

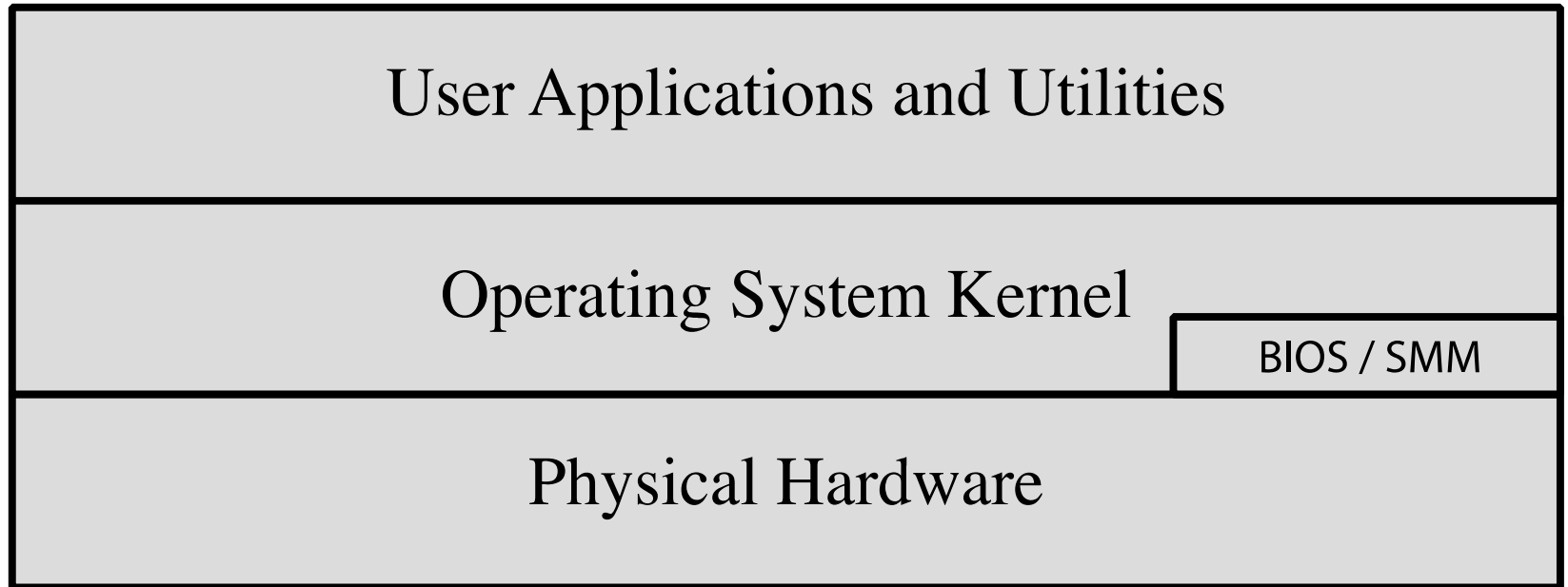
# Computer Security: Principles and Practice

Fourth Edition

By: William Stallings and Lawrie Brown

# Chapter 12

Operating System Security



**Figure 12.1 Operating System Security Layers**

# استراتژی ها

- در سال ۲۰۱۰، ASD فهرستی از "۳۵ راهکار برتر کاهش تهدیدات" منتشر کرد. بیش از ۸۵٪ از نفوذهای سایبری هدفمندی که ASD در سال ۲۰۰۹ بررسی کرد، قابل پیشگیری بودند.
- چهار راهکار برتر برای پیشگیری عبارت‌اند از:
- تهیه لیست سفید برنامه‌های مجاز
- به‌روزرسانی برنامه‌های شخص ثالث و رفع آسیب‌پذیری‌های سیستم‌عامل
- محدود کردن دسترسی‌های مدیریتی
- ایجاد یک سیستم دفاعی چندلایه
- این راهکارها تا حد زیادی با "۲۰ کنترل حیاتی" که توسط وزارت امنیت داخلی ایالات متحده، آژانس امنیت ملی، وزارت انرژی، مؤسسه SANS و سایر نهادهای آمریکایی توسعه داده شده است، هماهنگ هستند.

