Secure Operating Systems

John Mitchell

Multics-Orange Book-Claremont Lecture

IEEE Security and Privacy … "The Oakland Conference"

Multics

- Operating System
  - Designed 1964-1967
    - MIT Project MAC, Bell Labs, GE
  - At peak, ~100 Multics sites
  - Last system, Canadian Department of Defense, Nova Scotia, shut down October, 2000
- Extensive Security Mechanisms
  - Influenced many subsequent systems

http://www.multicians.org/security.html

Rainbow Series

DoD Trusted Computer Sys Evaluation Criteria (Orange Book)
Audit in Trusted Systems (Tan Book)
Configuration Management in Trusted Systems (Amber Book)
Trusted Distribution in Trusted Systems (Dark Lavender Book)
Security Modeling in Trusted Systems (Aqua Book)
Formal Verification Systems (Purple Book)
Covert Channel Analysis of Trusted Systems (Light Pink Book)
… many more

http://www.radium.ncsc.mil/tpep/library/rainbow/index.html

Questions

- Most systems have significant vulnerabilities
  - Is it possible to build a secure system?
  - If so, what will it take to do it?

Government and military are biggest users of many the ideas described in this lecture. Commercial applications are possible and may become more prevalent in the future.

Outline

- Security requirements and models
- Designing a secure OS
- Requirements
  - Orange Book, Common Criteria
- Information flow
  - Covert channels

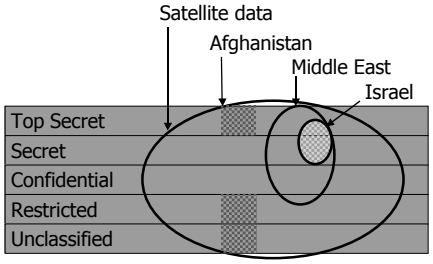
Reading: Pfleeger book; Saltzer and Schroeder

Security Policies

- Security = control over information
- Example controls over information
  - Military security policy
    - Classification involves sensitivity levels, compartments
    - Do not let classified information leak to unclassified files
  - Separation of duty
  - Chinese Wall Policy

Military security policy

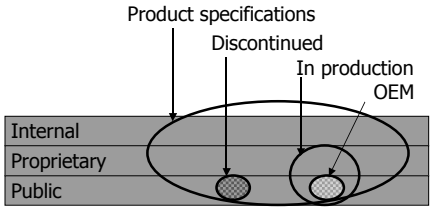
- Sensitivity levels
- Compartments



Military security policy

- Classification of personnel and data
  - Class = (rank, compartment)
- Dominance relation
  - $D_1 \geq D_2$ iff $rank_1 \geq rank_2$
    and $compartment_1 \supseteq compartment_2$
  - Example: (Restricted, Israel) ≤ (Secret, Middle East)
- Applies to
  - Subjects – users or processes
  - Objects – documents or resources

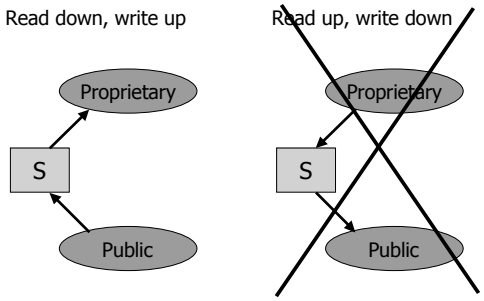
Commercial version



Bell-LaPadula Confidentiality Model

- When is it OK to release information?
- Two Properties (with silly names)
  - Simple security property
    - A subject S may read object O only if $C(O) \leq C(S)$
  - *-Property
    - A subject S with read access to O may write object P only if $C(O) \leq C(P)$
- In words,
  - You may only *read below* your classification and only *write above* your classification

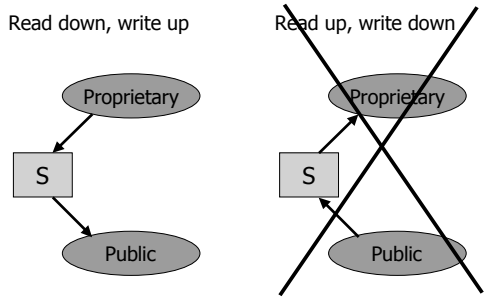
Picture: Confidentiality



Biba Integrity Model

- Rules that preserve integrity of information
- Two Properties (with silly names)
  - Simple integrity property
    - A subject S may write object O only if C(S) ≥ C(O)
      (Only trust S to modify O if S has higher rank …)
  - *-Property
    - A subject S with read access to O may write object P only if C(O) ≥ C(P)
      (Only move info from O to P if O is more trusted than P)
- In words,
  - You may only *write below* your classification and only *read above* your classification

