

Computer Security Seminar — Lecture 2

Orr Dunkelman

Computer Science Department

15th March, 2015



Outline

1 Security Engineering — Introduction

- Motivation
- Framework
- Example — Airport Security
- Example — A Bank

2 Get to Know Your Adversaries

- Who Acts Adversarially?
- Why to Act Adversarially?
- How to Attack?
- How to Protect?

Computer Security

- ▶ Most engineering fields try to optimize:
 - ▶ Minimal costs (production, deployment, maintenance),
 - ▶ Maximal reuse (chemicals, designs, code snippets),
 - ▶ Safety margins,
- ▶ Safety margins are the outcome of experience and risk assessment processes:
 - ▶ Ground type (the more solid — lower safety margins),
 - ▶ Risk of earth-quacks (the safer — lower safety margins),
 - ▶ Failure “cost” (less users — lower safety margins),
 - ▶ Identification of “wear and tear” (easier identification — lower safety margins)

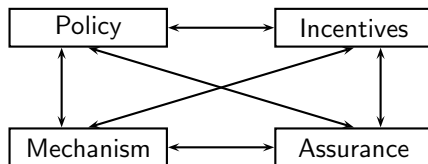
Computer Security (cont.)

- ▶ Security engineering is very different from typical engineering:
 - ▶ The damage is not caused randomly, but is targeted.
 - ▶ The adversary is malicious, rather than “randomly distributed” (e.g., in communication systems).
 - ▶ The adversary looks for the weakest link.
 - ▶ The adversary may have a great deal of resources at his disposal.
- ▶ In addition, the risk assessment process is biased.
 - ▶ We have very little experience with the effects of failed security mechanisms,
 - ▶ The economic incentives are not always aligned correctly.
 - ▶ The working environments of running code changes, leaving “internal” systems open to the “world”.
 - ▶ Security engineering — not a very common practice.

Security Engineering

- ▶ Building dependable systems in face of malice, error, or mischance.
- ▶ Composed of tools, processes, and methods for:
 - ▶ Design,
 - ▶ Implementation,
 - ▶ Testing,
 - ▶ Auditing,
 - ▶ Adaptation,
- ▶ to target a varying set of attacks and adversaries.

The Security Engineering Framework



- ▶ Policy — the intended outcome (security level).
- ▶ Mechanism — how to achieve the security level.
- ▶ Assurance — the trust needed from each mechanism.
- ▶ Incentives — motivating the entities in the “world”.

Example — Airport Security

- ▶ The 9/11 success was due to policy failure (small knives were allowed through security at that time).
- ▶ The policy has changed to ban knives.
- ▶ Now, the policy has changed to ban many “possible” weapons (e.g., umbrellas, liquids).
- ▶ Of course, even a good policy does not cover all cases.
- ▶ Moreover, airport security prefers to “err” to the safe side.
- ▶ Obviously, this approach is wrong.
- ▶ As noted by Freakonomics writers, the total time wasted in these security checks is equivalent to the lives of several tens of people a year. . .
- ▶ Which makes the 6–8 billion US\$/year spent by the TSA a huge waste.

Example — Airport Security (cont.)

- ▶ Other good policies/mechanisms would be:
 - 1 Fortify the cockpits (one time investment).
 - 2 Guarding airports at night.
 - 3 True identification of flyers (and maintaining a database of true suspects).
 - 4 Profiling — identifying which person is more of a threat.
- ▶ However, political, psychological, and moral issues, tend to interfere with these policies.

The Security Evaluation Process

- ▶ To define the policies, we first need to identify the threat model:
 - ▶ What are the assets to protect,
 - ▶ What are the possible threats (and their probabilities),
 - ▶ What are the risks which arise from these threats,
 - ▶ Who is the adversary, and what resources he has at his disposal,
 - ▶ What is the “security budget” (purchase, training, maintenance, interference with usability, etc.)
 - ▶ What are the impacts of applying the policies.
- ▶ Usually the threats are organized in *attack vectors*, which identify the weakness source, and the adversarial plan.

The Security Evaluation Process — Threats

- ▶ Confidentiality, secrecy, and privacy — obtaining access to restricted information.
- ▶ Integrity — changing values or system behavior by unauthorized entities.
- ▶ Availability — preventing access from authorized entities.
- ▶ Destruction — disabling resources.
- ▶ Money stealing/laundrying/hiding — performing illegal/illegitimate actions with money (or equivalent tokens).

Security Analysis of a Typical Bank

- ▶ Identify the systems in use:
 - ▶ Bookkeeping system (teller, branch, county, bank).
 - ▶ Automatic teller machine systems.
 - ▶ Website (information, promotional, users' accounts).
 - ▶ Messaging systems (between branches, banks, stock exchange, etc.)
 - ▶ Alarms in branches.
 - ▶ Identification (account holders, personal, safes).

Threats on the Bookkeeping System

- ▶ Tellers:
 - ▶ “Creative” transaction registration,
 - ▶ Report faulty loses in case of a bank robbery,
 - ▶ “Manipulating” account holders,
- ▶ Accountants:
 - ▶ “Creative” transaction registration,
 - ▶ Embezzlement,
- ▶ Loan agents: Abusing credit supplied by bank,
- ▶ Bookkeeping software developer/system personal:
 - ▶ Installation of backdoors in software/system,
 - ▶ Collaboration with other fraudulent individuals,
 - ▶ Obtaining access codes of other users in the system,

Threats on the Automated Teller Machines

- ▶ “Insiders”:
 - ▶ Developer/system personal,
 - ▶ Bank agents (abusing new bank cards),
- ▶ Account holders: Reporting “unsuccessful” withdrawals.
- ▶ Crooks:
 - ▶ Stealing bank cards and PINs,
 - ▶ Mugging,
 - ▶ Rouge ATM machines deployment,
 - ▶ Stealing an ATM machine.

Who are the Adversaries?

Everyone!

- ▶ Users and insiders.
- ▶ “Old school” hackers.
- ▶ Script Kiddies.
- ▶ Criminals.
- ▶ Terrorists.
- ▶ Countries and superpowers.

Why to Hack/Attack?

- ▶ Fun.
- ▶ Money.
- ▶ Espionage (business/intelligence).
- ▶ Causing damage.
- ▶ Reputation (as an attacker).
- ▶ Hurting reputation (as a defender).
- ▶ Instantiating fear.
- ▶ Cyber warfare.

How to Attack?

- ▶ Social engineering.
- ▶ Wiretapping.
- ▶ Manipulating communications.
- ▶ Manipulating data (at transit or at rest).
- ▶ Physical entry/Inside access.
- ▶ The use of malware (viruses, Trojan horses, worms, ...).
- ▶ (Distributed) Denial of Service.
- ▶ Spam.
- ▶ Targeted attacks.

How to Protect?

- ▶ Physical security.
- ▶ Authentication and identification.
- ▶ Security protocols.
- ▶ Cryptographic tools.
- ▶ Security products (firewalls, proxies, ...).
- ▶ Audit trails.
- ▶ Redundancy.
- ▶ Virtualization.
- ▶ Access control.
- ▶ Failsafe design methodologies.
- ▶ Awareness.
- ▶ Penetration testing.