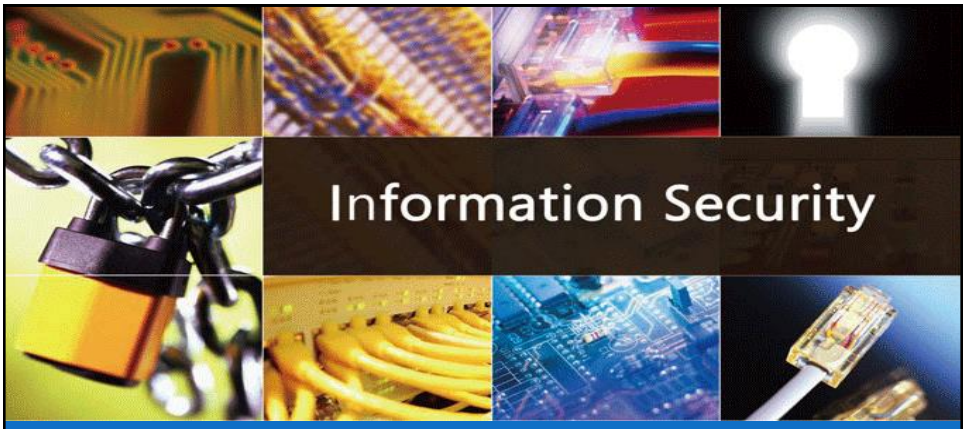


Operating Systems Security

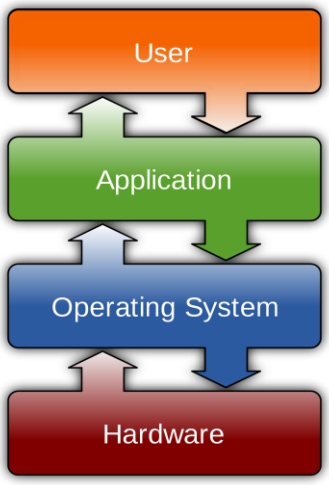
Lecturer: Nguyễn Thị Thanh Vân – FIT - HCMUTE

Contents

- Introduction To OS and OS Security
- System Security Planning
- The Components of an OS Security Environment
- Vulnerabilities of OS
- Secure an operating system
- Operating Systems Hardening
  - Linux/Unix Security
  - Windows Security
- Virtualization Security

Operating System Overview

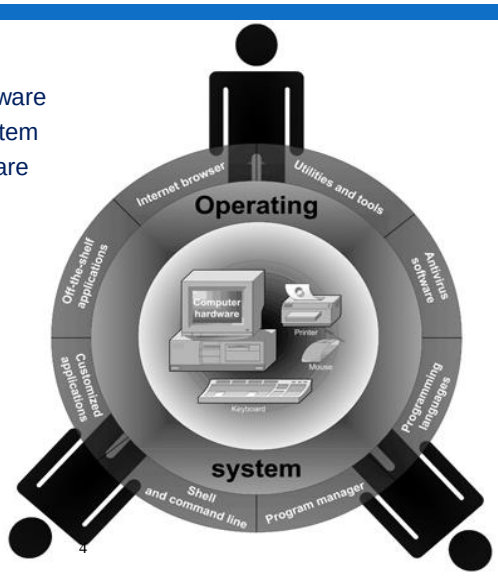
- Operating system: collection of programs that allows user to operate computer hardware



3

Operating System Overview

- Three layers:
  - Inner layer, computer hardware
  - Middle layer, operating system
  - Outer layer, different software



