Chapter 18: Evaluating Systems

- Goals
- Trusted Computer System Evaluation Criteria
- FIPS 140
- Common Criteria
- SSE-CMM

©2004 Matt Bishop

Overview

- Goals
  - Why evaluate?
- Evaluation criteria
  - TCSEC (*aka* Orange Book)
  - FIPS 140
  - Common Criteria
  - SSE-CMM

©2004 Matt Bishop

Goals

- Show that a system meets specific security requirements under specific conditions
  - Called a *trusted* system
  - Based on specific assurance evidence
- *Formal evaluation methodology*
  - Technique used to provide measurements of trust based on specific security requirements and evidence of assurance

©2004 Matt Bishop

Evaluation Methodology

- Provides set of requirements defining security functionality for system
- Provides set of assurance requirements delineating steps for establishing that system meets its functional requirements
- Provides methodology for determining that system meets functional requirements based on analysis of assurance evidence
- Provides measure of result indicating how trustworthy system is with respect to security functional requirements
  - Called *level of trust*

©2004 Matt Bishop

Why Evaluate?

- Provides an independent assessment, and measure of assurance, by experts
  - Includes assessment of requirements to see if they are consistent, complete, technically sound, sufficient to counter threats
  - Includes assessment of administrative, user, installation, other documentation that provides information on proper configuration, administration, use of system
- Independence critical
  - Experts bring fresh perspectives, eyes to assessment

©2004 Matt Bishop

Bit of History

- Government, military drove early evaluation processes
  - Their desire to use commercial products led to businesses developing methodologies for evaluating security, trustworthiness of systems
- Methodologies provide combination of
  - Functional requirements
  - Assurance requirements
  - Levels of trust

©2004 Matt Bishop

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
  - Integrity addressed by *-property

©2004 Matt Bishop

Functional Requirements

- Discretionary access control requirements
  - Control sharing of named objects
  - Address propagation of access rights, ACLs, granularity of controls
- Object reuse requirements
  - Hinder attacker gathering information from disk or memory that has been deleted
  - Address overwriting data, revoking access rights, and assignment of resources when data in resource from previous use is present

©2004 Matt Bishop

Functional Requirements

- Mandatory access control requirements (B1 up)
  - Simple security condition, *-property
  - Description of hierarchy of labels
- Label requirements (B1 up)
  - Used to enforce MAC
  - Address representation of classifications, clearances, exporting labeled information, human-readable output
- Identification, authentication requirements
  - Address granularity of authentication data, protecting that data, associating identity with auditable actions

©2004 Matt Bishop

Functional Requirements

- Audit requirements
  - Define what audit records contain, events to be recorded; set increases as other requirements increase
- Trusted path requirements (B2 up)
  - Communications path guaranteed between user, TCB
- System architecture requirements
  - Tamperproof reference validation mechanism
  - Process isolation
  - Enforcement of principle of least privilege
  - Well-defined user interfaces

©2004 Matt Bishop

Functional Requirements

- Trusted facility management (B2 up)
  - Separation of operator, administrator roles
- Trusted recovery (A1)
  - Securely recover after failure or discontinuity
- System integrity requirement
  - Hardware diagnostics to validate on-site hardware, firmware of TCB

©2004 Matt Bishop

Assurance Requirements

- Configuration management requirements (B2 up)
  - Identify configuration items, consistent mappings among documentation and code, tools for generating TCB
- System architecture requirements
  - Modularity, minimize complexity, etc.
  - TCB full reference validation mechanism at B3
- Trusted distribution requirement (A1)
  - Address integrity of mapping between masters and on-site versions
  - Address acceptance procedures

©2004 Matt Bishop

Assurance Requirements

- Design specification, verification requirements
  - B1: informal security policy model shown to be consistent with its axioms
  - B2: formal security policy model proven to be consistent with its axioms, descriptive top-level specification (DTLS)
  - B3: DTLS shown to be consistent with security policy model
  - A1: formal top-level specification (FTLS) shown consistent with security policy model using approved formal methods; mapping between FTLS, source code

