

Computer Security

CS 426

Lecture 14



Trusted Operating Systems and Assurance

Review

- Security model = system model + security policy
- Bell-LaPadula model = state transition system + every reachable state must satisfy simple security property and *-property
- Covert channel

Plan for this lecture

- Trusted vs. trustworthy, TCB, Trusted Path
- Design principles for security mechanisms
- Security features of a “Trusted OS”
- Assurance criteria
 - TCSEC
 - Common criteria
- Readings:
 - Security Engineering: Chapter 23
 - The Protection of Information in Computer Systems: Section 1A

Trusted vs. Trustworthy

- A component of a system is trusted means that
 - the security of the system depends on it
 - if the component is insecure, so is the system
 - determined by its role in the system
- A component is trustworthy means that
 - the component deserves to be trusted
 - e.g., it is implemented correctly
 - determined by intrinsic properties of the component

Trusted Operating System is actually a misnomer

Terminology: Trusted Computing Base

- The set of all hardware, software and procedural components that enforce the security policy.
 - in order to break security, an attacker must subvert one or more of them.
- What consists of the conceptual Trusted Computing Base in a Unix/Linux system?
 - hardware, kernel, system binaries, system configuration files, etc.

Terminology: Trusted Path

- Mechanism that provides confidence that the user is communicating with what the user intended to communicate with (typically TCB)
 - attackers can't intercept or modify whatever information is being communicated.
 - defends attacks such as fake login programs
- Example: Ctrl+Alt+Del for log in on Windows

Terminology: Trusted Computing and Trusted Platform Module

- Trusted Computing means that the computer will consistently behave in specific ways, and those behaviors will be enforced by hardware and software.
- Trusted Computing Group
 - an alliance of Microsoft, Intel, IBM, HP and AMD;
 - promotes a standard for a 'more secure' PC.
 - formally Trusted Computing Platform Alliance (TCPA)
- Trusted Platform Module
 - a specification by TCP or implementation of the specification
 - a hardware module (integrated circuit) that provides
 - secure generation of cryptographic keys,
 - storage of keys that cannot be retrieved
 - a Hardware Random Number Generator.
 - remote attestation, etc

Terminology: Trusted Computing

- Next-Generation Secure Computing Base (NGSCB) by Microsoft
 - formally Palladium
 - a software architecture claims to intend to provide strong process isolation, sealed storage, secure path to and from the user, and attestation
 - relies on Trusted Platform Module
- Ensure that users can't tamper with the application software, and these applications can communicate securely with their authors and with each other
 - driven by Digital Right Management needs
- Criticisim
 - vendor lock-in, privacy, etc.

Design Principles of Security Mechanisms (Saltzer and Schroeder 75)

1. Economy of mechanism
 - keep the design as simple and small as possible
2. Fail-safe defaults
 - default is no-access
3. Complete mediation
 - every access must be checked
4. Open design
 - security does not depend on the secrecy of mechanism

Design Principles of Security Mechanisms (Saltzer and Schroeder 75)

5. Separation of privilege

- a system that requires two keys is more robust than one that requires one

6. Least privilege

- every program and every user should operate using the least privilege necessary to complete the job

7. Least common mechanism

- “minimize the amount of mechanism common to more than one user and depended on by all users”

8. Psychological acceptability

- “human interface should be designed for ease of use”
- the user’s mental image of his protection goals should match the mechanism

Plan for this lecture

- Trusted vs. trustworthy, TCB, Trusted Path
- Design principles for security mechanisms
- Security features of a “Trusted OS”
- Assurance criteria
 - TCSEC
 - Common criteria
- Readings:
 - Security Engineering: Chapter 23
 - The Protection of Information in Computer Systems: Section 1A

What makes a “Secure” OS? Or “Trusted OS”

