

Review Aid for Midterm II

CS 5323 Principles of Information Security

October 9, 2007

Anderson Chapter 1

- Which of the following are security objectives and which are security mechanisms? Confidentiality, privacy, integrity, authenticity, availability, authentication, access control, encryption
- The *trust assumptions* of a system tell us what components are required to perform correctly to ensure security of the systems as a whole.
- Understand the differences between groups and roles.
- Understand the relationships between authenticity, integrity, and freshness.
- *Anonymity* refers to the quality of a message being of unknown origin and/or destination.
- A *vulnerability* is a property of a system or its environment, which, in conjunction with an internal or external *threat*, can lead to a *security failure*, which is a state of affairs contrary to the systems security policy.

Anderson Chapter 2

- What is a *threat model* and what is its relationship to the design and evaluation of a protocol?
- What is a *nonce* and what purpose does it serve in protocol designs? What is a *replay attack*?
- BAN logic is one of the formal methods used for protocol verification. It focuses on justifying the appropriate beliefs of protocol participants as a result of their interactions in the protocol. It makes simplifying assumptions, such as, cryptography cannot be broken.

Gollman Chapter 4

- Access control can be enforced at several levels in the system. It is pretty good at managing *integrity*, but less good at managing *confidentiality*. (Why?)
- What is a *reference monitor*? What does it do and how does it work?
- What is the relationship between subjects and principals? Note that source code can be considered a kind of principal in the sense that based on its estimated trustworthiness, it can be granted appropriate privileges. This is essentially what happens in controlled invocation mechanisms, such as Unix's SUID and SGID programs. It also is what is going on when Java applets downloaded from the internet are run in a sandbox, thus isolating them from the rest of the system.
- Be familiar with the notion of *access modes*. Be aware of the access modes available in Unix, how they are expressed or presented, and what the various modes mean for files and for directories.
- What is the basic difference between *discretionary* access control (DAC) and *mandatory* access control (MAC)?
- What is the access control matrix model and how is it related to access control lists (ACLs) and capabilities? What are some drawbacks of the capabilities model?

- Be familiar with the notion of protection rings. What is the 80x86 security policy based on protection rings? (See Chapter 5.)
- What is the definition of a partial order? What are some examples? What does it mean for a partial order to be a lattice?
- Be familiar with the basic structure of multilevel security (MAC according to the “Orange Book”) including knowing what are classifications and compartments, plus being able to define the order relation on security labels in this model.

Gollman Chapter 5

- Be able to define “reference monitor,” “security kernel,” and “trusted computing base (TCB).” Be aware of some of the ways that a reference monitor can be positioned so as to ensure that access requests can be mediated and prevented if necessary.
- Be aware of the basic notion of an *execution monitor* in the sense of [Schneider, 2000].
- Understand the use of *modes of operation* and *controlled invocation* in the context of operating system security.

Gollman Chapter 6

- Be aware of the roles played by authentication, access control, auditing, and configuration in achieving a secure deployed system.
- In Unix, be aware of the information stored in `/etc/passwd` and `/etc/group`. What is a user’s *primary group*? How is the user added to other groups?
- What is the difference between a process’s *real* UID/GID and its *effective* UID/GID?
- What is an inode? Understand that multiple directories can have references to a single file. The directory entries give the name(s) for the file—the inode does not. The following are stored in the inode: type of file, permissions, owner, group, last access time, last modification time, size of file, physical location of file.
- One method of controlled invocation in Unix is through *set UID* (SUID) and *set GID* (SGID) programs. Understand how the UID and GID, respectively, are effected when these programs are executed. What are some of the risks associated with using this mechanism for controlled invocation?
- What issues need to be considered when attempting to ensure that a sensitive file has been fully removed from a hard drive used in a Unix system?
- Why is it dangerous to put the current directory in your Unix search path?