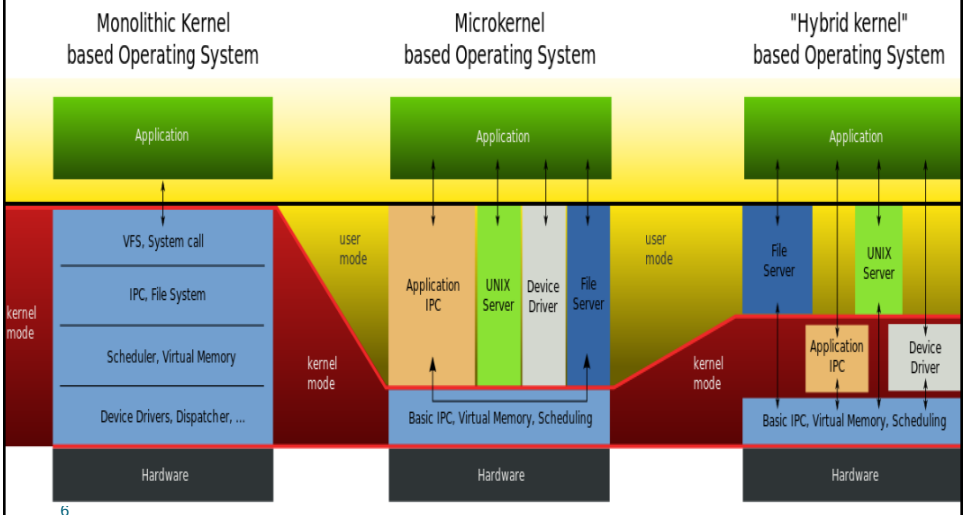
4

Operating System Overview

- Key functions of an operating system:
  - Multitasking, multisharing
  - Computer resource management
  - Controls the flow of activities
  - Provides a user interface
  - Administers user actions and accounts
  - Runs software utilities and programs
  - Enforce security measures
  - Schedule jobs
  - Provide tools to configure the operating system and hardware

5

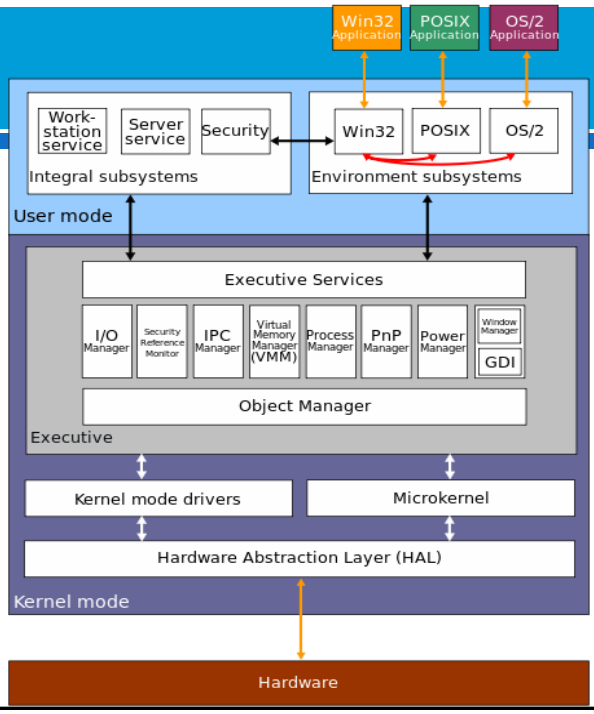
Comparison kernel types



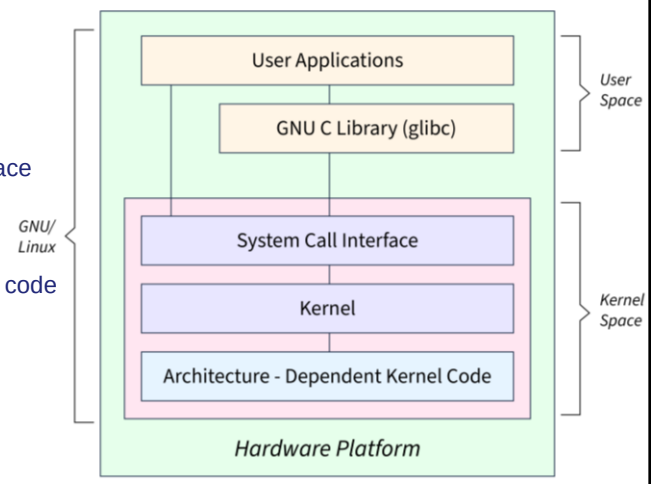
6

Windows kernel

- User mode
- Kernel mode
  - Kernel mode drive
  - Microkernel



- User space
- Kernel space
  - System call Interface
  - Kernel
  - Dependent kernel code



Kernel types

- Comparison Windows & Linux.
- Version:
  - Linux

Initial release	0.01. 17 September 1991; 22 years ago (1991-09-17))
Latest release	4.15-rc8 (14 January 2018)

  - Windows NT:

Initial release	3.1. July 27, 1993 (1993-07-27) (as Windows NT 3.1)
Latest release	1709 (10.0.16299.192) (January 3, 2018)

OS Programming

- Much variation
  - Early OSes in assembly language
  - Then system programming languages like Algol, PL/1
  - Now C, C++
- Actually, usually a mix of languages
  - Lowest levels in assembly
  - Main body in C
  - Systems programs in C, C++, scripting languages like PERL, Python, shell scripts
- More high-level language easier to **port** to other hardware
  - But slower
- **Emulation** can allow an OS to run on non-native hardware