# امنیت سیستم عامل

- امکان دارد یک سیستم در حین فرآیند نصب، قبل از اینکه بتواند آخرین به روزرسانی‌ها را نصب کند، مورد نفوذ قرار گیرد.
- بنابراین، ساخت و استقرار یک سیستم باید یک فرآیند برنامه‌ریزی شده باشد که برای مقابله با این تهدید طراحی شده است.
- این فرآیند باید شامل موارد زیر باشد:
- ارزیابی ریسک‌ها و برنامه‌ریزی برای استقرار سیستم
- ایمن‌سازی سیستم عامل زیرساختی و سپس برنامه‌های کلیدی
- تضمین امنیت هرگونه محتوای حساس و حیاتی
- استفاده از مکانیزم‌های مناسب برای محافظت از شبکه
- اجرای فرآیندهای مناسب جهت حفظ امنیت سیستم

# برنامه ریزی امنیت سیستم

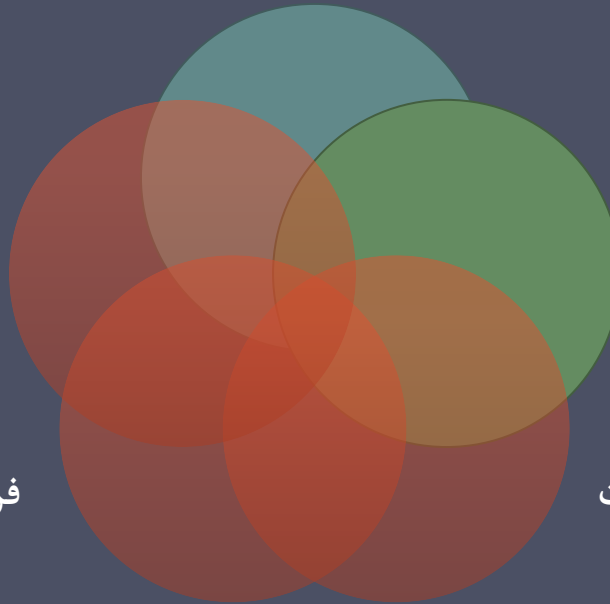
اولین قدم در مستقر کردن یک سیستم جدید برنامه ریزی است.

طرح امنیتی باید افراد مناسب و آموزش‌های لازم برای نصب و مدیریت سیستم را مشخص کند.

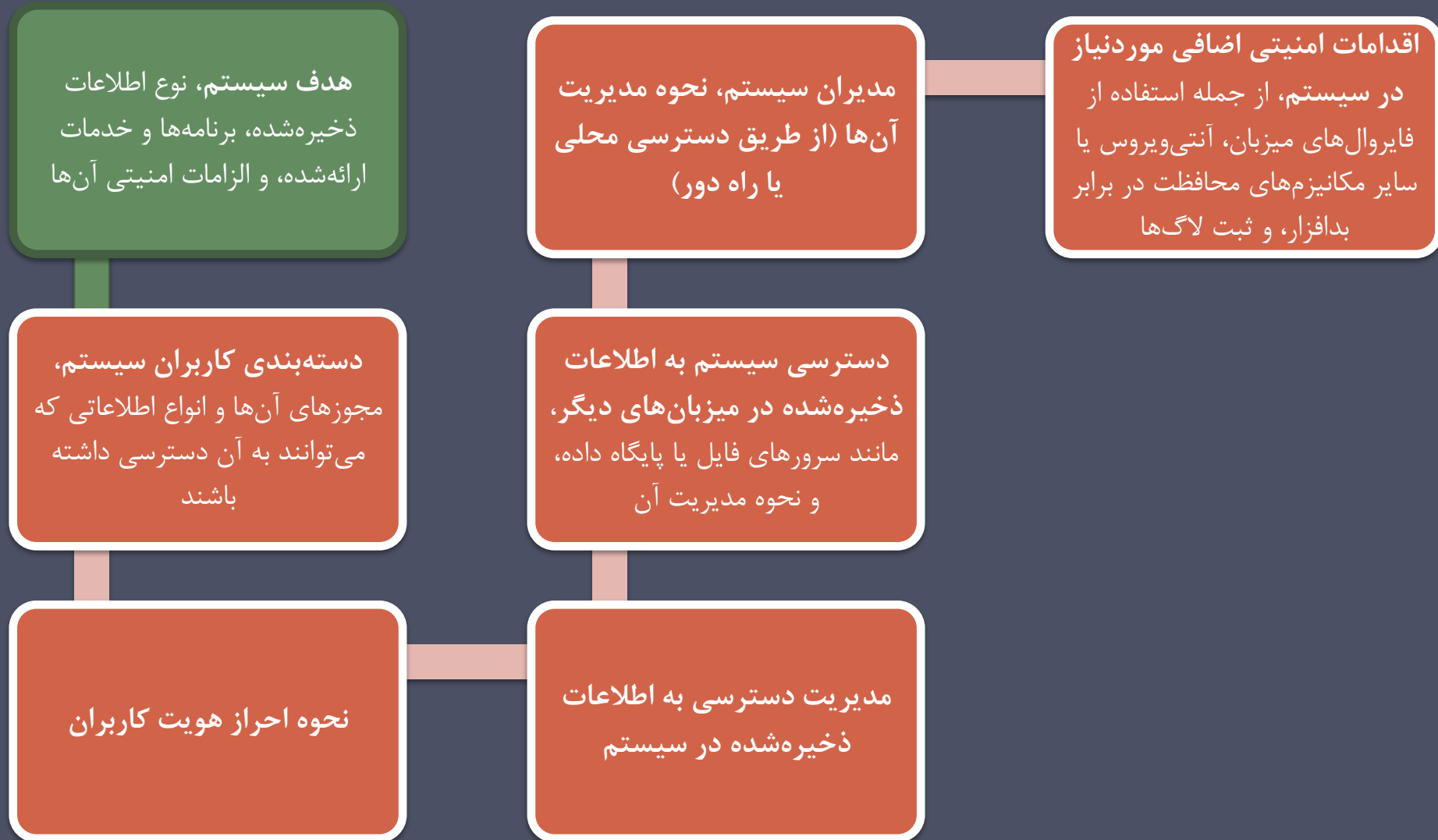
برنامه‌ریزی باید شامل یک ارزیابی جامع امنیتی از سازمان باشد.

