Comp 310
Computer Systems and Organization

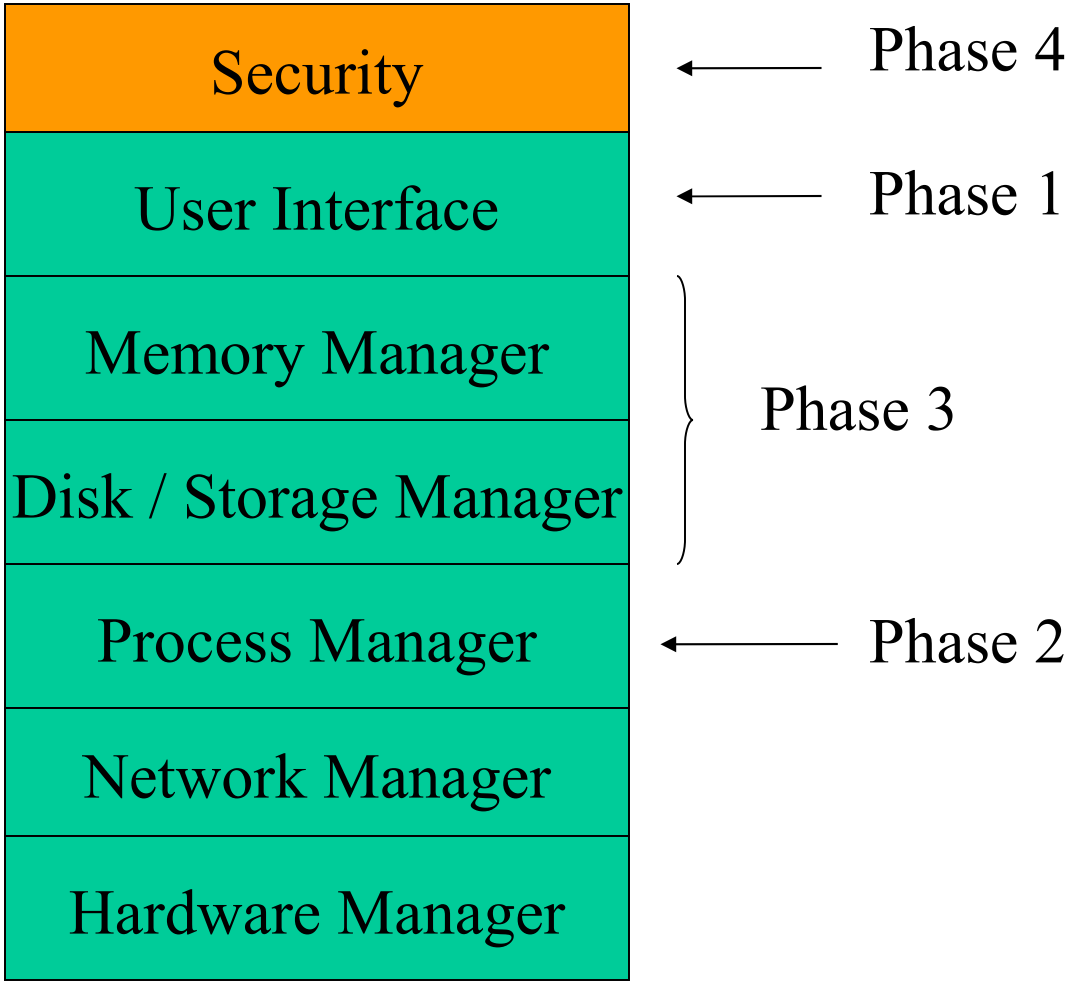
Lecture #24

Safety Systems

Prof. Joseph Vybihal

Basic OS Architecture

(Course Table of Contents)



Security

User Interface

Memory Manager

Disk / Storage Manager

Process Manager

Network Manager

Hardware Manager

Phase 4

Phase 1

Phase 3

Phase 2

Announcements

- Final Exam Dec 9, 2PM
- **Course Evaluations**
  - Online (Minerva / Web CT)
- Office Hours
  - TBD
- Tutorials … TBA

Topics on Exam

- Memory Management
- Virtual Memory
- File Systems
- Disk Drives
- Protection Issues
- Security Issues

Concepts on Exam

- Pages, frames and backing store
- Performance issues
- NO Unix, some C programming possible
- Domains and access rights
- Implementation methods
- Know your definitions and acronyms
- Know your probabilities

Final Exam Format

- 5 questions
- Mostly problems, little programming
- No definitions, but know your definitions
- Even though no questions before midterm you need to know those concepts and techniques (e.g. semaphores)
- Note, on course outline their are 6 topics and we have 5 questions…
- Rules
  - Closed book, No calculators
  - Answer questions in booklet
  - Part marks are given

What is covered?

- Everything after the midterm
  - Memory management, Chapter 9
  - Virtual Memory. Chapter 10
  - File Systems, Chapters 11 & 12
  - Disk Drives, Chapters 13 & 14
  - Protection Issues, Chapter 18
  - Security Issues, Chapter 19
- Things you should know but not directly tested:
  - Process Management
  - Deadlocks
  - Basic OS & Motherboard Architecture

Types of questions

- Describe and draw
- No programming questions but you may be asked to produce pseudo-code or to describe an algorithm in words
- Given a situation provide a solution
- Compare features
  - – between real things
  - – between imaginary and real things
- Analyze the complexity

Questions?

Problems

- Least recent used page replacement – min/dirty swaps
- Backup/security rules for an ATM with power out problems (make sure correct deposit)
- Security: process send msg(s) over internet (lost, made it)
- Worst / average case: seek SCAN/C-SCAN

Research/Project Opportunities

- Joseph Vybihal
  - AI: The Prometheus project (simulation env.)
  - Virtual Office: Internet OS and work environment

- Muthucumaru Maheswaran
  - Parallel & Distributed Systems simulator
    - Network and OS

Part 1

About Security

Security

- When all the resources are used and accessed as intended under all circumstances.