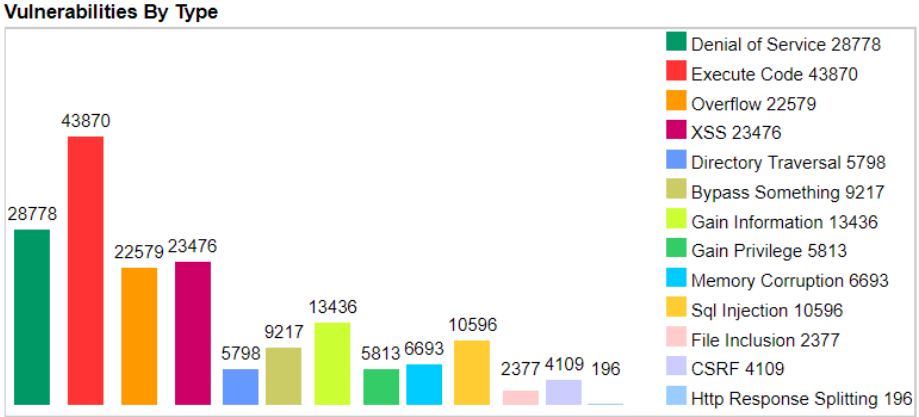
OS Vulnerabilities - Top 50 products (2022)

	Product Name	Vendor Name	Product Type	Number of Vulnerabilities
1	Debian Linux	Debian	OS	6879
2	Android	Google	OS	4639
3	Fedora	Fedoraproject	OS	3645
4	Ubuntu Linux	Canonical	OS	3555
5	Mac Os X	Apple	OS	3019
6	Linux Kernel	Linux	OS	2942
7	Windows 10	Microsoft	OS	2889
8	Iphone Os	Apple	OS	2738
9	Windows Server 2016	Microsoft	OS	2676
10	Chrome	Google	Application	2518
11	Windows Server 2008	Microsoft	OS	2358
12	Windows 7	Microsoft	OS	2208
13	Windows Server 2012	Microsoft	OS	2207
14	Windows Server 2019	Microsoft	OS	2126
15	Windows 8.1	Microsoft	OS	2060

Vulnerabilities by types

- https://www.cvedetails.com/

Vulnerabilities By Type



Trusting an Operating System

Why do we need to **trust** the operating system?
(a **trusted computing base or TCB**)
What requirements must it meet to be trusted?

- HĐH đáng tin cậy thường đề cập đến một HĐH cung cấp đủ hỗ trợ cho bảo mật đa cấp và bằng chứng về tính chính xác để đáp ứng một bộ yêu cầu cụ thể
  - HĐH đó tồn tại một mức độ tin cậy (dựa trên phân tích và thử nghiệm nghiêm ngặt) rằng các nguyên tắc và cơ chế bảo mật (ví dụ: tách biệt, cô lập, đặc quyền tối thiểu, kiểm soát truy cập, đường dẫn đáng tin cậy, xác thực và thực thi chính sách bảo mật) được triển khai chính xác và hoạt động như mong muốn ngay cả khi có hoạt động đối đầu.
- HĐH đáng tin cậy dựa trên cơ sở tính toán đáng tin cậy (Trusted computing base - TCB)
- Nó phải đáp ứng những yêu cầu nào để được tin cậy

Trusting an Operating System

TCB Requirements:

1. Complete mediation
   - The reference validation mechanism must always be invoked (before executing security-sensitive operations)
2. Tamperproof
   - The reference validation mechanism must be tamperproof
3. Verifiable
   - The reference validation mechanism must be small enough to be subject to analysis and tests, the completeness of which can be assured

Trusted OS provides:

1. Bảo vệ bộ nhớ: kiểm soát quyền truy cập vào các phần bị hạn chế của không gian c.trình.
2. Bảo vệ tệp: ngăn các chương trình thay thế các tệp hệ điều hành quan trọng.
3. Kiểm soát truy cập đối tượng chung: không gây ảnh hưởng tiêu cực đến những user khác
4. Xác thực người dùng: password, sinh trắc học…
5. Kiểm soát truy cập thiết bị I/O
6. Dịch vụ công bằng được đảm bảo
7. Chính sách: Yêu cầu bảo mật, được xác định rõ ràng, nhất quán
8. Mô hình: Biểu diễn chính sách, chính thức. Không được làm giảm chức năng.
9. Thiết kế: Bao gồm chức năng, tùy chọn triển khai
10. Tin cậy: Xem xét các tính năng, đảm bảo khiến hệ điều hành trở nên đáng tin cậy.

