Lecture 2: Introduction to Computer Security

RK Shyamasundar

Dangers Being Protected Against

- Damage to information
- Disruption of service
- Theft of physical resources like money
- Theft of information
- Loss of privacy

- Integrity
- Availability
- Integrity
- Secrecy (confidentiality)
- Secrecy (confidentiality)

Taxonomy of Cybersecurity Threats

- Incomplete, inquisitive, and unintentional blunders.
- Hackers driven by technical challenges.
- Disgruntled employees or customers seeking revenge.
- Criminals interested in personal financial gain, stealing services, or industrial espionage.
- Organized crime with the intent of hiding something or financial gain.
- Organized terrorist groups attempting to influence U.S. policy by isolated attacks.
- Foreign espionage agents seeking to exploit information for economic, political, or military purposes.
- Tactical countermeasures intended to disrupt specic weapons or command structures.
- Multifaceted tactical information warfare applied in a broad orchestrated manner to disrupt a major military mission.
- Large oganised groups or nation-states intent on overthrowing a government.

Variants of confidentiality

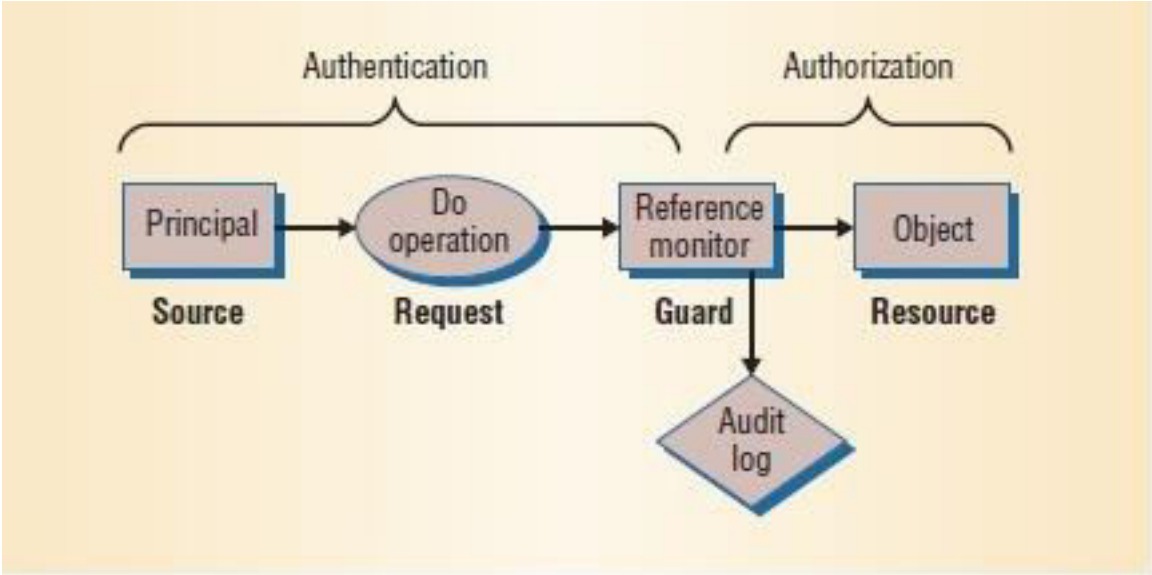
- Data protection/personal data privacy – fair collection and use of personal data, in Europe a set of legal requirements
- Anonymity/untraceability – ability to use a resource without disclosing identity/location
- Unlinkability – ability to use a resource multiple times without others being able to link these uses together
  - HTTP "cookies" and the Global Unique Document Identifier (GUID) in Microsoft Word documents were both introduced to provide linkability.
- Pseudonymity – anonymity with accountability for actions.
- Unobservability – ability to use a resource without revealing this activity to third parties
  - low probability of intercept radio, steganography, information hiding
- Copy protection
- Information flow control- ability to control the use and flow of information
- Further details: Pfitzmann/Kohntopp: http://www.springerlink.com/link.asp?id=xkedq9pftwh8j752

MECHANISM: IMPLEMENTING SECURITY

- Security Implementation:
  - Code: The actual program on which the security depends
  - Setup: data that controls the programs' operations: folder structure, access control lists, group memberships, user passwords or encryption keys, and so on.
- Implementation must defend against:
  - Bad, buggy and hostile vulnerabilities