- Types:
  - Physical
    - the room the server and workstation are located
  - Human
    - Identifying intruders & unauthorized access to resources
  - Network
    - Much data travels over public or shared lines
  - Operating System
    - The OS protects itself from security breaches

Physical

Algorithmic

Physical Security

- Access to building
  - Cameras
  - Signup sheet
  - Guards
- Access to room
  - Key locks
  - Combination Locks
  - Restricted access areas (guard / signup sheet)
- Server
  - Authorized users (login)
  - Backup power supply
  - Server cloning or tape backup (also RAID)
- Workstations
  - Authorized users (login)
  - Limited resource access

cumulative

Passwords

Problems

- Guessing
  - Works: common words, about the user
  - Overcome:
    - force lowercase, uppercase and number (legal password formats)
    - System generates a random string from dictionary
- Bruit force
  - Works: reduce target space (w/ guessing or auto via program)
  - Overcome:
    - Logout after n attempts
    - Record failed attempts to a log file
    - Do not say which was in error (user name or password)
- Asking
  - Works: People are too trustworthy (stories…)
  - Overcome:
    - Policies
    - Training
    - Auto password change rules (if breached – short term access)

Password Storage

- Files
  - Difficulty: anyone can access, copy and edit (or damage)
  - Overcome: Encryption, but if can copy then can try-and-try
- System Database
  - Difficulty: can be copied or damaged
  - Overcome: Copy and then try-and-try (maybe harder)
- System Partition
  - Bad: If problem hard to fix (maybe need to reformat / partition)
  - Good: Does not permit user login (but does permit login from root terminal)

Special Password Systems

One-Time Passwords

- Paired Authentication
  - Computer & User share a secret
  - Computer prompts with integer number n
  - User replies with y = secret(n)
  - Computer verifies y with its own computation of secret(n)
  - Good: if someone sees y, no good without secret

- Two Factor Authentication (PIN Authentication)
  - User's PIN + server's n and ATM's secret(n)

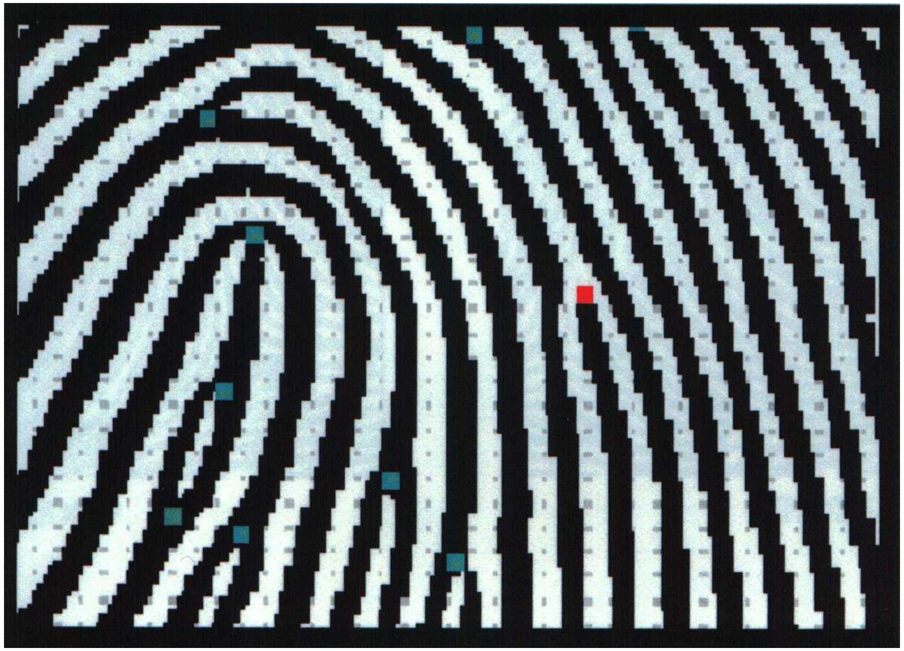
- Code Book Authentication (S/Key Systems)
  - User and System have a list of single use passwords
  - They are used in order (based on user name)
  - Once used, cannot be used again.

Complex Passwords

- Lower/upper case & numbers
- Two-password systems

Biometrics

- Eye
  - Photo matching
  - Vein configurations
- Finger print
  - Temperature grids
  - Photo matching
  - Minutiae Identification
    - Pattern matching
    - Neural Networks



Research

Program Threats

Trojan Horses

- Definition
  - A program that executes under another user’s security rights but performs an activity the user does not want to occur.
- Examples:
  - Valid: a compiler, installed by root used by all
  - Invalid:
    - A copy-cat login page
    - Program name swap (overwrite gcc with your own)
    - Launch your program as another user (user-bit)

Trap Door

- Definition
  - – A secret key-stroke opens a secret menu or feature in a software program
- Examples:
  - – Games (unlimited strength or no death mode)
  - – Debug mode
  - – Access to command-line prompt
    - If user-bit has changed the security level…