فرآیند برنامه‌ریزی باید الزامات امنیتی برای سیستم، برنامه‌ها، داده‌ها و کاربران را تعیین کند.

هدف این است که امنیت را به حداکثر رسانده و هزینه‌ها را به حداقل برسانیم.



# فرآیند برنامه‌ریزی امنیت سیستم

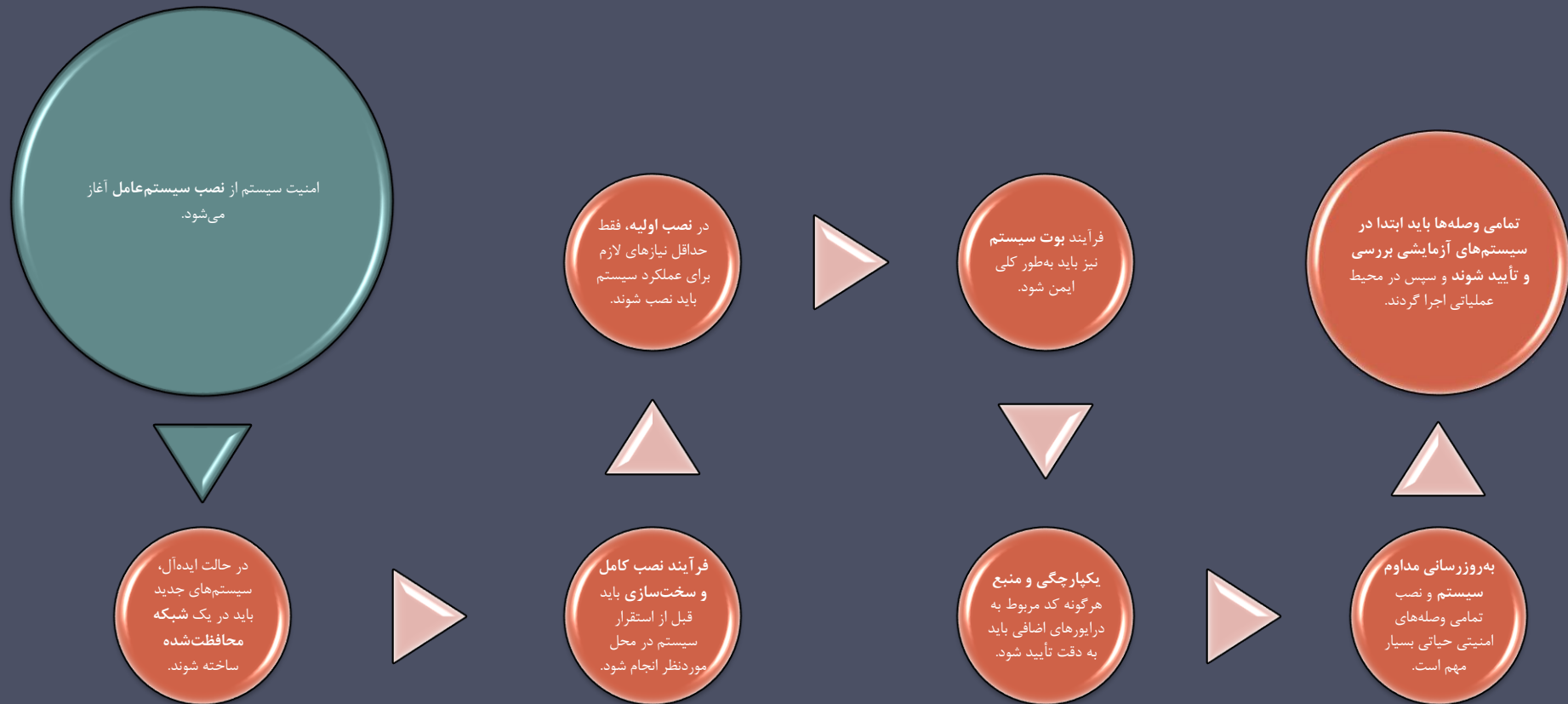


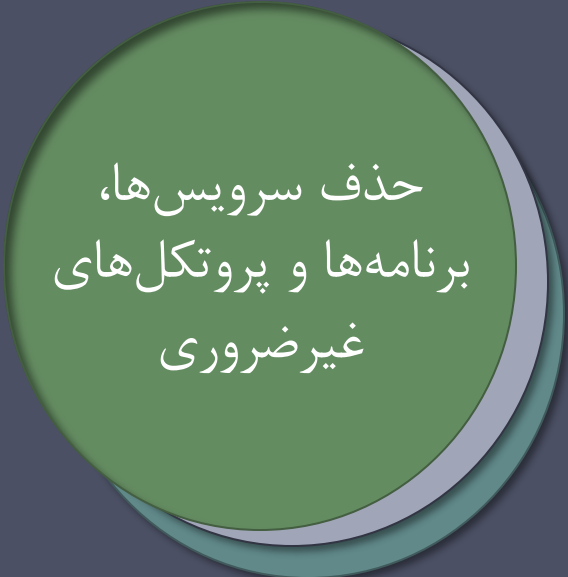
# ایمن سازی سیستم عامل

- اولین گام حیاتی در ایمن سازی یک سیستم، تأمین امنیت سیستم عامل پایه است.
- مراحل اصلی:
- نصب و به روز رسانی سیستم عامل
- سخت سازی و پیکربندی سیستم عامل برای پاسخگویی مناسب به نیازهای امنیتی شناسایی شده، شامل:
  - حذف سرویس ها، برنامه ها و پروتکل های غیر ضروری
  - پیکربندی کاربران، گروه ها و مجوزهای دسترسی
  - تنظیم کنترل های منابع
- نصب و پیکربندی کنترل های امنیتی اضافی، مانند:
  - آنتی ویروس
  - فایروال مبتنی بر میزبان
  - سیستم تشخیص نفوذ IDS
- آزمایش امنیت سیستم عامل پایه برای اطمینان از اینکه اقدامات انجام شده به طور مناسب نیازهای امنیتی را برآورده می کنند.