Picture: Integrity



Problem: Models are contradictory

- Bell-LaPadula Confidentiality
  - Read down, write up
- Biba Integrity
  - Read up, write down
- Want both confidentiality and integrity
  - Only way to satisfy both models is only allow read and write at same classification

In reality: more people seem interested in Bell-LaPadula model than Biba model

Other policy concepts

- Separation of duty
  - If amount is over $10,000, check is only valid if signed by two authorized people
  - Two people must be *different*
  - Policy involves role membership and ≠
- Chinese Wall Policy
  - Lawyers L1, L2 in Firm F are experts in banking
  - If bank B1 sues bank B2,
    - L1 and L2 can each work for either B1 or B2
    - No lawyer can work for opposite sides in any case

Designing a Secure OS

- How do we design a system that can enforce security policies?
- Discretionary access control
  - Users are allowed to specify restrictions
- Mandatory access control
  - Policy must be enforced for all users, all data

Background: Security in Standard OS

- Authentication
  - Password, Smart card, biometrics, (Future lecture)
- Access control
  - File systems, network connections, etc. (Last lecture)
- Memory protection
  - Some issues covered: Stack overflow, SFI
  - Brief review of standard OS techniques here

Memory Protection [Pfleeger sec 6.2]

- OS mechanisms using hardware support
  - Fence
  - Base/Bounds registers
  - Tagged architecture
  - Segmentation
  - Paging
  - Combining paging and segmentation

Features of trusted OS [Pfleeger sec 7.4]

- User identification and authentication
- Mandatory access control
- Discretionary access control
- Object reuse protection
- Complete mediation
- Trusted path
- Audit
- Audit log reduction
- Intrusion detection

User identification and authentication

- Requirements
  - Secure identification of individuals
  - Each individual uniquely identified

Mandatory access control

- Control policies
  - Can be set by configuration not under user control
  - Can be combined with discretionary access control
    - MAC takes precedence over DAC

Object reuse protection

- Object reuse attack
  - Space for new file comes from freed file space
  - File area may initially contain old data
  - Security risk
- Protection
  - Write over old data when file space is allocated

Complete mediation

- Access control mechanisms
  - Subject requests object through reference monitor
- Mediation
  - Prevent any form of access that circumvents monitor

Trusted path

- Spoofing
  - Fool user/process into thinking they are communicating with secure part of system
  - Intercept communication
- Trusted path
  - Mechanisms to prevent spoofing
    - Special key sequence for passwd command intercepted by trusted kernel (e.g, ctrl-alt-delete)
    - Allow some actions only at boot time, before user processes loaded

Audit

- Log security-related events
- Protect audit log
  - Write to write-once non-volatile medium

Audit log reduction

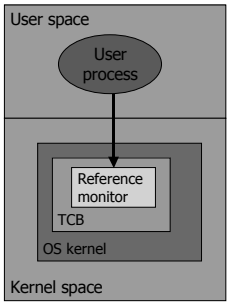
- Audit logs can become huge
  - Manage size by following policy
    - Storage becomes more feasible
    - Analysis more feasible since entries more meaningful
  - Example policies
    - Audit only first, last access by process to a file
    - Do not record routine, expected events
      - E.g., starting one process always loads …

Intrusion detection

- Profiles and anomaly detection
  - Monitor system and learn normal activity
  - Report abnormal actions
- Attack detection
  - Recognize patterns associated with known attacks

Kernelized Design

- Trusted Computing Base
  - Hardware and software for enforcing security rules
- Reference monitor
  - Part of TCB
  - All system calls go through reference monitor for security checking
  - Most OS not designed this way

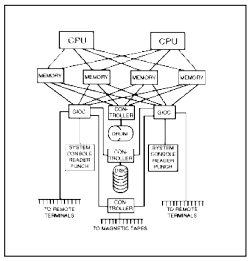


Multics

- Ring structure
  - A ring is a domain in which a process executes
  - Numbered 0, 1, 2, … ; Kernel is ring 0
  - Graduated privileges
    - Processes at ring i have privileges of every ring j > i
- Segments
  - Each data area or procedure called a segment
  - Segment protection (b1, b2, b3) with b1 ≤ b2 ≤ b3
    - Process/data can be accessed from rings b1 … b2
    - A process from rings b2 … b3 can only call segment at restricted entry points

Multics time period

- Timesharing was new concept
  - Serve Boston area with one 386-based PC



SELinux

- Security-enhanced Linux system (NSA)
  - Enforce the separation of information based on confidentiality and integrity requirements
  - Mandatory access control incorporated into the major subsystems of the kernel
    - Limit tampering and bypassing of application security mechanisms
    - Confine damage caused by malicious applications

http://www.nsa.gov/selinux/

Why Linux?