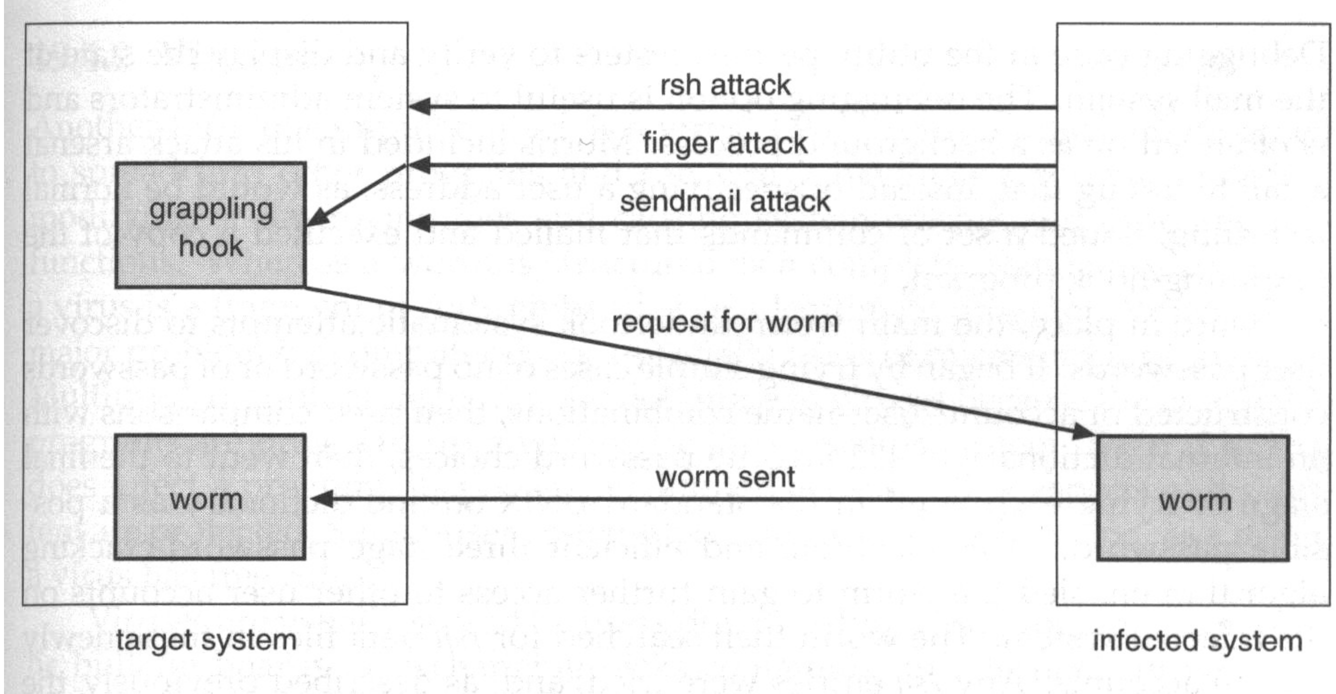
Run-Time Crash

- Definition
  - Program terminates execution unexpectedly or calls error-handler. Caller program return address saved on stack and OS called. When OS finished handler uses stack to return to user’s program.
    - But if that address was changed …
- Examples:
  - Ask for INT input STRING
  - Buffer overflow

You can artificially force this to happen

Worms

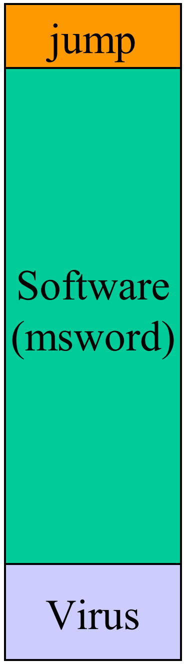
- Definition
  - A complete stand-alone program that can travel about a single computer system or network replicating and spawning itself thereby reducing the performance of the system.



Worms can be used to search out and log or return info

Viruses

- Definition
  - Like a worm they infect other system and reduce their performance but they are different in the following ways:
    - They are not complete programs but are snippets that attach themselves to other programs
    - They vigorously effect the system and network by replication at each launch of infected program
    - They often have secondary motives of system damage



jump

Software (msword)

Virus

Denial of Service

- Definition
  - A system has not been penetrated but its resources are no longer available for legitimate use.
- Examples
  - Penetrated system: format or delete system files
  - Not penetrated
    - Request all the system's resources and then lock
    - Request for login but do not login, spawn & repeat
      - OS assumes slow human user and waits…

      (All port assigned and waiting)

