CSE 543
Information Assurance and Security
IA Management, Evaluation Systems, and Formal Methods

Professor Stephen S. Yau

Spring 2017

Why Need IA Management?

- Many managers tend to overlook IA since it is not directly related to their revenue
- Two basic factors affecting competition for resources
  - Value of your products (including services) to customers
  - Cost of providing them
- IA management staff needs to ***persuade senior managers*** that IA comes with a price tag, and has a return for saving cost for damages due to information lost or misused
- ***Outsourcing*** is more popular, but it may bring in more threats and vulnerabilities

IA Management Personnel

- **Information Systems Security Officer (ISSO)**
  - Responsible to DAA who ensures that security of an information system is implemented properly and throughout its entire life cycle
- **Operation Security (OPSEC) Manager**
  - Responsible to ISSO who prevents sensitive information from being available to potential adversaries
- **System Manager**
  - Responsible for proper operations and management of classified and unclassified Automated Information System (AIS).
  - Supervises system staff in implementing AIS security policies, provides advices, and supports to ISSO on AIS security issues.

IA Management Personnel (cont.)

- **Program or Functional Manager**
  - Responsible for determining, with system manager, which users have verified needs to access their applications.
  - Responsible for informing ISSO of any security incidents related to the application or the users of the application.
- **Communication Security (COMSEC) Custodian**
  - Responsible for the receipt, transfer, accounting, safeguarding and destruction of COMSEC material assigned to a COMSEC account.
- **Telecommunications Officer**
  - Responsible for receipt, transfer, accounting, safeguarding telecommunication processes in organization

Challenges for IA Management

- ***Increasing complexity*** of systems, networks, and interconnectivity
- ***More reliance*** on information and information systems, which rapidly be improved and more complicated and powerful
- Ever-changing internal and external ***threats***
- Increasing ***competing demands***
- Unavailable r***esource***s
- Decreasing ***assets***
- Lack of ***experience***
- Lack of training
- Lukewarm ***support from management***

IA Management Tasks

- ***Managing resources***
- ***Coordination***
- ***Budgeting, including possible outsourcing***
- ***Selling the need***
- ***Dispensing technical guidance:*** A written regulation or directive or policy can ensure consistency between process and standard operating procedure
- ***Dealing with legal issues:*** IA manager should be familiar with applicable legal issues in order to know when it is appropriate and necessary to contact a law enforcement agency in the event of security incident.

Life-cycle Management

- ***Initiation:*** Determine how required operational functions can be accomplished in a secure manner
- ***Definition:*** The functions of the system will determine the security requirements
- ***Design:*** Security requirements, including risk, cost, operations, must be integrated in system design
- ***Acquisition:*** IA manager must ensure that only reliable sources are used for software procurement
- ***Development:*** Security controls are built into the system

Life-cycle Management (cont.)

- ***Implementation:*** Incorporating the following:
  - ***Risk Management***
  - ***C&A Process:*** Certification and Accreditation
  - ***Approval to Operate (ATO):*** Upon successful security evaluation of the system, IA manager ***recommends*** to the DAA that ***ATO or Interim approval to operate (IATO)*** should be granted. IATO is a temporary approval pending an accreditation decision.
  - ***Operation and Maintenance:*** Once the system has been turned on for operation, security of the system must be scrutinized to verify that it ***continues*** to meet requirements
  - ***Destruction and Disposal:*** Ensure that information processed and stored in the system is not inadvertently compromised because of improper destruction and disposal.

Security Review and Testing

- Conducted throughout ***system life-cycle:***
- Common process:
  - Review security policies, documents, patches and updates,
  - Develop security matrix summarizing threats and protected assets
  - Review audit capability and use
  - Run analysis tools
  - Correlate all information
  - Develop reports
  - Make recommendations to correct problems

Identify Weaknesses in a System

- ***Vulnerability scanning:*** Scan for unused ports, unauthorized software
- ***Discovery scanning:*** Inventory and classification of information on OS and available ports, identification of running applications to determine device functions
- ***Workstation scanning:*** Make sure standard software configuration is current with latest security patches, locate unauthorized software
- ***Server scanning:*** Make sure that software stored on server is updated with latest security patches, locate uncontrolled or unauthorized software
- ***Port scanning:*** Scan various active ports used for communication (TCP/UDP)
  - Stealth scans: also called spoofed scans
- Vulnerability ***testing***
  - False positives and false negative
  - Heavy traffic
  - System crash
  - Unregistered port numbers