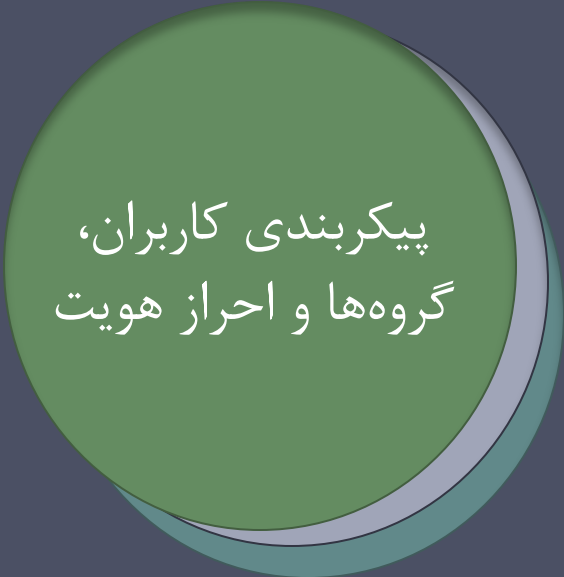
# نصب اولیه و به روز رسانی





## حذف سرویس‌ها، برنامه‌ها و پروتکل‌های غیرضروری

- هنگام انجام نصب اولیه، نباید از تنظیمات پیش‌فرض ارائه‌شده استفاده شود.
- تنظیمات پیش‌فرض به‌گونه‌ای تنظیم شده‌اند که بیشتر به راحتی استفاده و عملکرد سیستم توجه دارند تا امنیت.
- اگر بسته‌های اضافی بعداً نیاز شدند، می‌توان آن‌ها را زمانی که لازم است نصب کرد.
- اگر تعداد بسته‌های نرم‌افزاری کمتری برای اجرا در دسترس باشد، ریسک کاهش می‌یابد.
- فرآیند برنامه‌ریزی سیستم باید شناسایی کند که برای یک سیستم خاص، چه نرم‌افزارهایی به‌طور واقعی مورد نیاز هستند.



## پیکربندی کاربران، گروه‌ها و احراز هویت

- تمام کاربران با دسترسی به یک سیستم، دسترسی یکسان به تمام داده‌ها و منابع آن سیستم نخواهند داشت.
- امتیازات بالا باید محدود به کاربرانی باشند که به آن‌ها نیاز دارند، و تنها زمانی که برای انجام یک وظیفه خاص ضروری است، این امتیازات باید در دسترس قرار گیرند.