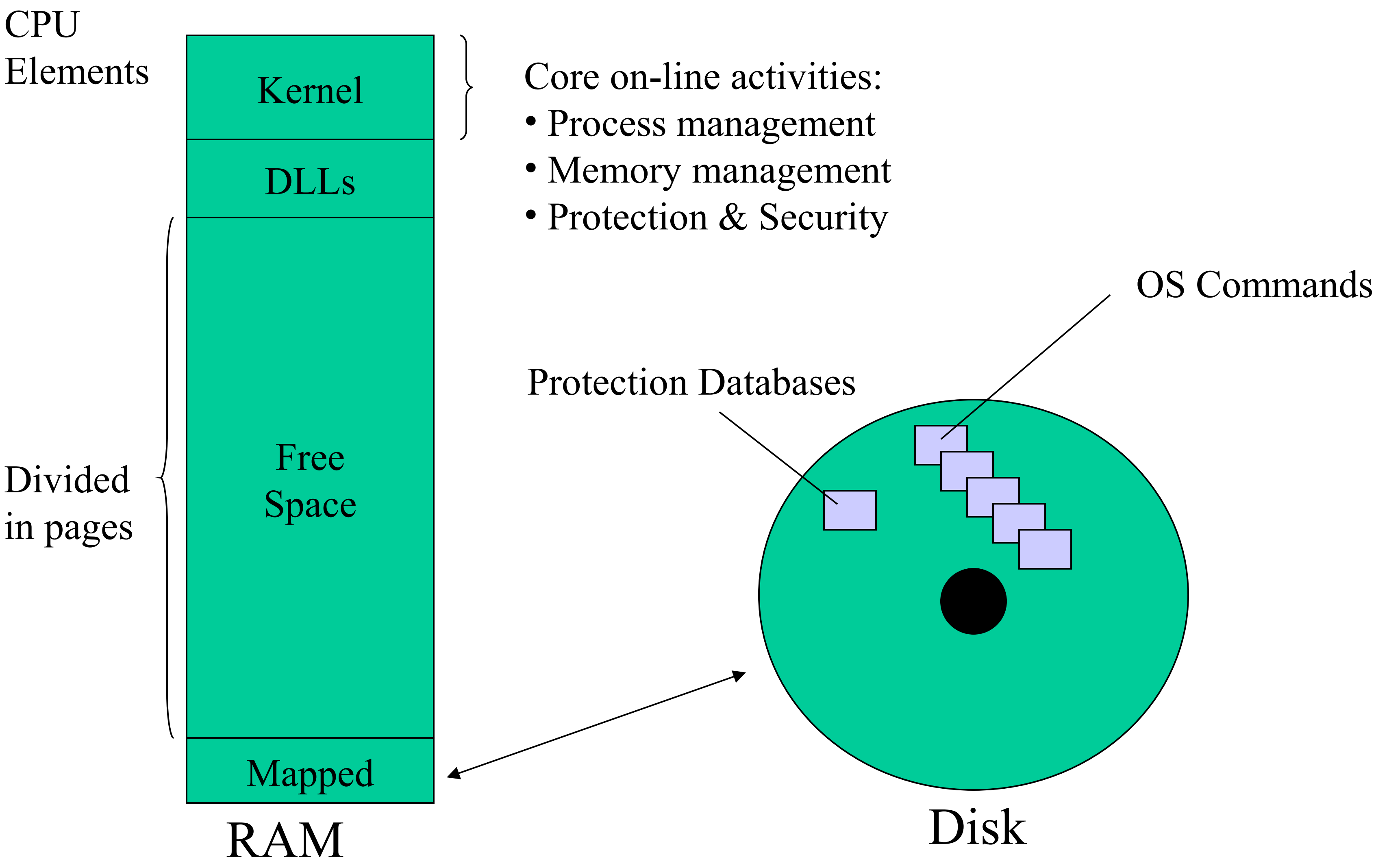
Part 2

OS Managed:
Domains of Protection

Where do we need security?

(Resources)

The Location of Protection



Locations of Protections

- Protect OS from process pointers
- Protect a page from an external reference
- Manage the sharing of files
- Manage the sharing of resources
  - Hardware
  - Software

Process ←→ OS
Process ←→ Process
Process ←→ Resource

A general concept for security

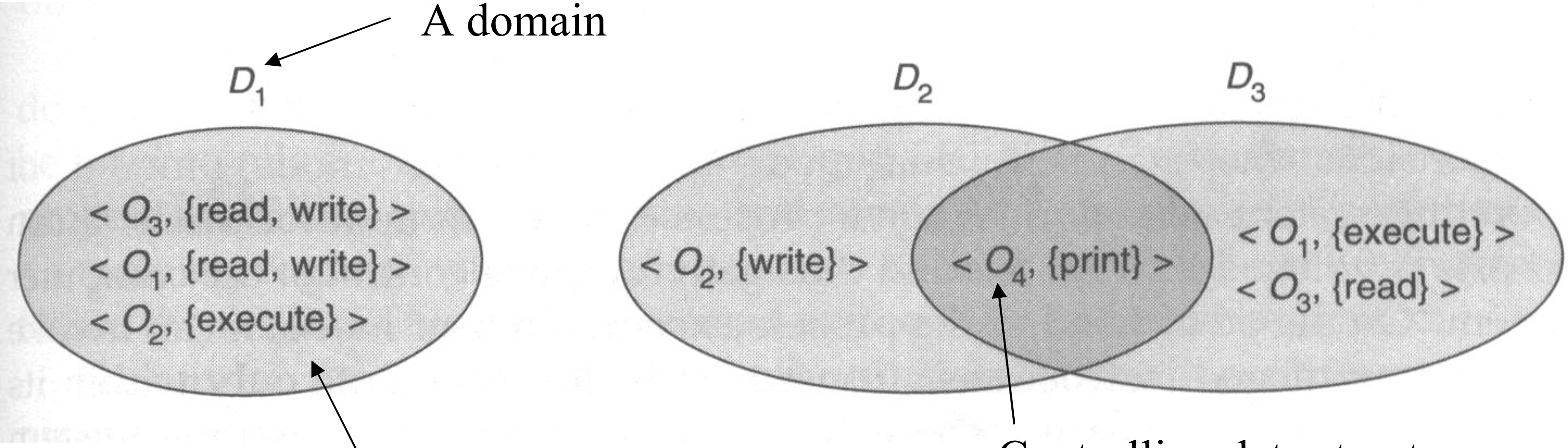
(Resources)

The Domain Concept

- A computer is composed of Objects:
  - Hardware Objects
    - CPU, printer, Disk
  - Software Objects
    - Process, file, semaphores
- Objects are composed of the following:
  - Unique ID
  - Supported physical operations
  - A domain pointer (a way to get access to it)
- A Domain structure is composed of:
  - A set of access rights:
    - Read, write, execute, owner, switch-able, super, user

Domain Interaction

- Each Object is assigned a data structure
  - The OS uses the data structure to manage the use of the Object.
- The OS and user create Domain data structures and then assign these to Objects.



A domain

Access rights

Controlling data structure of a physical object

When can we assign domains?