Methods to Promote Awareness

- Periodic awareness sessions to orient new employees and refresh senior employees which are direct, simple and clear
- Live/interactive presentations thorough lectures and videos
- Publishing/distributing posters, company newsletters
- Incentives: awards and recognition for security- related achievement
- Reminders

Training

- Training often held in specific classroom or through one-on-one training
- InfoSec examples:
  - ***Security-related job training*** for operators and specific users
  - ***Awareness training*** for specific departments or personnel groups with security-sensitive positions
  - ***Technical security training*** for IT support personnel and system administrators
  - ***Advanced InfoSec training*** for security practitioners and auditors
  - ***Security training*** for senior managers, functional managers

Summary

- Ensure ***security*** is planned and developed into all new systems
- ***Certify*** security features performing properly before system operate
- ***Approve and verify configuration changes*** to IA baseline that changes do not affect the terms of the system's accreditation.
- ***Assess the status of security features and system vulnerabilities*** through manual and automated reviews
- Dispose hardcopy printouts and nonvolatile storage media in a way that eliminates possible compromise of sensitive data
- Keep system documentation current, reflecting patches, version upgrades, and other baseline changes
- Ensure that HW/SW changes are approved and tested before installation and operation; IA manager is part of approval process

Evaluation for Functionality and Assurance

- A process in which the evidence for assurance is gathered and analyzed against criteria for functionality and assurance.
- Can result in a measure of ***trust***, indicating how well a system meets selected criteria
  - A system is trusted if it has been shown to meet users' security requirements under specific conditions
  - Trust is based on ***assurance evidence***

Evaluation for Functionality and Assurance (cont.)

- An evaluation methodology provides the following features:
  - A set of **requirements** defining security functionality
  - A set of ***assurance requirements*** specifying ***required evidence of assurance***
  - ***A methodology*** for determining whether the security requirements are satisfied based on assurance evidence.
  - A ***measure*** of the evaluation result (called a level of trust) indicating how ***trustworthy*** the product or system is

Trusted Computer System Evaluation Criteria (TCSEC)

- Developed in 1983-1999 by DoD
  - Also known as the ***Orange Book***
- Emphasizes ***confidentiality***, especially protection of government classified information
- Limitations:
  - Focus on security needs of U.S. government and military
  - ***Not address integrity, availability or other requirements critical to business applications***

Information Technology Security Evaluation Criteria (ITSEC)

- Developed in 1991-2001 by European Union
- Major distinction between TCSEC and ITSEC
  - ITSEC emphasizes ***integrity and availability***, while TCSEC emphasizes ***confidentiality***
- Impact:
  - Can be used to evaluate any kinds of products or systems
- Limitations:
  - Considered technically weak compared to TCSEC
  - Not used in Canada and US

Common Criteria (CC)

- Developed by Canada, France, Germany, Netherlands, United Kingdom and United States, starting 1998
  - Latest revision is Version 3.1 Revision 4 released in ***September 2012***
- An ***international standard***, also known as ***ISO 15408***
- Combines best features of TCSEC, ITSEC and FC
- Provides a common language and structure to express both security functional requirements and security assurance requirements
- Limitation:
  - Protection profile used in CC may not be as strong as TCSEC

Federal Criteria (FC)

- Developed by NIST and NSA
  - FC never completed (the last draft version was released in 1992), but was supplanted by ***Common Criteria*** in 1998
  - Many ideas of FC were adopted by the Common Criteria.
    - The concept of ***protection profile (PP)***, which is an abstract specification of the security aspects of an IT product
    - The concept of ***profile registry***, which is a collection of FC-approved protection profiles available to public for general use

References

- Federal Criteria

http://csrc.nist.gov/drivers/documents/Federal-IT-Security-Assessment-Framework.pdf

- Common Criteria

http://www.commoncriteriaportal.org/cc/

System Security Engineering – Capability Maturity Model (SSE-CMM)