©2004 Matt Bishop

Assurance Requirements

- Testing requirements
  - Address conformance with claims, resistance to penetration, correction of flaws
  - Requires searching for covert channels for some classes
- Product documentation requirements
  - Security Features User's Guide describes uses, interactions of protection mechanisms
  - Trusted Facility Manual describes requirements for running system securely
- Other documentation: test, design docs

©2004 Matt Bishop

Evaluation Classes A and B

A1 *Verified protection*; significant use of formal methods; trusted distribution; code, FTLS correspondence

B3 *Security domains*; full reference validation mechanism; increases trusted path requirements, constrains code development; more DTLS requirements; documentation

B2 *Structured protection*; formal security policy model; MAC for all objects, labeling; trusted path; least privilege; covert channel analysis, configuration management

B1 *Labeled security protection*; informal security policy model; MAC for some objects; labeling; more stringent security testing

©2004 Matt Bishop

Evaluation Classes C and D

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

C1 *Discretionary protection*; minimal functional, assurance requirements; I&A controls; DAC

D Did not meet requirements of any other class

©2004 Matt Bishop

Evaluation Process

- Run by government, no fee to vendor
- 3 stages
  - Application: request for evaluation
    - May be denied if gov't didn't need product
  - Preliminary technical review
    - Discussion of evaluation process, schedules, development process, technical content, etc.
    - Determined schedule for evaluation
  - Evaluation phase

©2004 Matt Bishop

Evaluation Phase

- 3 parts; results of each presented to technical review board composed of senior evaluators *not* on evaluating team; must approve that part before moving on to next part
  - Design analysis: review design based on documentation provided; developed initial product assessment report
    - Source code not reviewed
  - Test analysis: vendor's, evaluators' tests
  - Final evaluation report
- Once approved, all items closed, rating given

©2004 Matt Bishop

RAMP

- Ratings Maintenance Program goal: maintain assurance for new version of evaluated product
- Vendor would update assurance evidence
- Technical review board reviewed vendor's report and, on approval, assigned evaluation rating to new version of product
- Note: major changes (structural, addition of some new functions) could be rejected here and a full new evaluation required

©2004 Matt Bishop

Impact

- New approach to evaluating security
  - Based on analyzing design, implementation, documentation, procedures
  - Introduced evaluation classes, assurance requirements, assurance-based evaluation
  - High technical standards for evaluation
  - Technical depth in evaluation procedures
- Some problems
  - Evaluation process difficult, lacking in resources
  - Mixed assurance, functionality together
  - Evaluations only recognized in US

©2004 Matt Bishop

Scope 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 needs of US government
  - Most commercial firms do not need MAC
- Does not address integrity or availability
  - Critical to commercial firms

©2004 Matt Bishop

Process Limitations

- Criteria creep (expansion of requirements defining classes)
  - Criteria interpreted for specific product types
  - Sometimes strengthened basic requirements over time
  - Good for community (learned more about security), but inconsistent over time
- Length of time of evaluation
  - Misunderstanding depth of evaluation
  - Management practices of evaluation
  - As was free, sometimes lacking in motivation

©2004 Matt Bishop

Contributions

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

©2004 Matt Bishop

FIPS 140: 1994–Present

- Evaluation standard for cryptographic modules (implementing cryptographic logic or processes)
  - Established by US government agencies and Canadian Security Establishment
- Updated in 2001 to address changes in process and technology
  - Officially, FIPS 140-2
- Evaluates only crypto modules
  - If software, processor executing it also included, as is operating system

©2004 Matt Bishop

Requirements

- Four increasing levels of security
- FIPS 140-1 covers basic design, documentation, roles, cryptographic key management, testing, physical security (from electromagnetic interference), etc.
- FIPS 140-2 covers specification, ports & interfaces; finite state model; physical security; mitigation of other attacks; etc.