15

TCB and Resource Protection

TCB Controls access to protected resources

- Must establish the source of a request for a resource (authentication is how we do it)
- Authorization or access control
- Mechanisms that allow various policies to be supported

TCB design principles

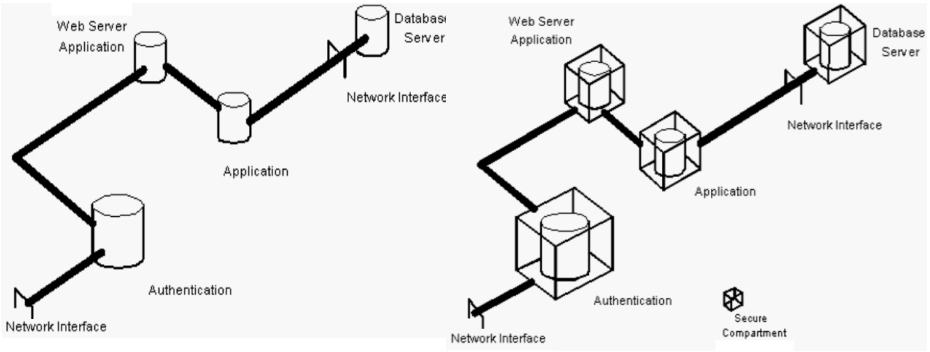
- Least privilege for users and programs
- Economy
  - Trusted code small
  - Easier to analyze & test
- Open design
- Complete mediation
  - Access check,
  - Prevent the bypass
- Fail-safe default
  - Deny: default
- Ease of use

Trusted Operating System – Architecture

Standard Architecture:

Open - after the user passes initial authentication

Trusted Architecture: separate components from each other and make access between areas more difficult



OS Protection - Hardware Support

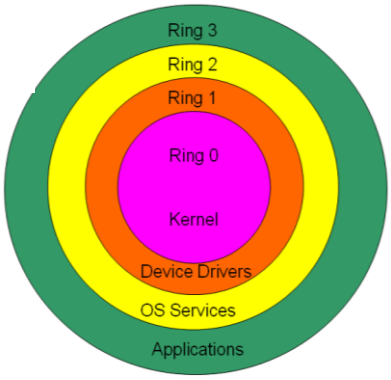
- Hardware Support: Protected Mode - Protected Virtual Address Mode, is an operating mode of x86-compatible central processing units (CPUs).
  - Virtual Memory,
  - Paging
  - Safe Multitasking is designed to increase the operating system's control over application software

OS Protection – execution mode

- Processor execution modes: ensure the safety of the kernel, avoid user program intrusion into the kernel
  - kernel mode: high level of privilege. It has direct access to the computer hardware that all programs running in Kernel mode, including the operating system, share the same address space
  - user mode: applications have less privileges, do not have direct access to hardware resources and cannot write to the address space of other applications
- OS only allows some processes to run in Kernel mode.
  - ensure that there are no problems when running in Kernel mode - which can cause the entire operating system to crash

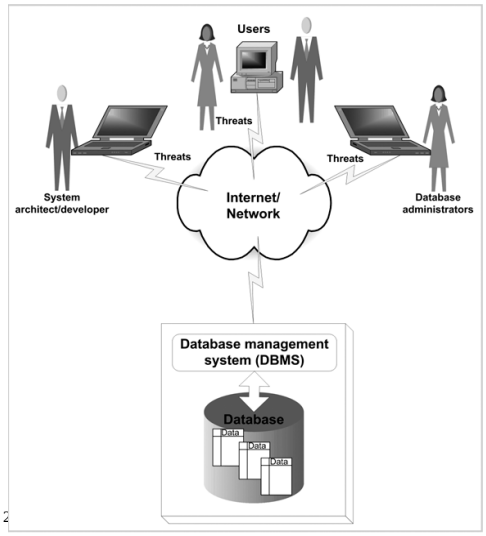
OS Protection - Rings

- More trusted processes operate within lower numbers ring
  - Inner Ring: more Privileges
  - Outer Rings: less
- Rings:
  - Define Access Level to resource.
- Protect system integrity
  - Protect kernel from services
  - Protect services from apps
  - So on..



The OS Security Environment

- A compromised OS can compromise a database environment
- Physically protect the computer running the OS (padlocks, chain locks, guards, cameras)
- Model:
  - Bank building (operating system)
  - Safe (database)
  - Money (data)



The Components of an OS Security Environment

- Used as access points to the database
- Three components:
  - Services
  - Files
  - Memory