- Domains can be attached to users at login
  - To change a domain you would have to login as someone else
- Domains can be attached to a process when it is launched
  - An executing process must ask the OS to change its domain
- A domain can be attached to a method/procedure
  - This refers to the local variables it is permitted to instantiate

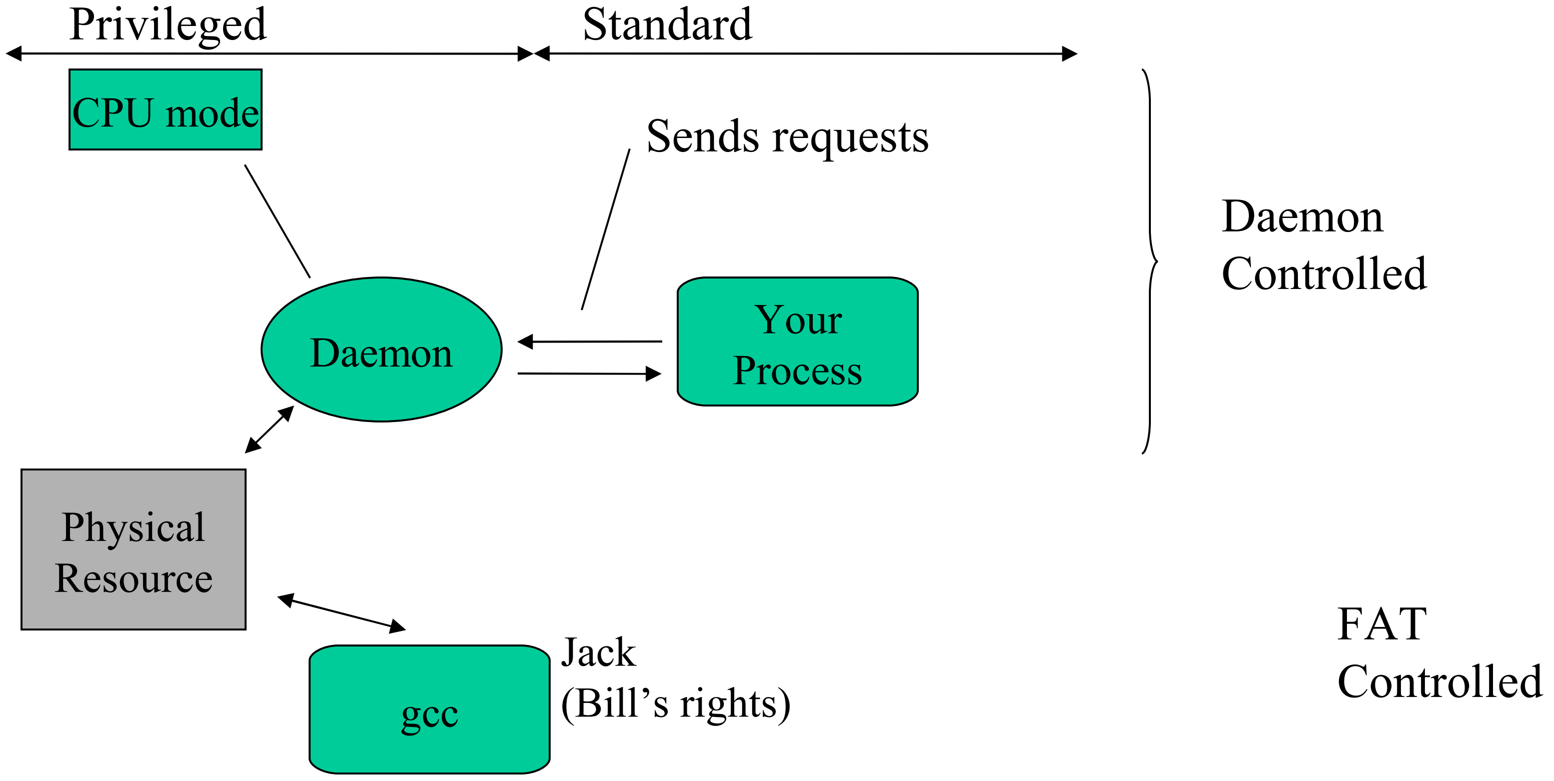
User Domains

User Domains

- User Domains are managed in this way:
  - Your user name is assigned a domain
  - At login the domain is initiated
  - The file system manages domain security
    - Each file & directory on disk is connected to your domain through fields in the FAT table

- Three User Domain levels exist:
  - Daemon Controlled
  - System Directory Controlled
  - Free FAT Controlled

User Domains

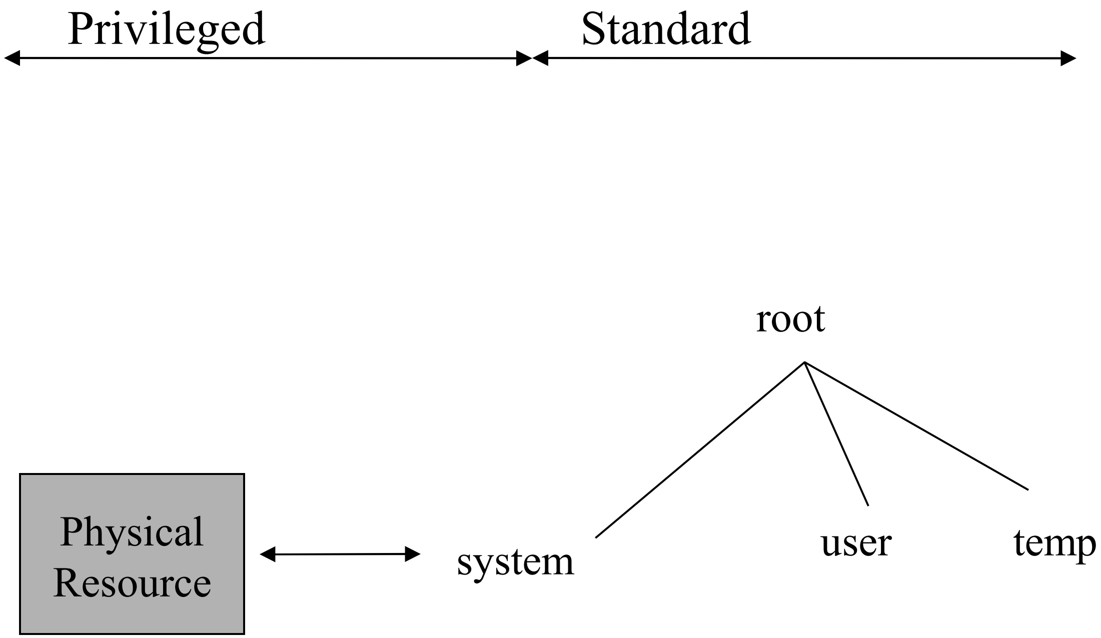


Name	Ptr	User (domain)	Domain Bit
Abc.txt	0x123	Jack	1
Gcc	0x948	Bill	0

Don't share

Permit domain change

User Domains



User Modes:

