TOPIC: 6.857 – Computer & Network Security

DATE: 2/4/15

FILE UNDER:

Admin:

Prof. Ronald L. Rivest
TAs: Britt Cyr, David Lazar, Katerina Sotiraki

Lec: Mon & Wed 11:00–12:30 in 4-270
Rec: Fri 11:00–12:30 in 4-270

Handout: http:courses.csail.mit.edu/6.857/2015/~~handouts~~ files/
H01-course-information.pdf
(handed out in paper too)

Outline: Administrivia
Course outline
Introduction to Security

Administrivia: Course info handout (review)

See "The Imitation Game" 2/12/15
(Note discovery of Turing's notes in Bletchley insulation...)

Course outline: Review of last year's topics

Intro to Security: (attached notes)

Content:

"Security" relates to "computing or communicating in the presence of adversaries"

Typically involves an "information system":

PC, network of computers, cell phone, email, ATM machine, car, smart grid, RFID, wireless link, medical device, ...

everything is "digital" now!

Security relates to a "security objective" or "security policy": what is being protected? what activities or events should be prevented/detected?

Security policy usually stated in terms of:

- principals (actors or participants) (perhaps in terms of their roles)
- giving permissible (or impermissable) actions or operations
- on (classes of) objects

Examples: "Each registered voter may vote at most once."
"Only an administrator may modify this file."
"The recipient of an email shall be able to authenticate its sender."

Security policies (goals) often fall into one of three classic categories:

- confidentiality: information should not be disclosed to unauthorized parties
- integrity: information should not be modified in an unauthorized manner
- availability: system or resource shall be available for use as intended

("CIA")

Security mechanism (aka "security control") is a component, technique, or method for (attempting to) achieve or enforce security policy.

Examples: smart card for voter
password for sysadmin
digital signature on email
locked cabinet for server

Security mechanisms are typically one of two forms:

① **prevention:** keep security policy from being violated

Examples: fence, password, encryption, memory bounds check, ...

② **dectection:** detect when policy is violated

Examples: motion sensor, tamper-evident seal, stored fingerprint ("hash") of executables, intrusion detection on network, virus scanner, ...

Detection mechanism often comes with recovery mechanism (remove intruder,
remove virus,
load files from backup, ...)

Detection may involve deterrence
(adversary risks being identified & being held accountable for security breach)
and so plays a role in prevention.

Who is adversary? (Know your enemy!)

- may be insider/outsider, vendor, ...

  Examples: Voter may wish to sell her vote.
  Election official may be corrupt.
  Vendor may install "back door" in system.
  Eavesdropper may manipulate communications.

- what does adversary know?

  Examples: system design & implementation details
  passwords
  facebook profiles of all personnel

- what resources does adversary have?

  Examples:
  - large computers
  - ability to intercept & modify all communications
  - ability to corrupt some participants
    (e.g. payTV subscriber, voter, server, ...)

We typically make generous assumptions about adversary's abilities.

Vocab:

"vulnerability" = weakness that might be exploited by an adversary
(e.g. poor password, buffer overflow possibility)

"threat" = potential violation of security policy
(e.g. by exploiting a vulnerability)

"risk" = likelihood that threat will materialize

"risk management" = balancing one risk against another, or
other factors, such as cost, ease-of-use,
understandability, availability, ...

No mechanism is perfect — we build fences, not
impenetrable walls
(how high is fence?)

Security mechanisms may involve:

- identification of principals (e.g. "user name")
- authentication of principals (e.g. password, biometric)
- authorization: checking to see if principal is authorized for requested action
- physical protection: locks, enclosures
- cryptography: math in service of security (hard computational problems)
- economics: (note model change here: parties are self-interested, e.g. spammer,...)
- deception: to get adersary to reveal himself or waste his efforts (e.g. honeypot)
- randomness, unpredictability: e.g. for passwords & crypto keys

Some principles:

- be sceptical & paranoid
- don't aim for perfection
  ("there are no secure systems, only degrees of insecurity..."
- tradeoff cost / security
  ("to halve the risk, double the cost..." - Adi Shamir)
- be prepared for loss
- "KISS" ("keep it simple, stupid!")
- ease of use is important
- separation of privilege - require 2 people to perform action
- defense in depth (layered defense)
- complete mediation (all requests checked for authorization)
- least privelege (don't give some more permissions than they need)
- education
- transparency (no security through obscurity)