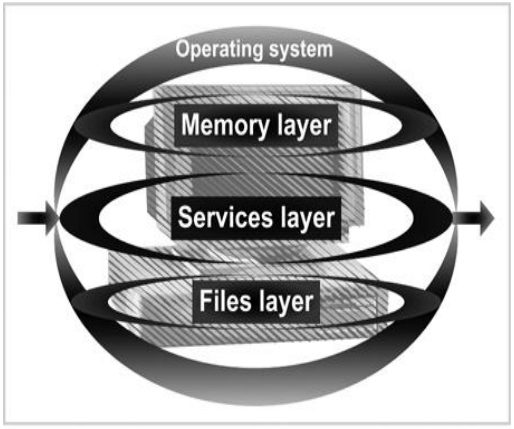


FIGURE 2-3 Operating system security environment

23

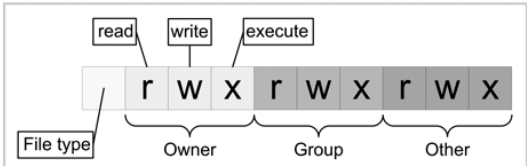
Files

- Common threats:
  - File permission
  - File sharing
- Files must be protected from unauthorized reading and writing actions
- Data resides in files; protecting files; protects data

24

File Permissions

- Read, write, and execute privileges
- In Windows:
  - Change permission on the Security tab on a file's Properties box
  - Allow indicates grant; Deny indicates revoke
- In UNIX/Linux
  - Three permission settings: owner; group to which owner belongs; all other users. Each setting consist of rwx
  - CHMOD command used to change file permissions
  - Ex: chmod 644 test



25

File Permissions (continued)

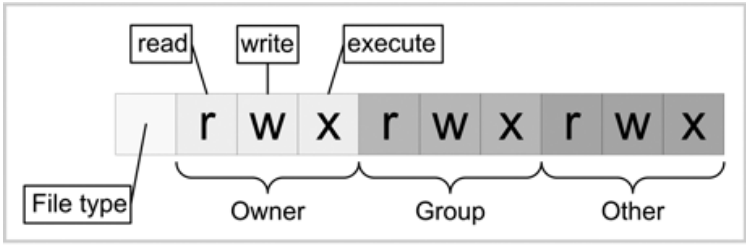


FIGURE 2-5 UNIX file permissions

```
$ chmod  644  mail_list
```

26

File Transfer

- FTP (File Transfer Protocol):
  - Internet service for transferring files from one computer to another
  - Transmits usernames and passwords in plaintext
  - Root account cannot be used with FTP
  - Anonymous FTP: ability to log on to the FTP server without being authenticated
- Best practices:
  - Use Secure FTP utility if possible
  - Make two FTP directories:
    - One for uploads with write permissions only
    - One for downloads with read permissions only
  - Use specific accounts with limited permissions
  - Log and scan FTP activities
  - Allow only authorized operations

27

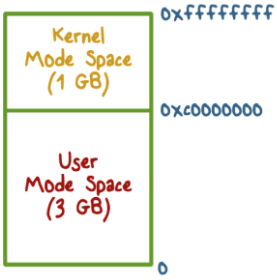
Sharing Files

- Naturally leads to security risks and threats
- Peer-to-peer programs: allow users to share files over the Internet
- Reasons for blocking file sharing:
  - Malicious code
  - Adware and spyware
  - Privacy and confidentiality
  - Pornography
  - Copyright issues

28

Memory

- Hardware memory available on the system can be corrupted by badly written software
- Can harm data integrity
- Two options:
  - Stop using the program
  - Apply a patch (service pack) to fix it
- Safe and efficient memory usage:
  - Memory division User space & Kernel space



- Prevent one program or user from interfering with another user program's memory space:
  - Segmentation
  - Paging

29

Services

- Services
  - Main component of operating system security environment
  - Used to gain access to the OS and its features
- Include
  - User authentication
  - Remote access
  - Administration tasks
  - Password policies

30

Authentication Methods

- Authentication: Verifies user access to the operating system
- Physical authentication:
  - Allows physical entrance to company property
  - Magnetic cards and biometric measures
- Digital authentication: verifies user identity by digital means
- Digital certificates: identifies and verifies holder of certificate
- Digital token (security token):
  - Small electronic device
  - Displays a number unique to the token holder;
  - Uses a different password each time
- Digital card: Also known as a security card or smart card
  - Similar to a credit card; uses an electronic circuit instead of a magnetic strip
  - Stores user identification information
- Kerberos:
  - Developed by MIT
  - Uses tickets for authentication purposes

31

Authorization

- Process that decides whether users are permitted to perform the functions they request
- Authorization is not performed until the user is authenticated
- Deals with privileges and rights

32

User Administration

- Create user accounts
- Set password policies
- Grant privileges to users
- Best practices:
  - Use a consistent naming convention
  - Always provide a password to an account and force the user to change it at the first logon
  - Protect passwords
  - Do not use default passwords

33

User Administration (continued)