- فرآیند برنامه‌ریزی سیستم باید موارد زیر را در نظر بگیرد:
- دسته‌بندی‌های کاربران در سیستم
- امتیازات آن‌ها
- نوع اطلاعاتی که می‌توانند به آن دسترسی داشته باشند
- نحوه و مکانی که کاربران تعریف و احراز هویت می‌شوند
- حساب‌های پیش‌فرض که به عنوان بخشی از نصب سیستم گنجانده شده‌اند باید ایمن شوند
- حساب‌هایی که نیاز نیستند باید حذف یا غیرفعال شوند
- سیاست‌هایی که برای تنظیم اعتبارنامه‌های احراز هویت اعمال می‌شود

## پیکربندی کنترل‌های منابع

- پس از تعریف کاربران و گروه‌ها، می‌توان مجوزهای مناسب را برای داده‌ها و منابع تنظیم کرد.
- بسیاری از راهنماهای سخت‌سازی امنیتی فهرستی از تغییرات پیشنهادی برای پیکربندی دسترسی پیش‌فرض ارائه می‌دهند.

## نصب کنترل‌های امنیتی اضافی

- امنیت بیشتر را می‌توان با نصب و پیکربندی ابزارهای امنیتی اضافی فراهم کرد:
- نرم‌افزار آنتی‌ویروس
- فایروال‌های مبتنی بر میزبان
- سیستم تشخیص نفوذ یا سیستم پیشگیری از نفوذ
- فهرست سفید برنامه‌ها



## Test the System Security

- Final step in the process of initially securing the base operating system is security testing
- Goal:
  - Ensure the previous security configuration steps are correctly implemented
  - Identify any possible vulnerabilities
- Checklists are included in security hardening guides
- There are programs specifically designed to:
  - Review a system to ensure that a system meets the basic security requirements
  - Scan for known vulnerabilities and poor configuration practices
- Should be done following the initial hardening of the system
- Repeated periodically as part of the security maintenance process

# Application Configuration

- May include:
  - Creating and specifying appropriate data storage areas for application
  - Making appropriate changes to the application or service default configuration details
- Some applications or services may include:
  - Default data
  - Scripts
  - User accounts
- Of particular concern with remotely accessed services such as Web and file transfer services
  - Risk from this form of attack is reduced by ensuring that most of the files can only be read, but not written, by the server

# Encryption Technology

Is a key enabling technology that may be used to secure data both in transit and when stored

Must be configured and appropriate cryptographic keys created, signed, and secured

If secure network services are provided using TLS or IPsec suitable public and private keys must be generated for each of them

If secure network services are provided using SSH, appropriate server and client keys must be created

Cryptographic file systems are another use of encryption

# Security Maintenance

- Process of maintaining security is continuous
- Security maintenance includes:
  - Monitoring and analyzing logging information
  - Performing regular backups
  - Recovering from security compromises
  - Regularly testing system security
  - Using appropriate software maintenance processes to patch and update all critical software, and to monitor and revise configuration as needed



# Logging

Can only inform you about bad things that have already happened

In the event of a system breach or failure, system administrators can more quickly identify what happened

Key is to ensure you capture the correct data and then appropriately monitor and analyze this data

Information can be generated by the system, network and applications

Range of data acquired should be determined during the system planning stage

Generates significant volumes of information and it is important that sufficient space is allocated for them

Automated analysis is preferred

# Data Backup and Archive

Performing regular backups of data is a critical control that assists with maintaining the integrity of the system and user data

May be legal or operational requirements for the retention of data

## Backup

The process of making copies of data at regular intervals

## Archive

The process of retaining copies of data over extended periods of time in order to meet legal and operational requirements to access past data

Needs and policy relating to backup and archive should be determined during the system planning stage

Kept online or offline

Stored locally or transported to a remote site

- Trade-offs include ease of implementation and cost versus greater security and robustness against different threats

# Linux/Unix Security

- Patch management
  - Keeping security patches up to date is a widely recognized and critical control for maintaining security
- Application and service configuration
  - Most commonly implemented using separate text files for each application and service
  - Generally located either in the /etc directory or in the installation tree for a specific application
  - Individual user configurations that can override the system defaults are located in hidden “dot” files in each user’s home directory
  - Most important changes needed to improve system security are to disable services and applications that are not required

# Linux/Unix Security

- Users, groups, and permissions
  - Access is specified as granting read, write, and execute permissions to each of owner, group, and others for each resource
  - Guides recommend changing the access permissions for critical directories and files
  - Local exploit
    - Software vulnerability that can be exploited by an attacker to gain elevated privileges
  - Remote exploit
    - Software vulnerability in a network server that could be triggered by a remote attacker

# Linux/Unix Security

## Remote access controls

- Several host firewall programs may be used
- Most systems provide an administrative utility to select which services will be permitted to access the system

## Logging and log rotation

- Should not assume that the default setting is necessarily appropriate

# Linux/Unix Security

- chroot jail
  - Restricts the server's view of the file system to just a specified portion
  - Uses chroot system call to confine a process by mapping the root of the filesystem to some other directory
  - File directories outside the chroot jail aren't visible or reachable
  - Main disadvantage is added complexity

# Windows Security

## Patch management

- “Windows Update” and “Windows Server Update Service” assist with regular maintenance and should be used
- Third party applications also provide automatic update support

