ΥΣ13 - Computer Security

Introduction

Κώστας Χατζηκοκολάκης

Logistics

- Παραδόσεις: Τετάρτη 9-12, αίθουσα ΣΤ
- Διδάσκων: Κ. Χατζηκοκολάκης
  - kostasc@di.uoa.gr
  - Γραφείο Α52
  - Office hours: Δευτέρα 5-6μμ (ή email)
- Course site:
  https://crypto.di.uoa.gr/csec/
- Course forum: https://piazza.com/uoa.gr/spring2019/ys13
  (γραφτείτε άμεσα!)

Logistics

- Βαθμολογία
  - 2-3 projects: 40%
  - Εξέταση: 60%
  - Βάση και στα δύο
- Teaching Assistants
  - Γιώργος Καδιανάκης
  - Θεόδωρος Πολύζος
- Material
  - Ross Anderson, Security Engineering
    https://www.cl.cam.ac.uk/~rja14/book.html
  - Papers, articles, …

Today's topic:

why are we here?
what is computer security?

What is compurity security?

- The task of achieving some goal
- In presence of some adversary that intentionally tries to make us fail
- Regardless of what the adversary is doing
- Essential elements:
  - Security property: confidentiality, integrity, availability, …
  - Threat model: what the adversary knows/is allowed to do
  - Mechanism: ensures that the property is satisfied

What is compurity security?

Why is security hard?

- "Negative" goal: hard to think about all possible adversaries, challenging to test
- Properties are hard to propertly state
- Threat models often miss a serious threat
- Mechanisms are insufficient or broken
- Edge cases are essential

Problems in threat models...

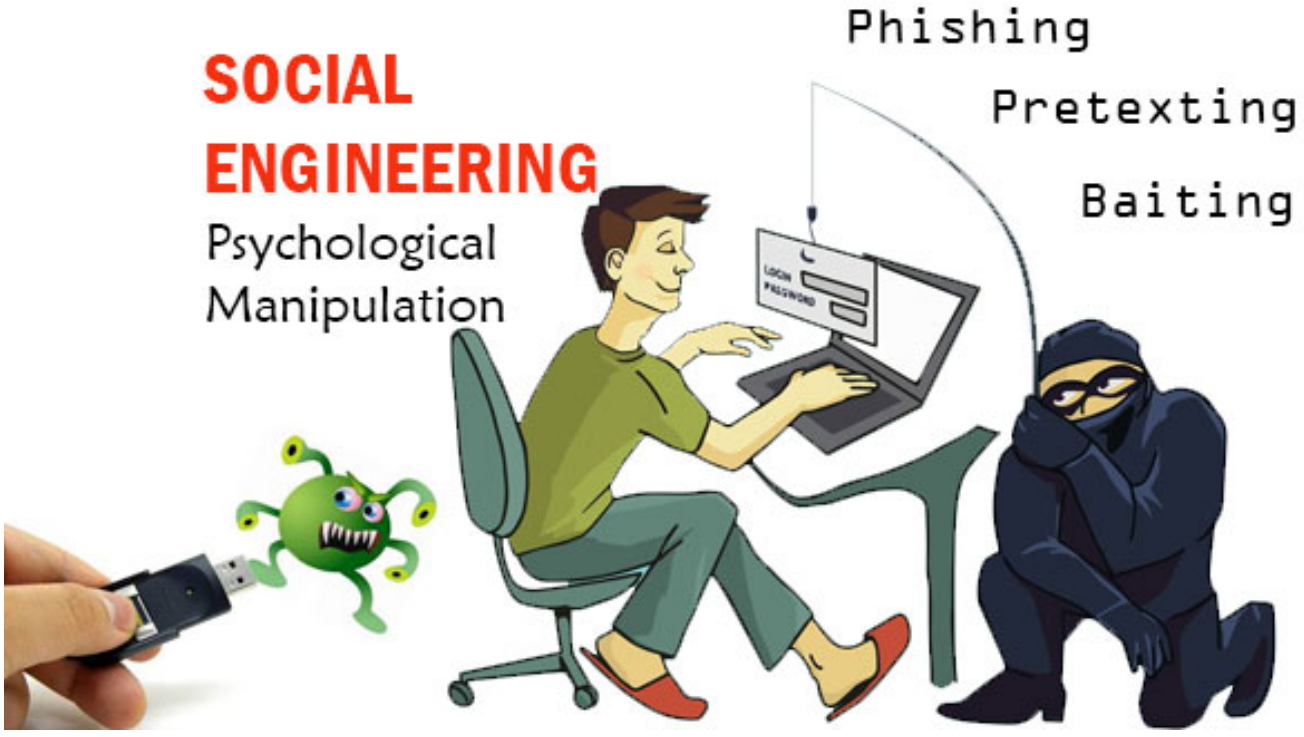
Pet names and passwords are equally hard to guess

Problems in threat models...