- Best practices (continued):
  - Create a specific file system for users
  - Educate users on how to select a password
  - Lock non-used accounts
  - Grant privileges on a per host basis
  - Do not grant privileges to all machines
  - Use ssh, scp, and Secure FTP
  - Isolate a system after a compromise
  - Perform random auditing procedures

34

E-mail Security

- Tool must widely used by public
- May be the tool must frequently used by hackers:
  - Viruses; Worms; Spam; Others
- Used to send private and confidential data as well as offensive material
- Used by employees to communicate with:
  - Clients
  - Colleagues
  - Friends
- Recommendations:
  - Do not configure e-mail server on the same machine where sensitive data resides
  - Do not disclose technical details about the e-mail server

35

Vulnerabilities of OS

- Top vulnerabilities to Windows systems:
  - Internet Information Services (IIS)
  - Microsoft SQL Server (MSSQL)
  - Windows Authentication
  - Internet Explorer (IE)
  - Windows Remote Access Services
  - Microsoft Data Access Components (MDAC)
  - Windows Scripting Host (WSH)
  - Microsoft Outlook and Outlook Express
  - Windows Peer-to-Peer File Sharing (P2P)
  - Simple Network Management Protocol (SNMP)

National Vulnerability Database:

http://nvd.nist.gov/

36

Vulnerabilities of OS

- Top vulnerabilities to UNIX systems:
  - BIND Domain Name System
  - Remote Procedure Calls (RPC)
  - Apache Web Server
  - General UNIX authentication accounts with no passwords or weak passwords
  - Clear text services
  - Sendmail
  - Simple Network Management Protocol (SNMP)
  - Secure Shell (SSH)
  - Misconfiguration of Enterprise Services NIS/NFS
  - Open Secure Sockets Layer (SSL)

National Vulnerability Database:

http://nvd.nist.gov/

Secure an operating system

- Basic steps
  - Install and patch the operating system.
  - Harden and configure the OS to adequately address the identified security needs of the system by:
    - Removing unnecessary services, applications, and protocols.
    - Configuring users, groups, and permissions.
    - Configuring resource controls.
  - Install and configure additional security controls, such as anti-virus, hostbased firewalls, and IDS, if needed.
  - Test the security of the basic OS to ensure that the steps taken adequately address its security needs.

Operating Systems Hardening

- Hardening:
  - attempting to make OS bulletproof.
  - Ideally - leave OS exposed to the general public on the Internet without any other form of protection.
  - A hardened system should serve only one purpose--it's a Web server or DNS or Exchange server, and nothing else. These systems need too many functions to be properly hardened.

Harden Windows - minimum

- **Disable all unnecessary services.**
  - determine which services can be disabled.
    - Remote Procedure Call (RPC) service.
    - little documentation exists to identify what services a given purpose will require.
    - knowing which services are required and which can be disabled is largely a matter of trial and error.
- **Remove all unnecessary executables and registry entries.**
  - Forgetting to remove unneeded executables and registry entries might allow an attacker to invoke something that had previously been disabled.
- **Apply appropriately restrictive permissions to files, services, end points and registry entries.**
  - Inappropriate permissions could give an attacker an opening.
  - The ability to launch CMD.EXE as "LocalSystem," for example, is a classic backdoor.

Harden Windows for maximum security

- **Adjusting retransmission of SYN-ACKS**. This makes connection responses time out more quickly during a SYN flood.
- **Determining how many times TCP retransmits** an unacknowledged data segment on an existing connection. TCP retransmits data segments until they are acknowledged or until this value expires.
- **Disabling ICMP Router Discovery Protocol** (IRDP) where an attacker may remotely add default route entries on a remote system.
- **Disabling these services**: Telnet, Universal Plug and Play Device Host, IIS, Disable Guest accounts
- **Use the Local Security Policy**
- **Disable File and Print Sharing**.
- **Disable Remote Assistance and Remote Desktop**
- **Use NTFS File system**.
- **Disable auto-logins**.

Harden Linux

- Encrypt Data Communication
- Avoid Using FTP, Telnet, And Rlogin / Rsh Services
- Minimize Software to Minimize Vulnerability
- One Network Service Per System or VM Instance
- Keep Linux Kernel and Software Up to Date
- Use Linux Security Extensions
- SELinux
- Password: Policy, Aging, Empty
- Login:
  - Locking User Accounts After Login Failures
  - Make Sure No Non-Root Accounts Have UID Set To 0
  - Disable root Login

Harden Linux

- Disable Unwanted Services
- Find Listening Network Ports
- Configure Iptables and TCPWrappers
- Linux Kernel /etc/sysctl.conf Hardening
- Separate Disk Partitions
- Disk Quotas
- Turn Off IPv6
- Disable Unwanted SUID and SGID Binaries
- Logging and Auditing
- Secure OpenSSH Server
- Install And Use Intrusion Detection System
- Disable USB/firewire/thunderbolt devices

