def**Definition:**

6.857
NETWORK AND COMPUTER SECURITY
LECTURE 1

JOHN WANG

1. Introduction to Cryptography

is computing or communicating in the presence of adversaries.

The presence of adversaries make security interesting, because you're working against the cleverness of other people. You are in the worst case scenario. Note that this is different than error correcting codes, for instance, where there is no adversary.

2. Security Policies

describes what is being protected and what activities or events should be protected.

If you don't have a policy, then you don't have security, because nothing is defined yet. Security policy is usually in terms of:

- Principals (actors or participants).
- Permissible (or inpermissible) actions or operations.
- Classes of objects.

2.1. **Examples.** Security Policy: "Each registered voter may vote at most once." Principals are the voters and permissible actions are voting at most once. Security Policy: "Only an administrator can modify file x." Security Policy: "The recipient of an email should be able to authenticate the sender."

2.2. **Types of Policies.**

- C - confidentiality policies. Prevents unauthorized disclosure.
- I - integrity policies. Information should not be modifiable in an unauthorized manner.
- A - availability policies. Systems should remain available.

3. Security Mechanisms

are means for achieving security policies.

Examples: smart card for voters, password for sysadmin, digital signature for email, physical security.

Security mechanisms are usually one of two forms:

- Prevention: keeps policy from being violated.
- Detection: discovers if the policy has been violated.

If the detection mechanism goes off, then what? You must have a *recovery mechanism* for getting the system back to a good state. Notice that preventation and detection are not entirely unrelated. Detection system may involve deterrence, which helps prevents attacks.

4. Adversaries

The adversaries may be outsider or insider (ex: voter may want to be able to vote twice). Need to figure out who the adversary is. Note that there can be many adversaries.

What do the adversaries know? Usually, you assume that the adversary knows the engineering of the system and the security mechanisms. Security analysis is usually scenario based which is different depending on the assumptions one makes about the adversary and what he/she knows.

What resources does the adversary have? Does the adversary have a supercomputer or the ability to corrupt insiders or mathematical knowledge?

What are his motivations? Is the adversary economically motivated or is he just evil? Sometimes it is useful to assume that adversaries are rational economic players.

The best systems are those which are robust even in the worst-case scenarios. You want to have a system that is secure even when you have a perfect adversary.

5. Vulnerability

is a weakness in the system that can be exploited by the adversary.

Examples: poor password, buffer overflows, etc.

There is a distinction between the system as designed and the system as implemented. Implementations tend to have bugs, which could potentially introduce security vulnerabilities. Even if the design is perfect