Broad Defensive Startegies

- Isolate—keep everybody out
  - coarse-grained strategy provides the best security, but it keeps users from sharing info. or services.
  - impractical for all but a few applications.
- Exclude—keep the bad guys out
  - Medium grained strategy makes it all right for programs inside this defense to be gullible. Code signing and firewalls do this.
- Restrict—let the bad guys in, but keep them from doing damage.
  - Fine-grained strategy, also known as sandboxing, can be implemented traditionally with an OS process or with a more modern approach that uses a Java virtual machine.
  - Sandboxing typically involves access control on resources to define the holes in the sandbox. Programs accessible from the sandbox must be paranoid, and it's hard to get this right.
- Recover—undo the damage.
  - Exemplified by backup systems and restore points, doesn't help with secrecy, but it does help with integrity and availability.
- Punish—catch the bad guys and prosecute them.
  - Auditing and police do this.


Authentication
Authorization
Principal
Do operation
Reference monitor
Object
Source
Request
Guard
Resource
Audit log

ASSURANCE: MAKING SECURITY WORK

- Trusted Computing Base (TCB):
  - collection of hw, sw, and setup information on which a system's security depends.
  - if the security policy for a LAN's machines mandates that they can access the Web but no other Internet services, and no inward access is allowed, the TCB is just the firewall that allows outgoing port 80 TCP connections but no other traffic.
  - If the policy also states that no software downloaded from the Internet should run, the TCB also includes the browser code and settings that disable Java and other software downloads.

TCB

- is closely related to the end-to-end principle—just as reliability depends only on the ends, security depends only on the TCB.
- In either, performance and availability aren't guaranteed.
- **Unfortunately**, it's hard to figure out what is in the TCB for a given security policy.
- **Even writing the specs for the components is hard.**

Safety Critical Systems Vs Security

- Sometimes you do a top-down development. In that case you need to get the security spec right in the early stages of the project
- More often it's iterative. Then the problem is that the security requirements get detached
- In the safety-critical systems world there are methodologies for maintaining the safety case
- In security engineering, the big problem is often maintaining the security requirements, especially as the system – and the environment – evolve

Defense-in Depth

- through redundant security mechanisms is a good way to make defects in the TCB less harmful.
- Eg., a system might include
  - Network-level security, using a firewall
  - OS or VM security that uses sandboxing to isolate programs
  - Application-level security that checks authorization directly
- An attacker must find and exploit flaws in all the levels.
- Defense in depth offers no guarantees, but it does seem to help in practice.

END-TO-END ACCESS CONTROL

- Secure distributed systems need a way to handle authentication and authorization uniformly throughout the Internet.
- **Local access control**: like OS,…
- **Distributed Access Control**:
  - A distributed system can involve systems and people that belong to different organizations and are managed differently
  - Eg., A, an Infosys employee, belongs to a team working on a joint microsoft project called GOI. She logs in, using a smart card to authenticate herself, and uses SSL to connect to a project Web page at Microsoft called INDIA. The Web page grants her access according to a given process – may be several steps using SSL, private key , authentication mechanisms….
- **Chains of Trust**

Terminologies, Notation, Clarifications …

Security: Types

- **Computational security** – The most efficient known algorithm for breaking a cipher would require far more computational steps than any hardware available to an opponent can perform.
- **Unconditional security** – The opponent has not enough information to decide whether one plaintext is more likely to be correct than another, even if unlimited computational power were available.
- **Perfect secrecy** means that the cryptanalyst's a-posteriori probability distribution of the plaintext, after having seen the ciphertext, is identical to its a-priori distribution. In other words: looking at the ciphertext leads to no new information.

Cryptology
= Cryptography + Cryptanalysis

- ciphertext-only attack – the cryptanalyst obtains examples of ciphertext and knows some statistical properties of typical plaintext
- known-plaintext attack – the cryptanalyst obtains examples of ciphertext/plaintext pairs
- chosen-plaintext attack – the cryptanalyst can generate a number of plaintexts and will obtain the corresponding ciphertext
- adaptive chosen-plaintext attack – the cryptanalyst can perform several chosen-plaintext attacks and use knowledge gained from previous ones in the preparation of new plaintext

Clarifying Terminology (Anderson)

Clarifying terminology

- A *system* can be:
  - a product or component (PC, smartcard,…)
  - some products plus O/S, comms and infrastructure
  - the above plus applications
  - the above plus internal staff
  - the above plus customers / external users
- Common failing: policy drawn too narrowly

Clarifying terminology (2)