A single weak link can be catastrophic

Problems in threat models...

Human factors

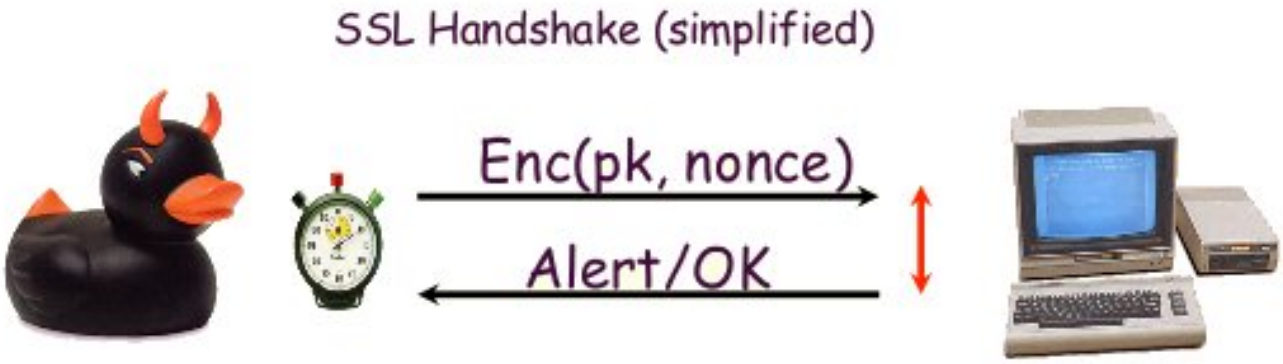


Problems in threat models...

Need to keep up to date

Problems in threat models...