- Privileged
- Standard

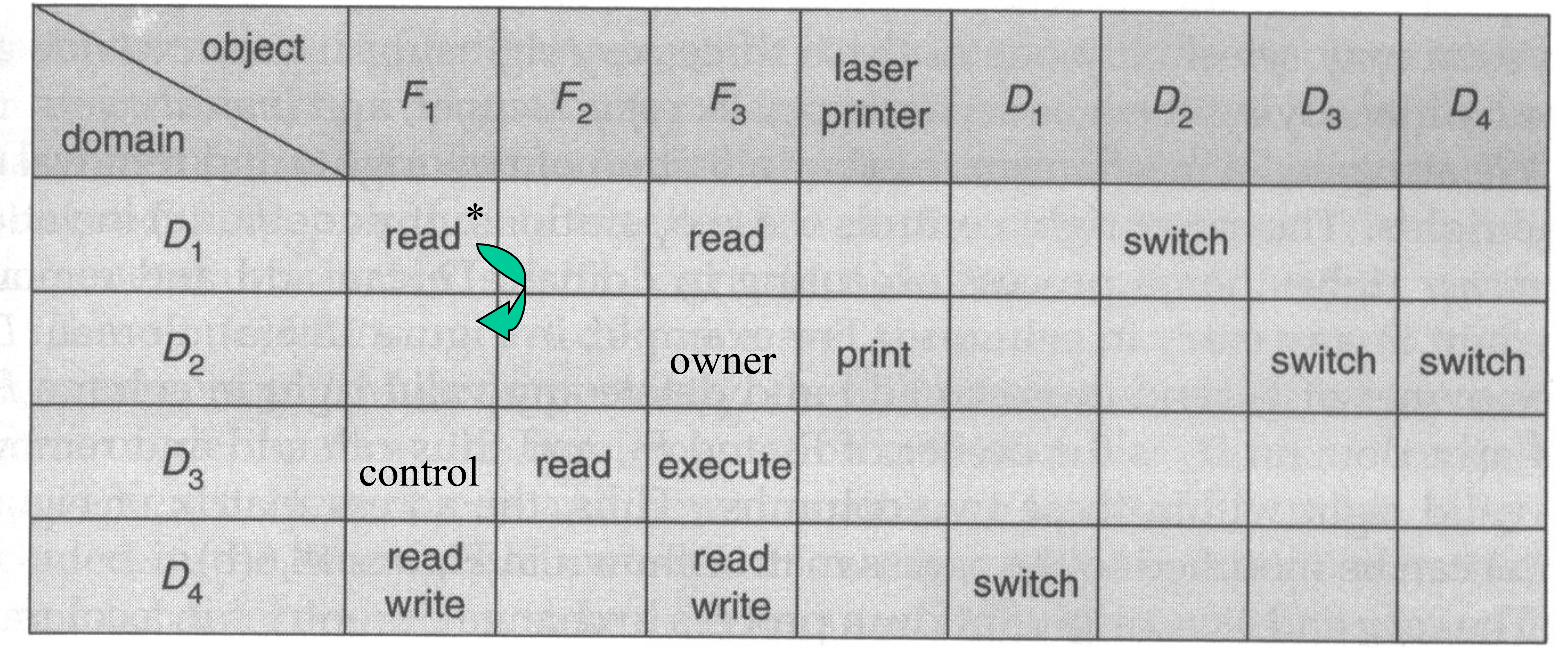
Only files copied into the system folder can have access to privileged resources.

Only privileged users can copy files into the system folder.

Part 2

OS Managed:
Access Matrix

The Access Matrix



domain \ object	F_1	F_2	F_3	laser printer	D_1	D_2	D_3	D_4
D_1	read*		read			switch		
D_2			owner	print			switch	switch
D_3	control	read	execute					
D_4	read write		read write		switch			

Access rights:

- read, write, execute
- action (print)
- copy (pass right and *) – Variations: Transfer (move right and *) / Propagate (right only)
- owner – can create and remove rights within a column
- control – can create and remove rights within a row (a super user)

Implementation Methods
-- A Matrix --

- Simplest implementation is a standard 2D matrix, but …
  - An access rights matrix tends to be sparsely populated
  - The amount of wasted/empty space is prohibitive

Implementation Methods

-- Global Database or Table --

Operation:

- Domain selection
    - User login, or
    - Process launch
- Command issued
    - Locate object
    - Match domain
    - verify right