## Users administration and access controls

- Systems implement discretionary access controls resources
- Vista and later systems include mandatory integrity controls
- Objects are labeled as being of low, medium, high, or system integrity level
- System ensures the subject's integrity is equal or higher than the object's level
- Implements a form of the Biba Integrity model

# Windows Security

## Users Administration and Access Controls

Windows systems also define privileges

- System wide and granted to user accounts

Combination of share and NTFS permissions may be used to provide additional security and granularity when accessing files on a shared resource

User Account Control (UAC)

- Provided in Vista and later systems
- Assists with ensuring users with administrative rights only use them when required, otherwise accesses the system as a normal user

Low Privilege Service Accounts

- Used for long-lived service processes such as file, print, and DNS services



# Windows Security

## Application and service configuration

- Much of the configuration information is centralized in the Registry
  - Forms a database of keys and values that may be queried and interpreted by applications
- Registry keys can be directly modified using the “Registry Editor”
  - More useful for making bulk changes

# Windows Security

## Other security controls

- Essential that anti-virus, anti-spyware, personal firewall, and other malware and attack detection and handling software packages are installed and configured
- Current generation Windows systems include basic firewall and malware countermeasure capabilities
- Important to ensure the set of products in use are compatible

## Windows systems also support a range of cryptographic functions:

- Encrypting files and directories using the Encrypting File System (EFS)
- Full-disk encryption with AES using BitLocker

## “Microsoft Baseline Security Analyzer”

- Free, easy to use tool that checks for compliance with Microsoft’s security recommendations

# Virtualization

- A technology that provides an abstraction of the resources used by some software which runs in a simulated environment called a virtual machine (VM)
- Benefits include better efficiency in the use of the physical system resources
- Provides support for multiple distinct operating systems and associated applications on one physical system
- Raises additional security concerns

# Hypervisor

- Software that sits between the hardware and the VMs
- Acts as a resource broker
- It allows multiple VMs to safely coexist on a single physical server host and share that host's resources
- Virtualizing software provides abstraction of all physical resources and thus enables multiple computing stacks, called virtual machines, to be run on a single physical host
- Each VM includes an OS, called the guest OS
  - This OS may be the same as the host OS, if present, or a different one

# Hypervisor Functions

The principal functions performed by a hypervisor are:

- Execution management of VMs
- Devices emulation and access control
- Execution of privileged operations by hypervisor for guest VMs
- Management of VMs (also called VM lifecycle management)
- Administration of hypervisor platform and hypervisor software