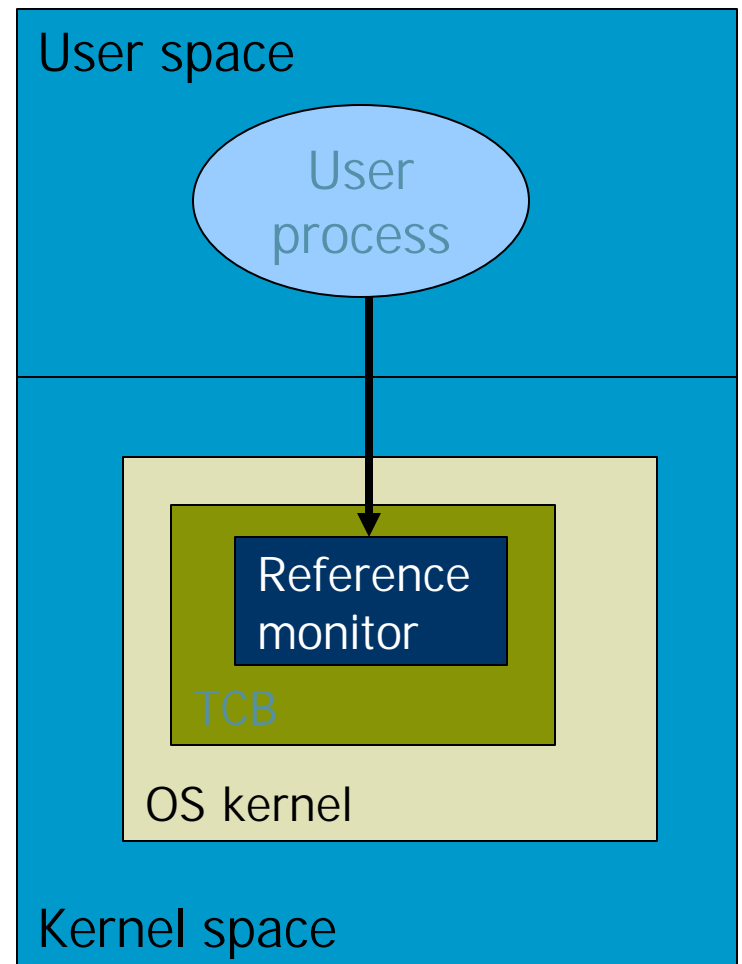
- Extra security features (compared to ordinary OS)
 - Stronger authentication mechanisms
 - Example: require token + password
 - More security policy options
 - Example: only let users read file f for purpose p
 - Logging and other features
- More secure implementation
 - Apply secure design and coding principles
 - Assurance and certification
 - Code audit or formal verification
 - Maintenance procedures
 - Apply patches, etc.

Sample Features of “Trusted OS”

- Mandatory access control
 - MAC not under user control, precedence over DAC
- Object reuse protection
 - Write over old data when file space is allocated
- Complete mediation
 - Prevent any access that circumvents monitor
- Audit
 - Log security-related events and check logs
- Intrusion detection
 - Anomaly detection: Learn normal activity, Report abnormal
 - Attack detection: Recognize patterns OF 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



Reference Monitor Revisited

- Three required properties for reference monitors in “trusted systems”
 - tamper-proof
 - non-bypassable (complete mediation)
 - small enough to be analyzable

Audit

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

Assurance

- Trusted OS = Additional Security Features + Higher level of assurance
- Assurance: “estimate of the likelihood that a system will not fail in some particular way”
- Based on factors such as
 - development process
 - who developed it
 - technical assessment

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

Assurance Criteria

- Criteria are specified to enable evaluation
- Originally motivated by military applications, but now is much wider
- Examples
 - Orange Book (Trusted Computer System Evaluation Criteria)
 - Common Criteria

TCSEC: 1983–1999

- Trusted Computer System Evaluation Criteria
 - Also known as the Orange Book
 - Series that expanded on Orange Book in specific areas was called *Rainbow Series*
 - Developed by National Computer Security Center, US Dept. of Defense
- Heavily influenced by Bell-LaPadula model and reference monitor concept
- Emphasizes confidentiality

Evaluation Classes C and D

Division D: Minimal Protection

D Did not meet requirements of any other class

Division C: Discretionary Protection

C1 *Discretionary protection*; DAC, Identification and Authentication, TCB should be protected from external tampering, ...