OS Attack: Privilege escalation

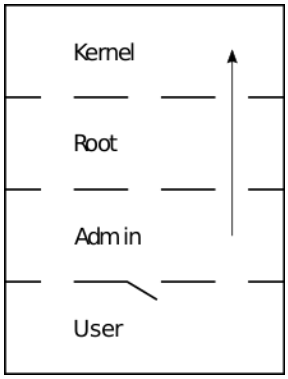
- Privilege escalation :
  - the act of exploiting a bug,
  - design flaw or configuration oversight in an OS or software application to gain elevated access to resources that are normally protected from an application or user.
  - => The result is that an application with more privileges than intended by the application developer or system administrator can perform unauthorized actions.
- **2 types:**
  - **Vertical privilege escalation**, also known as *privilege elevation*, where a lower privilege user or application accesses functions or content reserved for higher privilege users or applications

    (e.g. Internet Banking users can access site administrative functions or the password for a smartphone can be bypassed.)
  - **Horizontal privilege escalation**, where a normal user accesses functions or content reserved for other normal users

    (e.g. Internet Banking User A accesses the Internet bank account of User B)

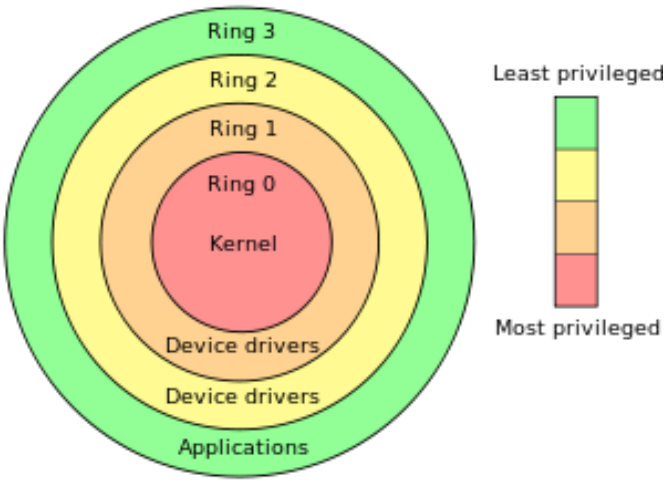
OS Attack: Privilege escalation

- a rootkit gaining access to the kernel, and the little gate represents normal privilege elevation



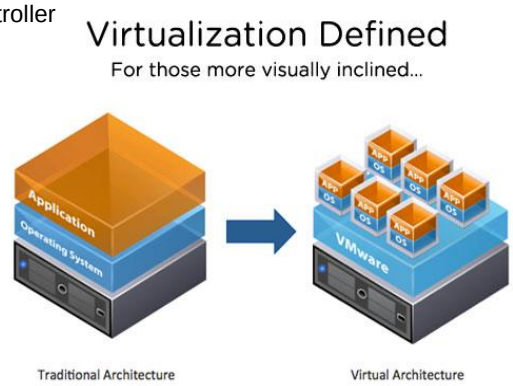
Protected mode

- Privilege rings for the x86 available in protected mode



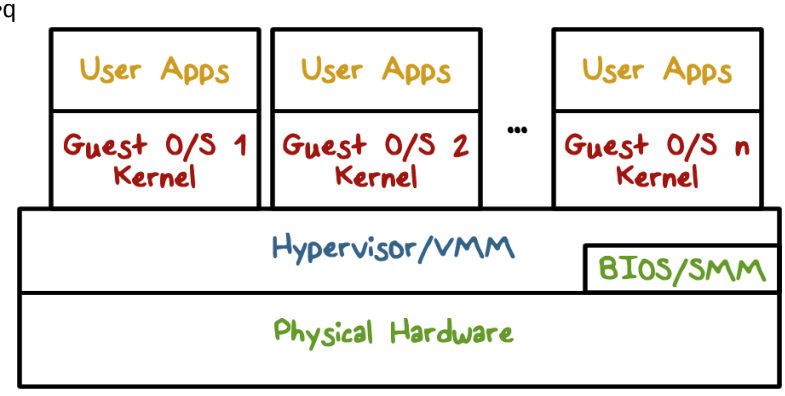
Virtualization Security - Introduction

• A VM is a software implementation of a machine that execute programs like a physical machine
• A VM can support individual processes or a complete system depending on the abstraction level where virtualization occurs.
• Virtualization – a technology that allows running two or more OS side by side on one PC or embedded controller