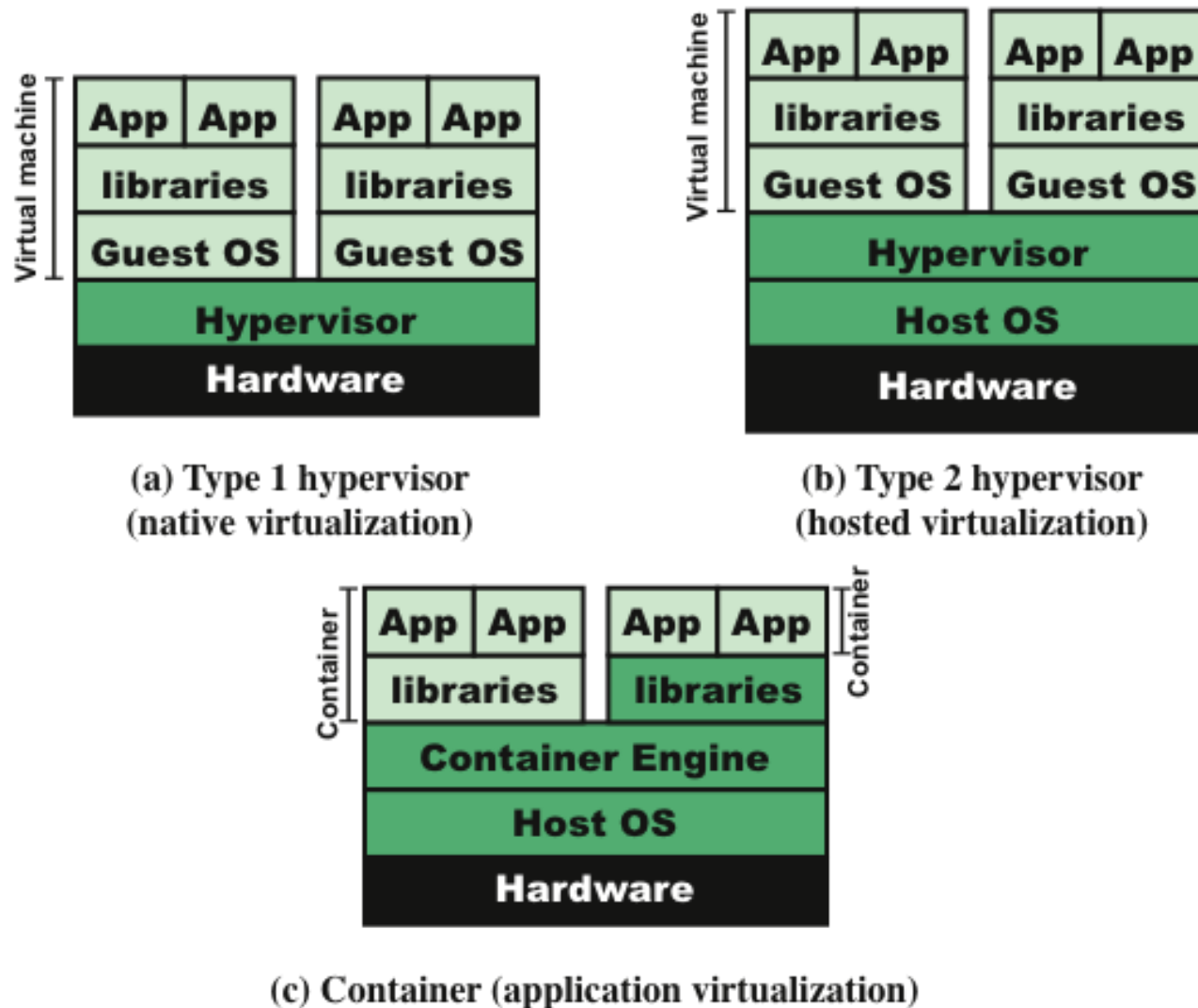


Figure 12.2 Comparison of Virtual Machines and Containers

# Virtualized Systems

- In virtualized systems, the available hardware resources must be appropriately shared among the various guest OS's
- These include CPU, memory, disk, network, and other attached devices
- CPU and memory are generally partitioned between these, and scheduled as required
- Disk storage may be partitioned, with each guest having exclusive use of some disk resources
- Alternatively, a “virtual disk” may be created for each guest, which appears to it as a physical disk with a full file-system, but is viewed externally as a single “disk image” file on the underlying file-system
- Attached devices such as optical disks, or USB devices are generally allocated to a single guest OS at a time

# Software Defined Networks (SDNs)

SDNs enable network segments to logically span multiple servers within and between data centers, while using the same underlying physical network

There are several possible approaches to providing SDNs, including the use of overlay networks

- These abstract all layer 2 and 3 addresses from the underlying physical network into whatever logical network structure is required
- This structure can be easily changed and extended as needed
- The IETF standard DOVE (Distributed Overlay Virtual Network) which uses VXLAN (Virtual Extended Local Area Network) can be used to implement such an overlay network
- With this flexible structure, it is possible to locate virtual servers, virtual IDS, and virtual firewalls anywhere within the network as required



# Containers

- A recent approach to virtualization is known as *container virtualization* or *application virtualization*
- In this approach, software known as a *virtualization container*, runs on top of the host OS kernel and provides an isolated execution environment for applications
- Unlike hypervisor-based VMs, containers do not aim to emulate physical servers
- All containerized applications on a host share a common OS kernel
- For containers, only a small container engine is required as support for the containers
- Containerization sits in between the OS and applications and incurs lower overhead, but potentially introduces greater security vulnerabilities

# Virtualization Security Issues

- Security concerns include:
  - Guest OS isolation
    - Ensuring that programs executing within a guest OS may only access and use the resources allocated to it
  - Guest OS monitoring by the hypervisor
    - Which has privileged access to the programs and data in each guest OS
  - Virtualized environment security
    - Particularly image and snapshot management which attackers may attempt to view or modify