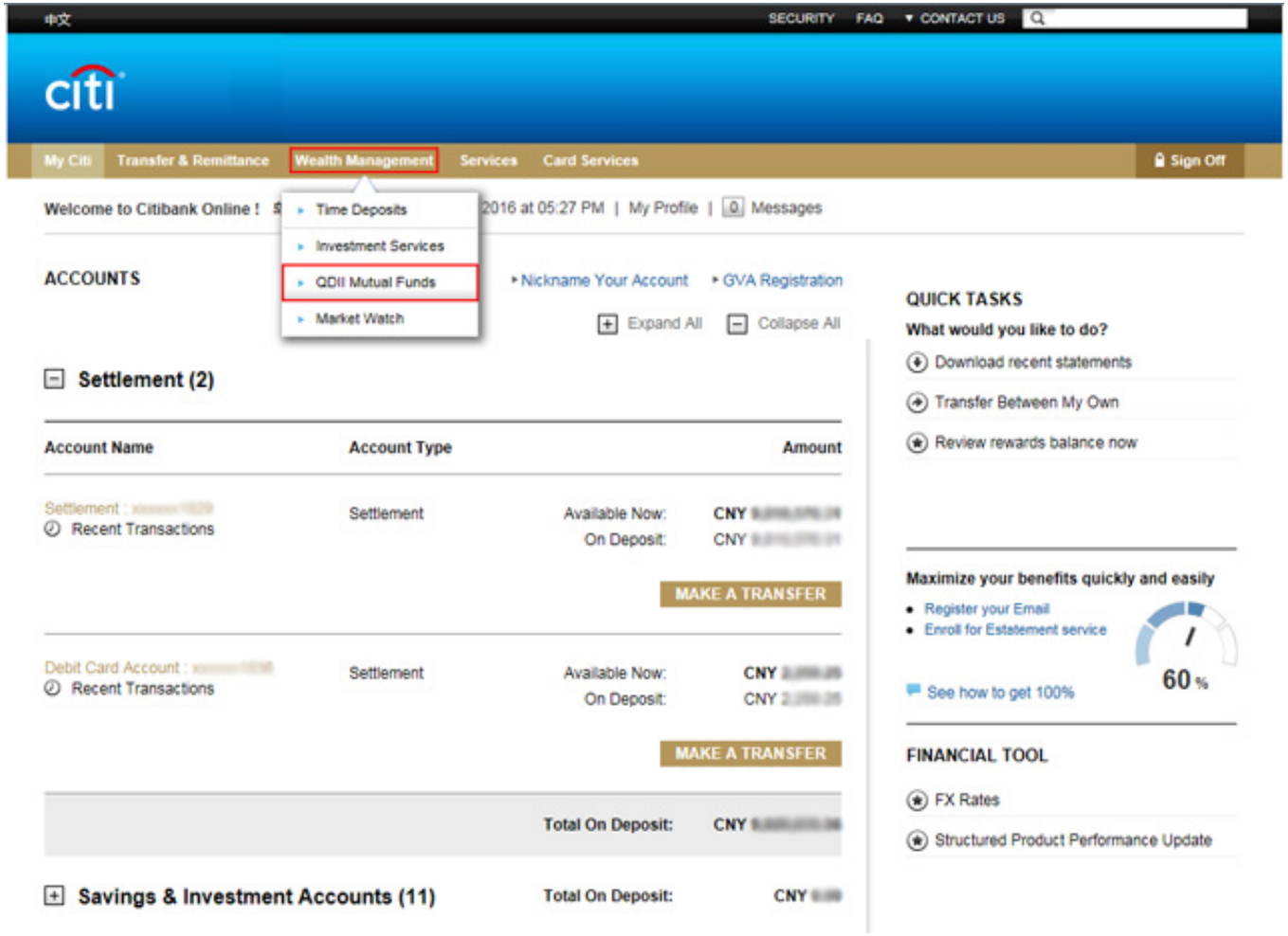
Side channels



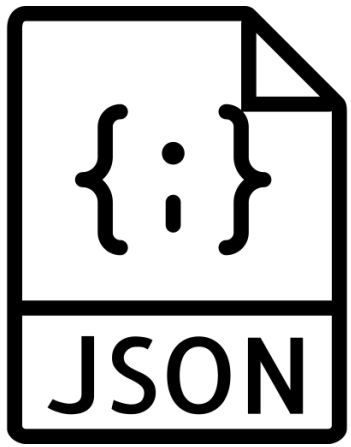
Problems in mechanisms...

https://github.com/hackappcom/ibrute

Problems in mechanisms...

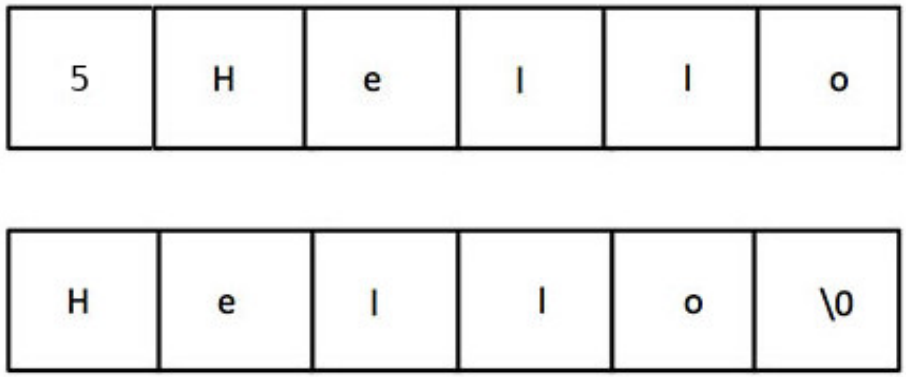


Problems in mechanisms...

eval(

)

Problems in mechanisms...

