input Processor (uid=mip) Privilege Escalation
- Executabile SUID

PRIVILEGE ESCALATION: WINDOWS

- Determination of the operating system:
  - systeminfo | findstr /b /c:"OS Name" /C:"OS Version"
- Determining the machine and user name
  - hostname, echo %USERNAME%
- User listing
  - net users
- List information about a user
  - net user name
- Listing Network Interfaces
  - ipconfig /all

PRIVILEGE ESCALATION: WINDOWS

- Route listing
  - route Print
- Listing the ARP table
  - arp –A
- List active connections
  - netstat –year
- List firewall information
  - netsh firewall show state
  - netsh firewall show config
- Scheduled tasks listing
  - schtasks /query /fo LIST /v

PRIVILEGE ESCALATION: WINDOWS

- Listing the processes and services they represent
  - tasklist /SVC
- Listing Services Started
  - net start
- Listing Active Drivers
  - driverquery
- WMIC
  - Windows Management Instrumentation Command Line
  - Supports a lot of commands
  - Sometimes, WMIC is not accessible to limited accounts

PRIVILEGE ESCALATION: WINDOWS

- Determining Hotfixes Installed Using WMIC
  - wmic qfe get Caption,Description,HotFixID,InstalledOn
- Determination of service entitlements
  - accesschk.exe -ucqv service

PRIVILEGE ESCALATION: WINDOWS

- Some (older) CVE's
  - CVE-2016-7255
  - CVE-2016-7226
  - CVE-2016-3373
  - CVE-2016-3373
  - CVE-2009-4452
  - CVE-2015-1305
- MySQL UDF

MITIGATIONS

- Hardware mitigations (NX, SMEP, SMAP, CET, etc.)
- ASLR, CFG
- Digital signatures
- Sandboxing
- Assigning privileges on the "need to know" principle
- Updates
- Appropriate configurations

CONCLUSIONS

- After gaining access to a system, we are interested in increasing our privileges
- This step is not imperative, but it is recommended
- There is no "magic guidline" to be followed for a privilege escalation
- It all depends on the vulnerabilities themselves: race-condition, unvalidated input, inappropriate rights, use-after-free, buffer overflow, etc.
- Privilege escalation can be done from one user to another user, from user to kernel, from user/kernel to host, from user/kernel to platform
- Following a privilege escalation, we have full access to the system