# Securing Virtualization Systems

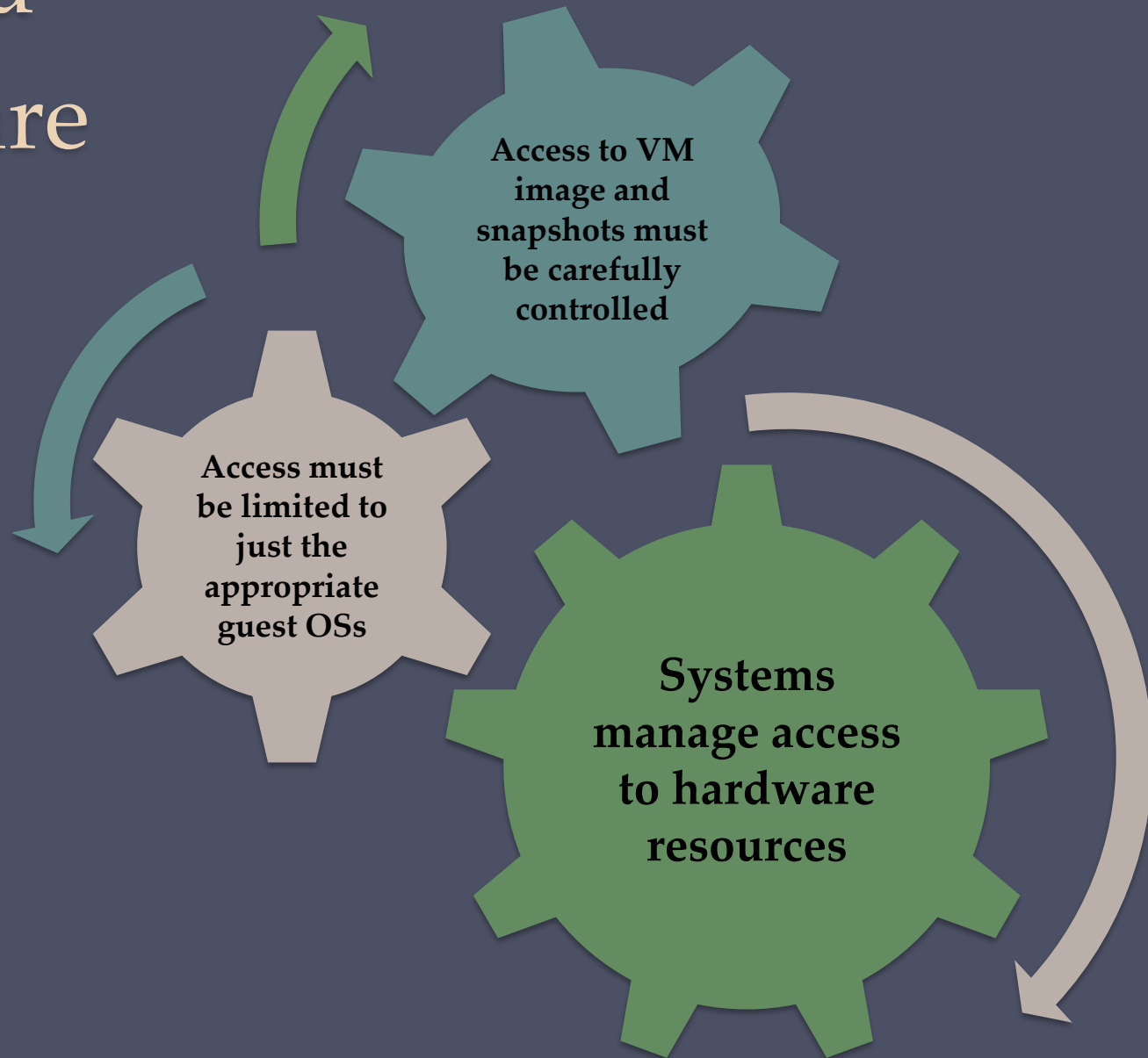
**Organizations  
using  
virtualization  
should:**

- **Carefully plan the security of the virtualized system**
- **Secure all elements of a full virtualization solution and maintain their security**
- **Ensure that the hypervisor is properly secured**
- **Restrict and protect administrator access to the virtualization solution**

# Hypervisor Security

- Should be
  - Secured using a process similar to securing an operating system
  - Installed in an isolated environment
  - Configured so that it is updated automatically
  - Monitored for any signs of compromise
  - Accessed only by authorized administration
- May support both local and remote administration so must be configured appropriately
- Remote administration access should be considered and secured in the design of any network firewall and IDS capability in use
- Ideally administration traffic should use a separate network with very limited access provided from outside the organization

# Virtualized Infrastructure Security



# Virtual Firewall

Provides firewall capabilities for the network traffic flowing between systems hosted in a virtualized or cloud environment that does not require this traffic to be routed out to a physically separate network supporting traditional firewall services

## VM Bastion Host

Where a separate VM is used as a bastion host supporting the same firewall systems and services that could be configured to run on a physically separate bastion, including possibly IDS and IPS services

## VM Host-Based Firewall

Where host-based firewall capabilities provided by the Guest OS running on the VM are configured to secure that host in the same manner as used in physically separate systems

## Hypervisor Firewall

Where firewall capabilities are provided directly by the hypervisor

# Summary

- Introduction to operating system security
- System security planning
- Operating systems hardening
  - Operating system installation: initial setup and patching
  - Remove unnecessary services, applications and protocols
  - Configure users, groups, and authentications
  - Configure resource controls
  - Install additional security controls
  - Test the system security
- Application security
  - Application configuration
  - Encryption technology
- Security maintenance
  - Logging
  - Data backup and archive
- Linux/Unix security
  - Patch management
  - Application and service configuration
  - Users, groups, and permissions
  - Remote access controls
  - Logging and log rotation
  - Application security using a chroot jail
  - Security testing
- Windows security
  - Patch management
  - Users administration and access controls
  - Application and service configuration
  - Other security controls
  - Security testing
- Virtualization security
  - Virtualization alternatives
  - Virtualization security issues
  - Securing virtualization systems