Foundations of Computer Security

Lecture 5: Policies and Metapolicies

Dr. Bill Young
Department of Computer Sciences
University of Texas at Austin

How Do We Define Security?

We said that in the most general terms, *security* seems to mean something like “protection of assets against threats.”

But this question is very specific to the context. Security is very different for:

- a wireless phone system
- a military database system
- an online banking system.

If these have different associated notions of security, how do you specify the security requirements of a given system?

Policies

Often, security for a given system is defined in terms of a *security policy*.

A policy is a set of rules for implementing specific security goals.

Another way to think of it is as a *contract* between the system designer/implementor and the customer. It should be both achievable (able to be followed) and adequate for the intended goals.

Thought Experiment

Students' academic records are stored on computers at the university. Design a security policy to protect them.

You might start by asking:

1. What does it mean "to protect them"?
2. What are the potential threats?
3. Which of the following are important here: confidentiality, integrity, availability?
4. Who are the stakeholders, i.e., whose interests are at risk?
5. Do their interests conflict? How are conflicts resolved?

Evaluating A Policy

If a policy is a set of rules, how do we decide if they are the *right* set of rules?

UT Austin has in place a policy with the following rules, among others:

1. Faculty/staff may not use student SSNs in documents/files/postings.
2. Documents containing SSNs must be destroyed unless deemed necessary.
3. Documents containing SSNs and deemed necessary for retention must be kept in secure storage.

These rules only make sense in service to a larger goal. *What is it?*

Metapolicy vs. Policy

A useful distinction is between the *metapolicy* and the *policy*.

metapolicy: The overall security goals of the system.

policy: A system-specific refinement of the metapolicy adequate to provide guidance to developers and users of the system.

If you don't understand the metapolicy, it becomes difficult to justify and evaluate the policy.

Often the metapolicy will be in terms of confidentiality, integrity, and availability; the policy will be in terms of mechanisms like firewalls, encryption, locked drawers, etc.

So Why Have a Policy?

If the “metapolicy” is what we really care about, why bother with the policy at all?

1. The metapolicy is often too general to provide adequate guidance.
2. The metapolicy may be subject to multiple interpretations.
3. There may be multiple acceptable policies that accomplish the security goals.
4. The policy provides specific and enforceable guidelines to the system user/developer.

Some Lessons

1. System security is often characterized in terms of a *security policy*, a set of rules governing activities within the system.
2. The policy will seem arbitrary unless you understand the *metapolicy*, the overarching security goals.

Next lecture: A Policy Example: MLS