- A *subject* is a physical person
- A *person* can also be a legal person (firm)
- A principal can be
  - a person
  - equipment (PC, smartcard)
  - a role (the officer of the watch)
  - a complex role (Alice or Bob, Bob deputising for Alice)
- The level of precision is variable – sometimes you need to distinguish ‘Bob’s smartcard representing Bob who’s standing in for Alice’ from ‘Bob using Alice’s card in her absence’. Sometimes you don’t

Clarifying terminology (3)

- *Secrecy* is a technical term – mechanisms limiting the number of principals who can access information
- *Privacy* means control of your own secrets
- *Confidentiality* is an obligation to protect someone else's secrets
- Thus your medical privacy is protected by your doctors' obligation of confidentiality

Clarifying terminology (4)

- *Anonymity* is about restricting access to metadata. It has various flavours, from not being able to identify subjects to not being able to link their actions
- An object's *integrity* lies in its not having been altered since the last authorised modification
- *Authenticity* has two common meanings –
  - an object has integrity plus freshness
  - you're speaking to the right principal

Trust vs Trustworthy (5)

- *Trust* -- complex :
  1. a warm fuzzy feeling
  2. a trusted system or component is one that can break my security policy
  3. a trusted system is one I can insure
  4. a trusted system won't get me fired when it breaks
- NSA definition – number 2 above.
- E.g. an NSA man selling key material to the Chinese is trusted but not trustworthy (assuming his action un-authorised)

Clarifying Terminology (6)

- A *security policy* is a succinct statement of protection goals – typically less than a page of normal language
- A *protection profile* is a detailed statement of protection goals – typically dozens of pages of semi-formal language
- A *security target* is a detailed statement of protection goals applied to a particular system – and may be hundreds of pages of specification for both functionality and testing

What often passes as ‘Policy’

1. This policy is approved by Management.
2. All staff shall obey this security policy.
3. Data shall be available only to those with a ‘need-to-know’.
4. All breaches of this policy shall be reported at once to Security.

???

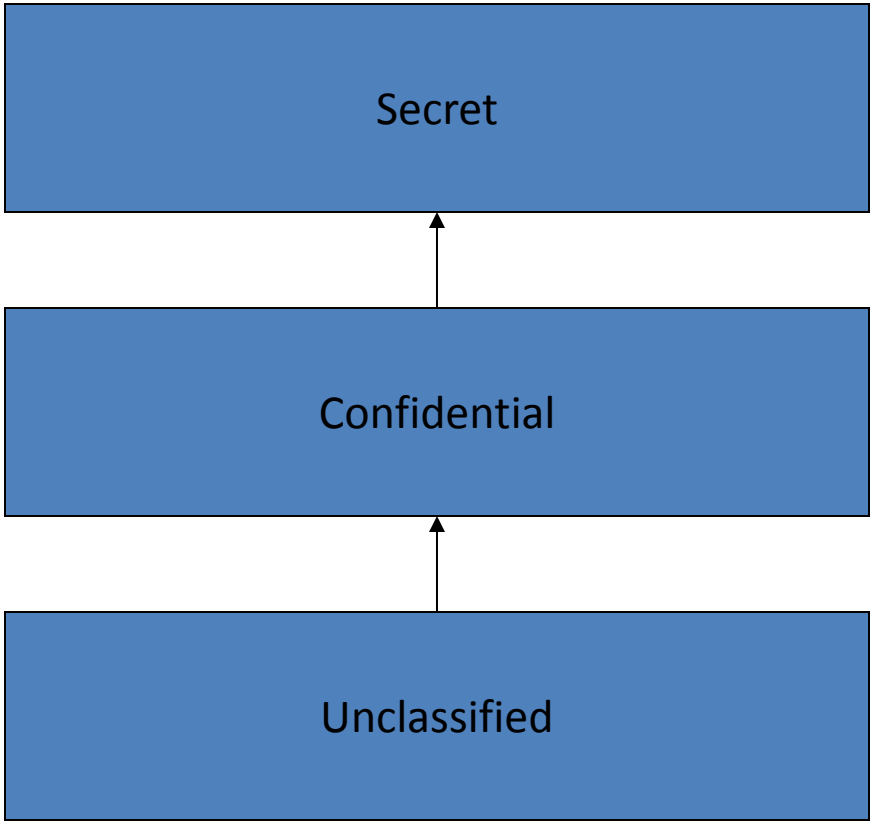
Policy Example – MLS

- Multilevel Secure (MLS) systems are widely used in government
- Basic idea: a clerk with ‘Secret’ clearance can read documents at ‘Confidential’ and ‘Secret’ but not at ‘Top Secret’
- 60s/70s: problems with early mainframes
- First security policy to be worked out in detail following Anderson report (1973) for USAF which recommended keeping security policy and enforcement simple