- Open source
  - Already subject to public review
    - This by itself does not guarantee security …
  - NSA can review source, modify and extend
  - Hope to encourage additional operating system security research
  - Released under the same terms and conditions as the original sources.
    - includes documentation and source code

SELinux Security Policy Abstractions

- Type enforcement
  - Each process has an associated domain
  - Each object has an associated type
  - Configuration files specify
    - How domains are allowed to access types
    - Allowable interactions and transitions between domains
- Role-based access control
  - Each process has an associated role
    - Separate system and user processes
  - configuration files specify
    - Set of domains that may be entered by each role

Orange Book Criteria

- Level D
  - No security requirements
- Level C For environments with cooperating users
  - C1 – protected mode OS, authenticated login, DAC, security testing and documentation (Unix)
  - C2 – DAC to level of individual user, object initialization, auditing (Windows???)
- Level B, A
  - All users and objects must be assigned a security label (classified, unclassified, etc.)
  - System must enforce Bell-LaPadula model

Levels B, A (continued)

- Level B
  - B1 – classification and Bell-LaPadula
  - B2 – system designed in top-down modular way, must be possible to verify, covert channels must be analyzed
  - B3 – ACLs with users and groups, formal TCB must be presented, adequate security auditing, secure crash recovery
- Level A1
  - Formal proof of protection system, formal proof that model is correct, demonstration that impl conforms to model, formal covert channel analysis

Orange Book Requirements

Security Policy	D	C1	C2	B1	B2	B3	A1
Discretionary access control		X	X	*	*	X	*
Object reuse			X	*	*	*	*
Labels				X	X	*	*
Label integrity				X	*	*	*
Export of labeled information				X	*	*	*
Labeling readable output				X	*	*	*
Mandatory access control				X	X	*	*
Subject sensitivity labels					X	*	*
Device labels					X	*	*

X – additional requirements; * - same req as lower category

Orange Book Requirements

Accountability	D	C1	C2	B1	B2	B3	A1
Identification and authorization		X	X	X	*	*	*
Audit			X	X	X	X	*
Trusted path					X	X	*

X – additional requirements; * - same req as lower category

Orange Book Requirements

Assurance	D	C1	C2	B1	B2	B3	A1
System architecture		X	X	X	X	X	*
System integrity		X	*	*	*	*	*
System testing		X	X	X	X	X	X
Design spec and verification				X	X	X	X
Covert channel analysis					X	X	X
Trusted facility management					X	X	*
Configuration management					X	*	X
Trusted recovery						X	*
Trusted distribution							X

X – additional requirements; * - same req as lower category

Orange Book Requirements

Documentation	D	C1	C2	B1	B2	B3	A1
Secure features user's guide		X	*	*	*	*	*
Trusted facilities manual		X	X	X	X	X	*
Test documentation		X	*	*	X	*	*
Design documentation		X	*	X	X	X	*

X – additional requirements; * - same req as lower category

Common Criteria

"The Common Criteria represents the outcome of a series of efforts to develop criteria for evaluation of IT security that are broadly useful within the international community."

http://www.commoncriteria.org/

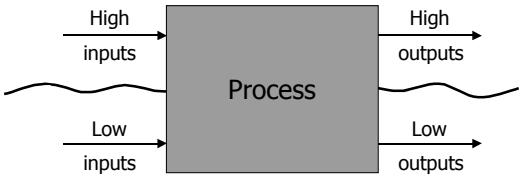
Assurance methods

- Testing
  - Can demonstrate existence of flaw, not absence
- Formal verification
  - Time-consuming, painstaking process
- "Validation"
  - Requirements checking
  - Design and code reviews
    - Sit around table, drink lots of coffee, …
  - Module and system testing

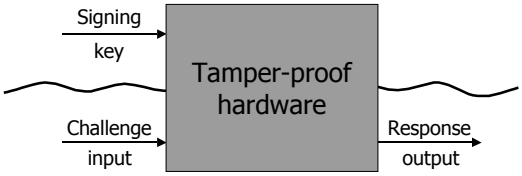
Noninterference

- Idea
  - Actions by high-level users (secret, top secret) should not be observable by low-level users (unclassified, …)

Noninterference



Example: Smart Card



Covert Channels

- Butler Lampson
  - Difficulty achieving confinement (paper handed out)
  - Communicate by using CPU, locking/unlocking file, sending/delaying msg, …
- Gustavus Simmons
  - Cryptographic techniques make it impossible to detect presence of a covert channel

Conclusions

- Desiderata
  - There are many security and information flow properties that highly managed organizations would like to enforce
- Analysis
  - Theories about requirements and security models
  - Some plausible intuitions are hard to make precise

Conclusions

- Implementation techniques
  - Standard OS protection mechanisms
  - Approaches for trusted systems
    - Try harder (complete mediation)
    - Add audit, intrusion detection, …
- Review criteria
  - Orange Book, Common Criteria
- This is hard work
  - Noninterference is slippery, hard to achieve
  - Covert channels easy in practice, theoretically impossible to eliminate