©2004 Matt Bishop

Security Level 1

- Encryption algorithm must be FIPS-approved algorithm
- Software, firmware components may be executed on general-purpose system using unevaluated OS
- No physical security beyond use of production-grade equipment required

©2004 Matt Bishop

Security Level 2

- More physical security
  - Tamper-proof coatings or seals or pick-resistent locks
- Role-based authentication
  - Module must authenticate that operator is authorized to assume specific role and perform specific services
- Software, firmware components may be executed on multiuser system with OS evaluated at EAL2 or better under Common Criteria
  - Must use one of specified set of protection profiles

©2004 Matt Bishop

Security Level 3

- Enhanced physical security
  - Enough to prevent intruders from accessing critical security parameters within module
- Identity-based authentication
- Strong requirements for reading, altering critical security parameters
- Software, firmware components require OS to have EAL3 evaluation, trusted path, informal security policy model
  - Can use equivalent evaluated trusted OS instead

©2004 Matt Bishop

Security Level 4

- "Envelope of protection" around module that detects, responds to all unauthorized attempts at physical access
  - Includes protection against environmental conditions or fluctuations outside module's range of voltage, temperatures
- Software, firmware components require OS meet functional requirements for Security Level 3, and assurance requirements for EAL4
  - Equivalent trusted operating system may be used

©2004 Matt Bishop

Impact

- By 2002, 164 modules, 332 algorithms tested
  - About 50% of modules had security flaws
  - More than 95% of modules had documentation errors
  - About 25% of algorithms had security flaws
  - More than 65% had documentation errors
- Program greatly improved quality, security of cryptographic modules

©2004 Matt Bishop

Common Criteria: 1998–Present

- 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

©2004 Matt Bishop

Evaluation Methodology

- CC documents
  - Overview of methodology, functional requirements, assurance requirements
- CC Evaluation Methodology (CEM)
  - Detailed guidelines for evaluation at each EAL; currently only EAL1–EAL4 defined
- Evaluation Scheme or National Scheme
  - Country-specific infrastructures implementing CEM
  - In US, it's CC Evaluation and Validation Scheme; NIST accredits commercial labs to do evaluations

©2004 Matt Bishop

CC Terms

- *Target of Evaluation* (TOE): system or product being evaluated
- *TOE Security Policy* (TSP): set of rules regulating how assets managed, protected, distributed within TOE
- *TOE Security Functions* (TSF): set consisting of all hardware, software, firmware of TOE that must be relied on for correct enforcement of TSP
  - Generalization of TCB

©2004 Matt Bishop

Protection Profiles

- *CC Protection Profile* (PP): implementation-independent set of security requirements for category of products or systems meeting specific consumer needs
  - Includes functional requirements
    - Chosen from CC functional requirements by PP author
  - Includes assurance requirements
    - Chosen from CC assurance requirements; may be EAL plus others
  - PPs for firewalls, desktop systems, etc.
  - Evolved from ideas in earlier criteria

©2004 Matt Bishop

Form of PP

1. Introduction
   - PP Identification and PP Overview
2. Product or System Family Description
   - Includes description of type, general features of product or system
3. Product or System Family Security Environment
   - Assumptions about intended use, environment of use;
   - Threats to the assets; and
   - Organizational security policies for product or system

©2004 Matt Bishop

Form of PP (*con't*)

4. Security Objectives
   - Trace security objectives for product back to aspects of identified threats and/or policies
   - Trace security objectives for environment back to threats not completely countered by product or systemand/or policies or assumptions not completely met by product or system
5. IT Security Requirements
   - Security functional requirements drawn from CC
   - Security assurance requirements based on an EAL
     - May supply other requirements without reference to CC

©2004 Matt Bishop

Form of PP (con't)

6. Rationale
   - Security Objectives Rationale demonstrates stated objectives traceable to all assumptions, threats, policies
   - Security Requirements Rationale demonstrates requirements for product or system and for environment traceable to objectives and meet them
   - This section provides assurance evidence that PP is complete, consistent, technically sound

©2004 Matt Bishop

Security Target

- CC Security Target (ST): set of security requirements and specifications to be used as basis for evaluation of identified product or system
  - Can be derived from a PP, or directly from CC
    - If from PP, ST can reference PP directly
  - Addresses issues for *specific* product or system
    - PP addresses issues for a family of potential products or systems

©2004 Matt Bishop

How It Works

- Find appropriate PP and develop appropriate ST based upon it
  - If no PP, use CC to develop ST directly
- Evaluate ST in accordance with assurance class ASE
  - Validates that ST is complete, consistent, technically sound
- Evaluate product or system against ST