VM Architecture

•q

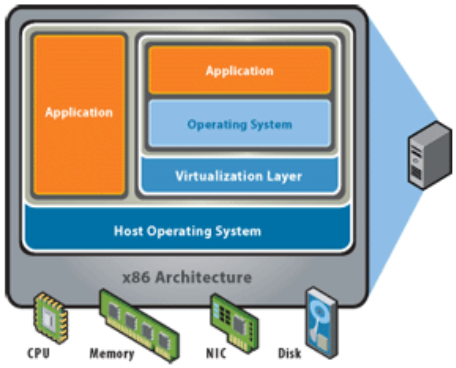


VM Architecture

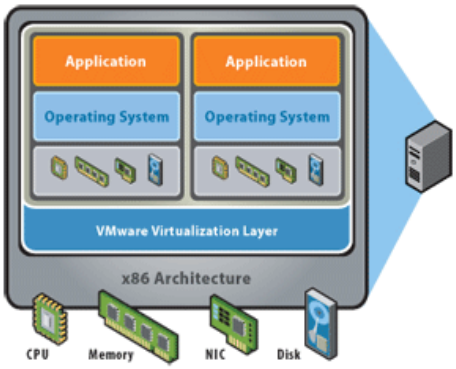
More security implications because of the reliance on the underlying OS, used in VMware and MS Virtual PC

VM is installed that communicates directly with system hardware rather than relying on a host OS

Hosted



Bare - Metal

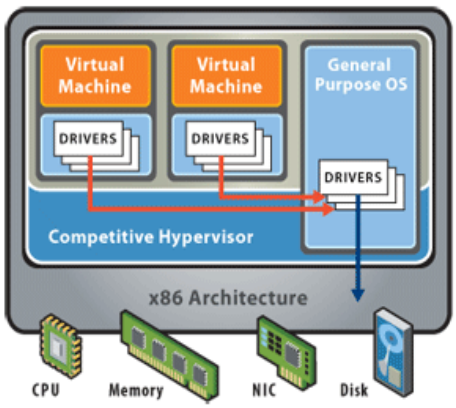


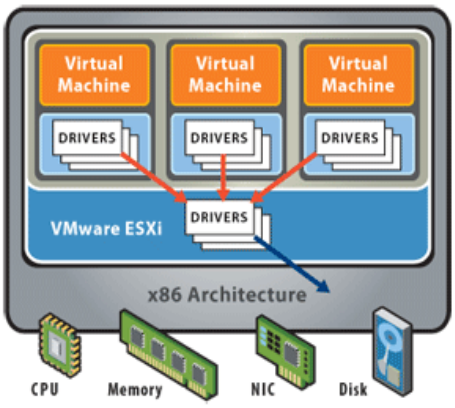
VM Architecture

- **Thin Virtualization – reduced size, independence**

=> Get Strong Security in a Small Package

  - the attack surface much smaller, and reduces the potential for vulnerabilities
  - far fewer interfaces to exploit and less malware threats





VM Architecture

- Security Concepts in Architecture
  - Extended computing stack
  - Guest isolation
  - Host Visibility from the Guest
  - Virtualized interfaces
  - Management interfaces
  - Greater co-location of data and assets on one box

VM Security Benefits

- Abstraction and Isolation
- Better Forensics and Faster Recovery After an Attack
- Patching is Safer and More Effective
- More Cost Effective Security Devices
- Future: Leveraging Virtualization to Provide Better Security

VM Security Issues

- VM Sprawl
- Mobility
- Hypervisor Intrusion
- Hypervisor Modification
- Communication
- Denial of Service

VM Security Issues

Issue	Hosted	Bare-Metal
Vulnerability of the underlying OS	susceptible to all the vulnerabilities and attacks that are prevalent on such systems.	a much smaller attack surface
Sharing of files and data between the guest and the host	vulnerable to data leakage and malicious code intrusion.	there is no mechanism share user information between virtual machines and their host.
Resource allocation	They are at the mercy of the host OS and other applications.	No single virtual machine can use all the resources or crash the system.
Target Usage	- is targeted for environments where the guest virtual machines can be trusted. (software development, testing, demonstration, and trouble-shooting.)	can potentially be exposed to malicious users and network traffic. Strong isolation and strict separation of management greatly reduce any risk of harmful activity going beyond the boundaries of the virtual machine.

VM Security Concerns

- **Managing oversight and responsibility**
- **Patching and maintenance**
- **Visibility and compliance**
- **VM sprawl**
- **Managing Virtual Appliances**

Summary

- Introduction To OS and OS Security
- System Security Planning
- The Components of an OS Security Environment
- Vulnerabilities of OS
- Secure an operating system
- Operating Systems Hardening
  - Linux/Unix Security
  - Windows Security
- Virtualization Security

Q & A