C2 *Controlled access protection*; object reuse, auditing, more stringent security testing

Division B: Mandatory Protection

- B1 *Labeled security protection*; informal security policy model; MAC for named objects; label exported objects; more stringent security testing
- B2 *Structured protection*; formal security policy model; MAC for all objects, labeling; trusted path; least privilege; covert channel analysis, configuration management
- B3 *Security domains*; full reference validation mechanism; increases trusted path requirements, constrains code development; more DTLS requirements; documentation

Division A: Verification Protection

A1 *Verified design*;

functionally equivalent to B3, by require the use of formal methods for assurance; trusted distribution; code, formal top-level specification (FTLS) correspondence

Limitations

- Written for operating systems
 - NCSC introduced “interpretations” for other things such as networks (*Trusted Network Interpretation*, the Red Book), databases (*Trusted Database Interpretation*, the Purple or Lavender Book)
- Focuses on BLP
 - Most commercial firms do not need MAC
- Does not address integrity or availability
 - Critical to commercial firms
- Combine functionality and assurance in a single linear scale

Contributions

- Heightened awareness in commercial sector to computer security needs
- Led to wave of new approaches to evaluation
 - As commercial firms could not use it for their products, some commercial firms began offering certifications
- Basis for several other schemes, such as Federal Criteria, Common Criteria

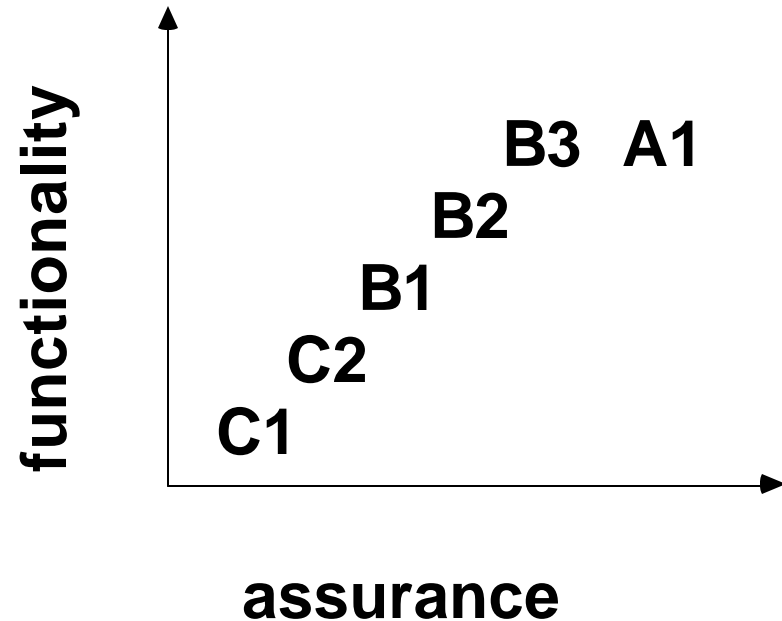
ORANGE BOOK CLASSES

UNOFFICIAL VIEW

- C1,C2 Simple enhancement of existing systems.
No breakage of applications
- B1 Relatively simple enhancement of existing systems. Will break some applications.
- B2 Relatively major enhancement of existing systems. Will break many applications.
- B3 Failed A1
- A1 Top down design and implementation of a new system from scratch

FUNCTIONALITY VS ASSURANCE

- **functionality is multi-dimensional**
- **assurance has a linear progression**



Common Criteria: 1998–Present

- An international standard (ISO/IEC 15408)
- Began in 1998 with signing of Common Criteria Recognition Agreement with 5 signers
 - US, UK, Canada, France, Germany
- As of May 2002, 10 more signers
 - Australia, Finland, Greece, Israel, Italy, Netherlands, New Zealand, Norway, Spain, Sweden; India, Japan, Russia, South Korea developing appropriate schemes
- Standard 15408 of International Standards Organization
- *De facto* US security evaluation standard, replaces TCSEC

Common Criteria

- Does not provide one list of security features
- Describes a framework where security requirements can be specified, claimed, and evaluated
- Key concepts
 - **Target Of Evaluation (TOE)**: the product or system that is the subject of the evaluation.
 - **Protection Profile (PP)**: a document that identifies security requirements relevant to a user community for a particular purpose.
 - **Security Target (ST)**: a document that identifies the security properties one wants to evaluate against
 - **Evaluation Assurance Level (EAL)** - a numerical rating (1-7) reflecting the assurance requirements fulfilled during the evaluation.