©2004 Matt Bishop

Form of ST

1. Introduction
   - ST Identification, ST Overview
   - CC Conformance Claim
     - Part 2 (or part 3) conformant if all functional requirements are from part 2 (or part 3) of CC
     - Part 2 (or part 3) extended if uses extended requirements defined by vendor as well
2. Product or System Description
   - Describes TOE as aid to understanding its security requirement

©2004 Matt Bishop

Form of ST (*con't*)

3. Product or System Family Security Environment
4. Security Objectives
5. IT Security Requirements
   - These are the same as for a PP

©2004 Matt Bishop

Form of ST (*con't*)

6. Product or System Summary Specification
   - Statement of security functions, description of how these meet functional requirements
   - Statement of assurance measures specifying how assurance requirements met
7. PP Claims
   - Claims of conformance to (one or more) PP requirements

©2004 Matt Bishop

Form of ST (*con't*)

8. Rationale
   - Security objectives rationale demonstrates stated objectives traceable to assumptions, threats, policies
   - Security requirements rationale demonstrates requirements for TOE and environment traceable to objectives and meets them
   - TOE summary specification rationale demonstrates how TOE security functions and assurance measures meet security requirements
   - Rationale for not meeting all dependencies
   - PP claims rationale explains differences between the ST objectives and requirements and those of any PP to which conformance is claimed

©2004 Matt Bishop

CC Requirements

- Both functional and assurance requirements
- EALs built from assurance requirements
- Requirements divided into *classes* based on common purpose
- Classes broken into smaller groups (*families*)
- Families composed of *components*, or sets of definitions of detailed requirements, dependent requirements and definition of hierarchy of requirements

©2004 Matt Bishop

Security Functional Requirements

©2004 Matt Bishop

SSE-CMM: 1997–Present

- Based on Software Engineering Capability Maturity Model (SE-CMM or just CMM)
- Defines requirements for *process* of developing secure systems, not for systems themselves
  - Provides maturity levels, not levels of trust
  - Used to evaluate an organization's security engineering

©2004 Matt Bishop

SSE-CMM Model

- *Process capability*: range of expected results that can be achieved by following process
  - Predictor of future project outcomes
- *Process performance*: measure of actual results
- *Process maturity*: extent to which a process explicitly defined, managed, measured, controlled, and is effective
- Divides process into 11 areas, and 11 more for project and organizational practices
  - Each process area contains a goal, set of base processes

©2004 Matt Bishop

Process Areas

- Process areas:
  - Administer security controls
  - Assess impact, security risk, threat, vulnerability
  - Build assurance argument
  - Coordinate security
  - Monitor system security posture
  - Provide security input
  - Specify security needs
  - Verify, validate security
- Practices:
  - Ensure quality
  - Manage configuration, project risk
  - Monitor, control technical effort
  - Plan technical effort
  - Define, improve organization's systems engineering process
  - Manage product line evolution
  - Provide ongoing skills, knowledge
  - Coordinate with suppliers

©2004 Matt Bishop

Example: Assess Threat

- Goal: threats to the security of the system will be identified and characterized
- Base processes:
  - Identify natural, man-made threats
  - Identify threat units of measure
  - Assess threat agent capability, threat likelihood
  - Monitor threats and their characteristics

©2004 Matt Bishop

Capability Maturity Levels

- *Performed informally*: perform base processes
- *Planned and tracked*: address project-level definition, planning, performance, verification issues
- *Well-defined*: focus on defining, refining standard practice and coordinating it across organization
- *Quantitatively controlled*: focus on establishing measurable quality goals, objectively managing their performance
- *Continuously improving*: improve organizational capability, process effectiveness

©2004 Matt Bishop

Using the SSE-CMM

- Begin with process area
  - Identify area goals, base processes
  - If all processes present, determine how mature base processes are
    - Assess them against capability maturity levels
    - May require interacting with those who use the base processes
  - Do this for each process area
    - Level of maturity for area is *lowest* level of the base processes for that area
    - Tabular representation (called *Rating Profile*) helps communicate results

©2004 Matt Bishop

Key Points

- First public, widely used evaluation methodology was TCSEC (Orange Book)
  - Criticisms led to research and development of other methodologies
- Evolved into Common Criteria
- Other methodologies used for special environments

©2004 Matt Bishop