- Development started in 1997 by US
- The SSE-CMM is now ***ISO Standard 21827***
  - The last version was released in 2008
- A process-oriented methodology for developing secure systems based on Software Engineering Capability Maturity Model (SE-CMM)
- Can be used to assess the ***capabilities of security engineering processes of an organization and provide guidance in designing and improving them***
- Limitation: Analysis of processes is complex

Security Evaluation – Formal Methods

- A ***formal method*** means a method which has a ***mathematical foundation***, and thus employs techniques and tools based on mathematics that support modeling, specification, and verification for hardware, software, systems, etc,
- A ***formal approach*** to security is the employment of a formal method in analyzing the security of a given information system or constructing a secure one.
- Formal methods can be applied at ***various levels of abstraction and during various development phases.***

Security Evaluation– Applications of Formal Methods

- ***Objective***: More precisely determine requirements and analyze the system so that security incidents can be prevented (or at least identified).
- Steps in using formal methods for security:

  1.System Specification: Abstraction and modeling with a well-defined syntactic and semantic structure. It ***documents*** how the system operates or should operate.

  2. Requirement Specification: Security modeling (e.g., BLP model). It documents the security requirements ***unambiguously***

  3.Verification: It can be formally done to validate the system with respect to its requirements, including

  - ***Model checkin***g (by searching the satisfiability of the given characteristics of the system in the possible models)
  - ***Theorem proving*** (by inference of the given characteristics of the system using syntactical inference rules in theory proving)
- Formal methods can be applied to part of the three steps, and/or certain critical parts of the system.

Formal Methods – Modeling

- ***Abstract representations*** of a system using mathematical entities and concepts
- ***Model*** should capture essential system characteristics and ignore irrelevant details
- ***Model*** can be used for ***mathematical reasoning*** to prove system properties or predict new behavior
- Two types of models: continuous and discrete
- Formal specification model does the following,
  - Clarify requirements and high level design
  - Articulate implicit assumptions
  - Identify undocumented or unexpected assumptions
  - Expose defects
  - Identify exceptions
  - Evaluate test coverage

Formal Methods – Generating Formal Specifications

- Need to translate non-mathematical description (diagrams, table, natural language) into a ***formal specification language***
- The specification is a concise and precise description of high-level behavior and properties of a system
- ***Well-defined language semantics*** are needed to support formal deduction of specification
- Types of formal specifications,
  - ***Model oriented*:** Based on a model of the system behavior in terms of mathematical objects, like sets, sequences etc.
    - Statecharts, SCR (Software Cost Reduction), VDM (Vienna Development Method)
    - Petri nets, automata theoretic models
  - ***Property oriented***: Based on a set of properties sufficient to describe system behavior in terms of axioms, rules, etc.
    - Algebraic semantics
    - Temporal logic

Formal Method – Role in System Design and Engineering

- Motivated by the expectation that performing appropriate mathematical analysis can contribute to the reliability and robustness of an information system design
- Formal specification of an information system may be used as a guide while the system is developed.
  - If the formal specification is in ***operational semantics*** (executable), the observed behavior of the system can be compared with the behavior of the specification.
  - If the formal specification is in ***axiomatic semantics***, the pre-conditions and post-conditions of the specification may become assertions in the executable code.*

* https://en.wikipedia.org/wiki/Formal_methods

Formal Methods – Bell-LaPadula (BLP) Model

- For ***enforcing*** ***access control*** in information systems and built on the concept of a ***state machine with allowable states in a computer system***.
- The model defines ***two MAC rules and one DAC rule with three security properties:***
  - Simple Security Property - a subject at a given security level ***may not read*** an object at a higher security level (**no read-up**)
  - ★-property (read "star"-property) - a subject at a given security level ***must not write*** to any object at a lower security level (**no write-down**)
  - Discretionary Security Property - use of an access matrix to specify the discretionary access control

http://en.wikipedia.org/wiki/Bell%E2%80%93LaPadula_model

Limitations of Formal Methods

- Requires sound mathematical knowledge of the developer
- Different aspects of a design may be represented by different formal specification methods
- Useful for ***consistency checks***, but cannot guarantee the ***completeness*** of a specifications
- For majority of systems, formal methods do not offer significant cost or quality advantages over others