Levels of Information

- Levels include:
  - Top Secret: compromise could cost many lives or do exceptionally grave damage to operations. E.g. intelligence sources and methods
  - Secret: compromise could threaten life directly. E.g. weapon system performance
  - Confidential: compromise could damage operations
  - Restricted: compromise might embarrass?
- Resources have classifications, people (principals) have clearances. Information flows upwards only

Information Flows



Formalising the Policy

- Initial attempt – WWMCCS
  - Worldwide Military Command and Control System – had rule that no process could read a resource at a higher level. Not enough!
- Bell-LaPadula (1973):
  - *simple security policy*: no read up
  - **-policy*: no write down
- With these, one can prove theorems etc.
- Ideal: minimise the Trusted Computing Base (set of hardware, software and procedures that can break the security policy) in a reference monitor

Limits of firewalls

- Once a host on an intranet behind a firewall has been compromised, the attacker can communicate with this machine by tunnelling traffic over an open protocol (e.g., HTTPS) and launch further intrusions unhindered from there.
- Little protection is provided against insider attacks.
- Centrally administered rigid firewall policies severely disrupt the deployment of new services. The ability to “tunnel” new services through existing firewalls with fixed policies has become a major protocol design criterion. Many new protocols (e.g., SOAP) are for this reason designed to resemble HTTP, which typical firewall configurations will allow to pass.
- Firewalls can be seen as a compromise solution for environments, where the central administration of the network configuration of each host on an intranet is not feasible. Much of firewall protection can be obtained by simply deactivating the relevant network services on end machines directly.

Enforcement (1)

- Monitoring: Because attacks (by definition) involve execution, a second means of defense can be to monitor a set of interfaces and halt execution before any damage is done using operations those interfaces provide. Three elements comprise this defense:
- a security policy, which prescribes acceptable sequences of operations from some set of interfaces;
- a reference monitor, which is a program that is guaranteed to receive control whenever any operation named in the policy is requested, and
- a means by which the reference monitor can block further execution that does not comply with the policy.

Enforcement (2)

- Principle of Complete Mediation: The reference monitor intercepts every access to every object
- Principle of Least Privilege. A principal should be only accorded the minimum privileges it needs to accomplish its task
  - impossible to implement if the same privilege suffices for multiple different objects or operation
- **Principle of Separation of Privilege. Different accesses should require different privileges.**
- Principle of Failsafe Defaults. The presence of privileges rather than the absence of prohibitions should be the basis for determining whether an access is allowed to proceed

Additional References Lectures 1-2

- Butler Lampson “ Computer Security in the Real World”, IEEE Computer, June 2004, pp.37-46
- Fred Schneider , Introduction, Excerpt from an as yet untitled work in progress. Draft of August 2007. http://www.cs.cornell.edu/fbs/publications/chptr.Intro.pdf
- Markus Kuhn, Introduction to Security, Security Univ of Cambridge

References Lectures 1

- Butler Lampson " Computer Security in the Real World", IEEE Computer, June 2004, pp.37-46
- Fred Schneider , Introduction, Excerpt from an as yet untitled work in progress. Draft of August 2007. http://www.cs.cornell.edu/fbs/publications/chptr.Intro.pdf
- Markus Kuhn, Introductiont o Security, SecurityUniv of Cambridge
- Books: Security Engg: Ross Anderson
- Cryptography and Data Security, D Denning

Security Management

Security Management and Engineering

Is this product/technique/service secure?

- Simple Yes/No answers are often wanted, but typically inappropriate.
- Security of an item depends much on the context in which it is used.
- Complex systems can provide a very large number of elements and interactions that are open to abuse. An effective protection can therefore only be obtained as the result of a systematic planning approach.

"No need to worry, our product is 100% secure. All data is encrypted with 128-bit keys. It takes billions of years to break these." : Such statements are abundant in marketing literature.

A security manager should ask:

- What does the mechanism achieve?
- Do we need confidentiality, integrity or availability of exactly this data?
- Who will generate the keys and how?
- Who will store / have access to the keys?
- Can we lose keys and with them data?
- Will it interfere with other security measures (backup, auditing, scanning, . . .)?
- Will it introduce new vulnerabilities or can it somehow be used against us?
- What if it breaks or is broken?