CC Functional Requirements

- Contains 11 classes of functional requirements
 - Each contain one or more families
 - Elaborate naming and numbering scheme
- Classes: Security Audit, Communication, Cryptographic Support, User Data Protection, Identification and Authentication, Security Management, Privacy, Protection of Security Functions, Resource Utilization, TOE Access, Trusted Path
- Families of Identification and Authentication
 - Authentication Failures, User Attribute Definition, Specification of Secrets, User Authentication, User Identification, and User/Subject Binding

CC Assurance Requirements

- Ten security assurance classes
- Classes:
 - Protection Profile Evaluation
 - Security Target Evaluation
 - Configuration Management
 - Delivery and Operation
 - Development
 - Guidance Documentation
 - Life Cycle
 - Tests
 - Vulnerabilities Assessment
 - Maintenance of Assurance

Protection Profiles (PP)

- “A CC protection profile (PP) is an implementation-independent set of security requirements for a category of products or systems that meet specific consumer needs”
 - Subject to review and certified
- Requirements
 - Functional
 - Assurance
 - EAL

Protection Profiles

- Example: Controlled Access PP (CAPP_V1.d)
 - Security functional requirements
 - Authentication, User Data Protection, Prevent Audit Loss
 - Security assurance requirements
 - Security testing, Admin guidance, Life-cycle support, ...
 - Assumes non-hostile and well-managed users
 - Does not consider malicious system developers

Security Targets (ST)

- “A security target (ST) is a set of security requirements and specifications to be used for evaluation of an identified product or system”
- Can be based on a PP or directly taking components from CC
- Describes specific security functions and mechanisms

Evaluation Assurance Levels 1 – 4

EAL 1: Functionally Tested

- Review of functional and interface specifications
- Some independent testing

EAL 2: Structurally Tested

- Analysis of security functions, incl. high-level design
- Independent testing, review of developer testing

EAL 3: Methodically Tested and Checked

- Development environment controls; config mgmt

EAL 4: Methodically Designed, Tested, Reviewed

- Informal spec of security policy, Independent testing

Evaluation Assurance Levels 5 – 7

EAL 5: Semiformally Designed and Tested

- Formal model, modular design
- Vulnerability search, covert channel analysis

EAL 6: Semiformally Verified Design and Tested

- Structured development process

EAL 7: Formally Verified Design and Tested

- Formal presentation of functional specification
- Product or system design must be simple
- Independent confirmation of developer tests

Example: Windows 2000, EAL 4+

- Level EAL 4 + Flaw Remediation
 - “EAL 4 ... represents the highest level at which products not built specifically to meet the requirements of EAL 5-7 ought to be evaluated.”
(EAL 5-7 requires more stringent design and development procedures ...)
 - Flaw Remediation: the tracking of security flaws, the identification of corrective actions, and the distribution of corrective action information to customers.
- Catch:
 - Evaluation based on specific configurations specified by the vendor in which the vendor can make certain assumptions about the operating environment and the strength of threats, if any, faced by the product in that environment.

Is Windows 2000 “Secure”?

- Good things
 - Design goals include security goals
 - Independent review, configuration guidelines
- But ...
 - “Secure” is a complex concept
 - What properties protected against what attacks?
 - Typical installation includes more than just OS
 - Many problems arise from applications, device drivers
 - Security depends on installation as well as system

Implications of EALs

- A higher EAL means nothing more, or less, than that the evaluation completed a more stringent set of quality assurance requirements.
- It is often assumed that a system that achieves a higher EAL will provide its security features more reliably, but there is little or no published evidence to support that assumption.
- Anything below EAL4 doesn't mean much
- Anything above EAL4 is very difficult for complex systems such as OS
- Evaluation is done for environments assumed by vendors

Highly Evaluated Systems

- SCOMP (Secure Communications Processor),
 - evaluated to A1 under TCSEC
- XTS-400
 - multi-level secure operating system
 - developed by BAE systems (largest defense contractor in Europe)
 - released in December of 2003
 - As of July 2006, the only general-purpose operating system with a Common Criteria assurance level rating of EAL5 or above
- Interactive Link
 - only product evaluated to EAL7
 - is a suite of hardware and software products to implement network separation

Criticism of CC:

- Evaluation is a costly process (often measured in hundreds of thousands of US dollars) -- and the vendor's return on that investment is not necessarily a more secure product
- Evaluation focuses primarily on assessing the evaluation documentation, not the product itself
- The effort and time to prepare evaluation-related documentation is so cumbersome that by the time the work is completed, the product in evaluation is generally obsolete
- Industry input, including that from organizations such as the Common Criteria Vendor's Forum, generally has little impact on the process as a whole

Coming Attractions ...

- October 11:
 - Access Control for Integrity Protection: Biba, Clark-Wilson, Chinese Wall