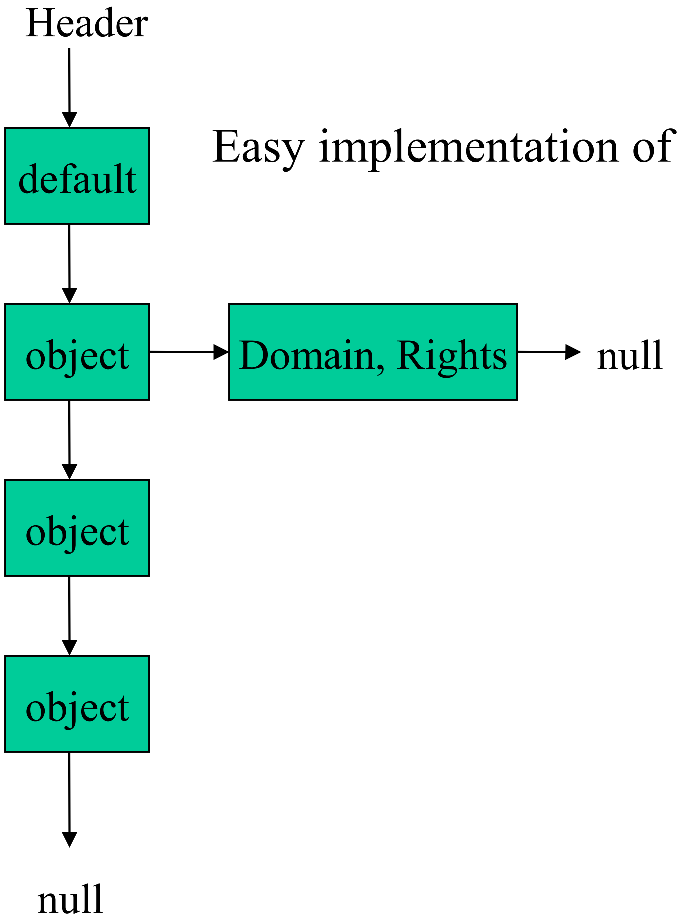
Domain	Object	Rights

Problems:

- Too big to fit in RAM!
- Cannot group objects or domains

Implementation Methods

-- Access List --



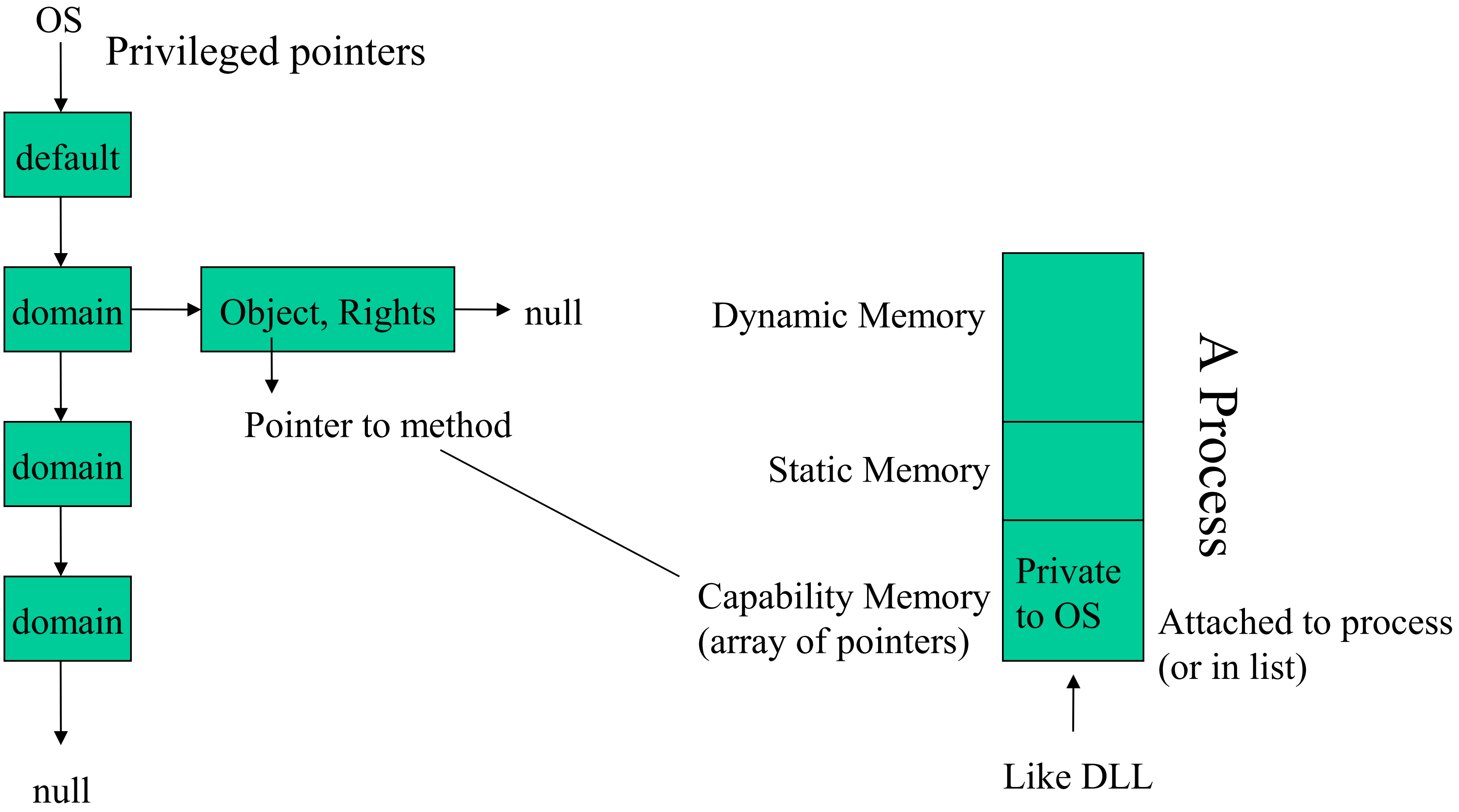
Easy implementation of default rights

Takes care of sparse population

Don’t need to load all of it into RAM
only what is needed

Implementation Methods

-- Capability List --

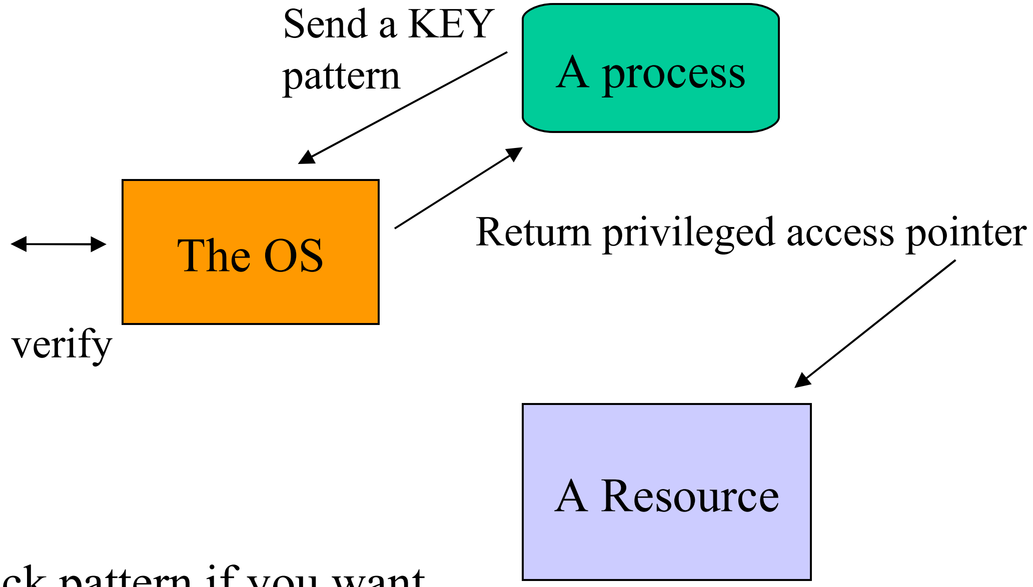


Implementation Methods

-- Lock & Key Database --

Database

Object	Lock Pattern



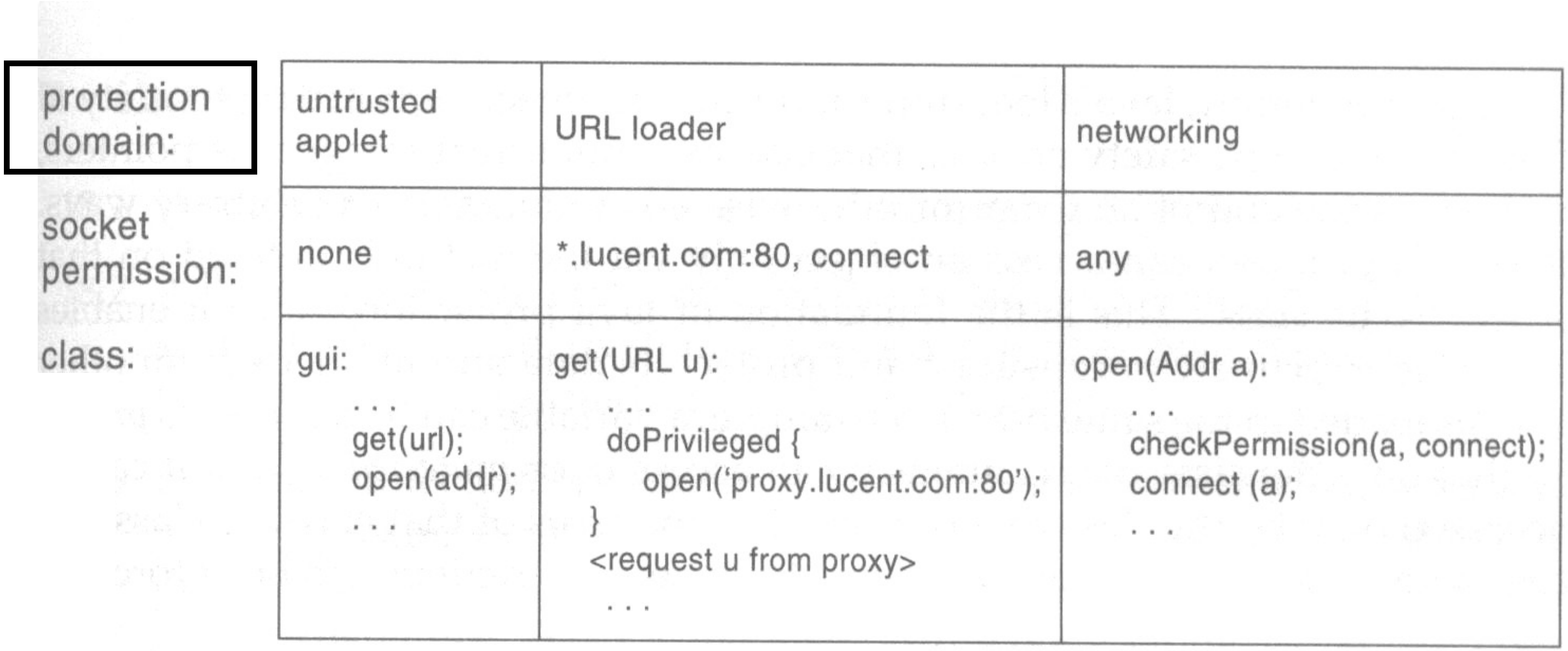
Simply change the lock pattern if you want to restrict access.

Share the lock patterns with who you want.

Part 3

Language Based Management: Compilers & Java

A Java Implementation



protection domain:	untrusted applet	URL loader	networking
socket permission:	none	*.lucent.com:80, connect	any

class:	gui:
	. . .
	get(url);
	open(addr);
	. . .
		get(URL u):
		. . .
		doPrivileged {
		open('proxy.lucent.com:80');
		}
		<request u from proxy>
		. . .
			open(Addr a):
			. . .
			checkPermission(a, connect);
			connect (a);
			. . .

The OS cannot know everything…

- Encapsulation of objects is security
- Privileged access enforced through
  - Stack inspection (requires proper interface implementation)
  - Type safety (references are not exactly pointers)

Part 3

At Home

Things to try out

1. Identify and experiment with your OS’ file access rights procedures
   - Find the windows way of setting them
   - Find the command-line way
     - Attrib, chmod, …