Security policy development

- Step 1: Security requirements analysis
- Identify assets and their value
- Identify vulnerabilities, threats and risk priorities
- Identify legal and contractual requirements

Step 2: Work out a suitable security policy

The security requirements identified can be complex and may have to be abstracted first into a high-level security policy, a set of rules that clarifies which are or are not authorised, required, and prohibited activities, states and information flows.

Security policy models are techniques for the precise and even formal definition of such protection goals. They can describe both automatically enforced policies (e.g., a mandatory access control configuration in an operating system, a policy description language for a database management system, etc.) and procedures for employees (e.g., segregation of duties).

Step 3: Security policy document

Once a good understanding exists of what exactly security means for an organisation and what needs to be protected or enforced, the highlevel security policy should be documented as a reference for anyone involved in implementing controls. It should clearly lay out the overall objectives, principles and the underlying threat model that are to guide the choice of mechanisms in the next step.

Step 4: Selection and implementation of controls

Issues addressed in a typical low-level organisational security policy:

- General (affecting everyone) and specific responsibilities for security
- Names manager who “owns” the overall policy and is in charge of its continued enforcement, maintenance, review, and evaluation of effectiveness
- Names individual managers who “own” individual information assets and are responsible for their day-to-day security
- Reporting responsibilities for security incidents, vulnerabilities, software malfunctions
- Mechanisms for learning from incidents
- Incentives, disciplinary process, consequences of policy violations
- User training, documentation and revision of procedures
- Personnel security (depending on sensitivity of job) Background checks, supervision, confidentiality agreement
- Regulation of third-party access
- Physical security: Definition of security perimeters, locating facilities to minimise traffic across perimeters, alarmed fire doors, physical barriers that penetrate false floors/ceilings, entrance controls, handling of visitors and public access, visible identification, responsibility to challenge unescorted strangers, location of backup equipment at safe distance, prohibition of recording equipment, redundant power supplies, access to cabling, authorisation procedure for removal of property, clear desk/screen policy, etc.

- Segregation of duties: Avoid that a single person can abuse authority without detection (e.g., different people must raise purchase order and confirm delivery of goods, croupier vs. cashier in casino)
- Audit trails: What activities are logged, how are log files protected from manipulation
- Separation of development and operational facilities
- Protection against unauthorised and malicious software
- Organising backup and rehearsing restoration
- File/document access control, sensitivity labeling of documents and media
- Disposal of media: Zeroise, degauss, reformat, or shred and destroy storage media, paper, carbon paper, printer ribbons, etc. before discarding it.
- Network and software configuration management
- Line and file encryption, authentication, key and password management
- Duress alarms, terminal timeouts, clock synchronisation, . . .

UK Computer Misuse Act 1990

- Knowingly causing a computer to perform a function with the intent to access without authorisation any program or data held on it ⇒ up to 6 months in prison and/or a fine
- Doing so to further a more serious crime

⇒ up to 5 years in prison and/or a fine

- Knowingly causing an unauthorised modification of the contents of any computer to impair its operation or hinder access to its programs or data ⇒ up to 5 years in prison and/or a fine

The intent does not have to be directed against any particular computer, program or data. In other words, starting automated and selfreplicating tools (viruses, worms, etc.) that randomly pick where they attack is covered by the Act as well. Denial-of-service attacks in the form of overloading public services are not yet covered explicitly.

UK Data Protection Act 1998

Anyone processing personal data must comply with the eight principles of data protection, which require that data must be

1. Fairly and lawfully processed [
   - Person's consent or organisation's legitimate interest needed, no deception about purpose, sensitive data (ethnic origin, political opinions, religion, trade union membership, health, sex life, offences) may only be processed with consent or for medical research or equal opportunity monitoring, etc.
2. Processed for limited purposes
   - In general, personal data can't be used without consent for purposes other than those for which it was originally collected.
3. adequate, relevant and not excessive
4. Accurate
5. Not kept longer than necessary
6. processed in accordance with the data subject's rights:
   - Persons have the right to access data about them, unless this would breach another person's privacy, and can request that inaccurate data is corrected.
7. secure
8. not transferred to countries without adequate protection

- This means, no transfer outside the European Free Trade Area. Special "safe harbour" contract arrangements with data controllers in the US are possible.