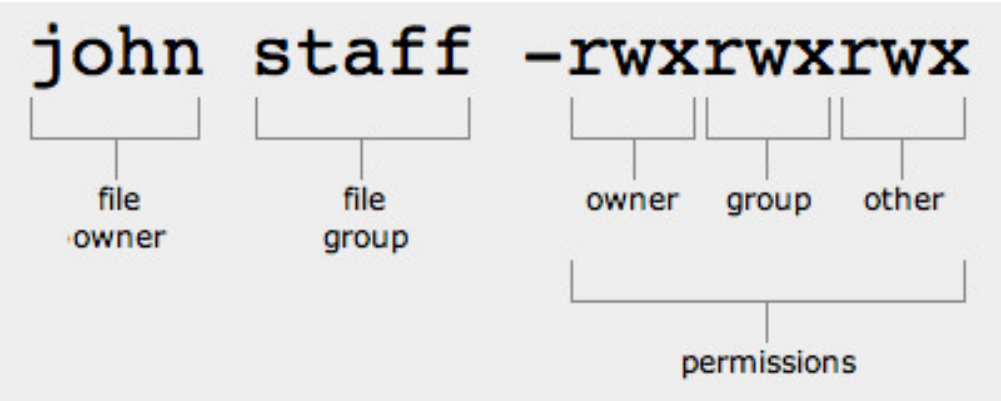


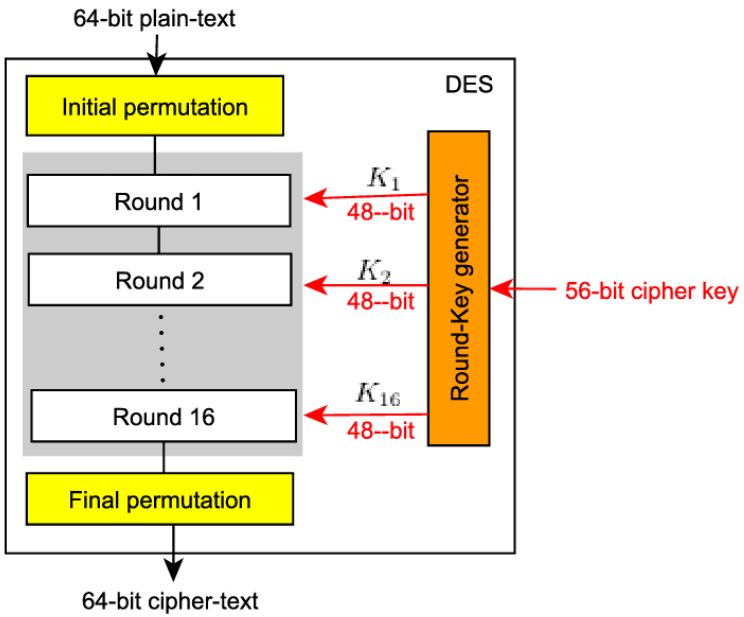
Problems in mechanisms...

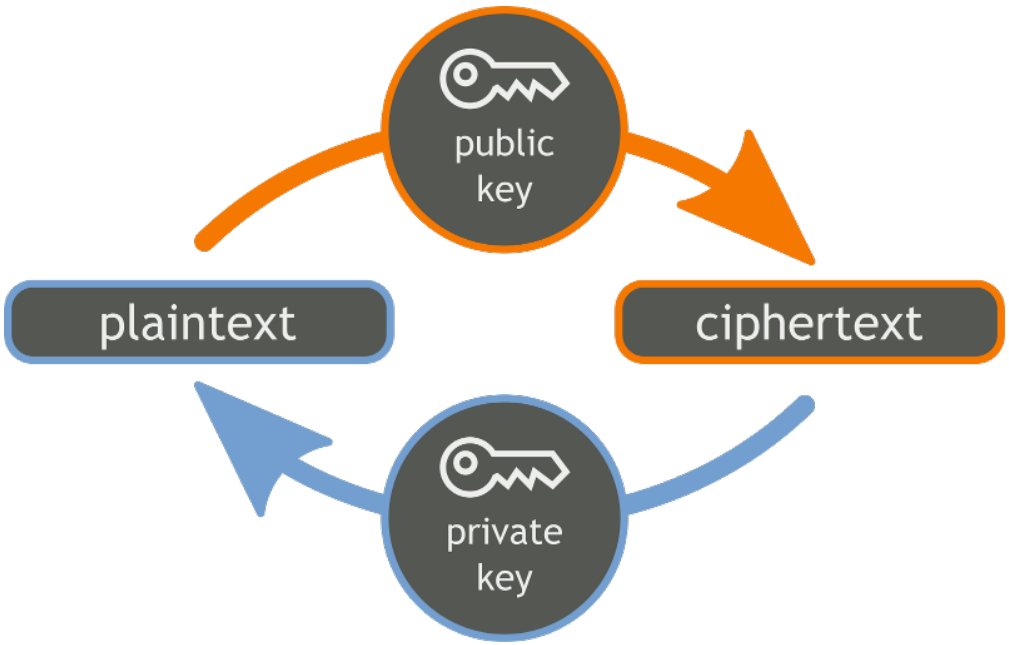
A bit of history of computer security...

70s : the era of the mainframe

70s : the era of the mainframe



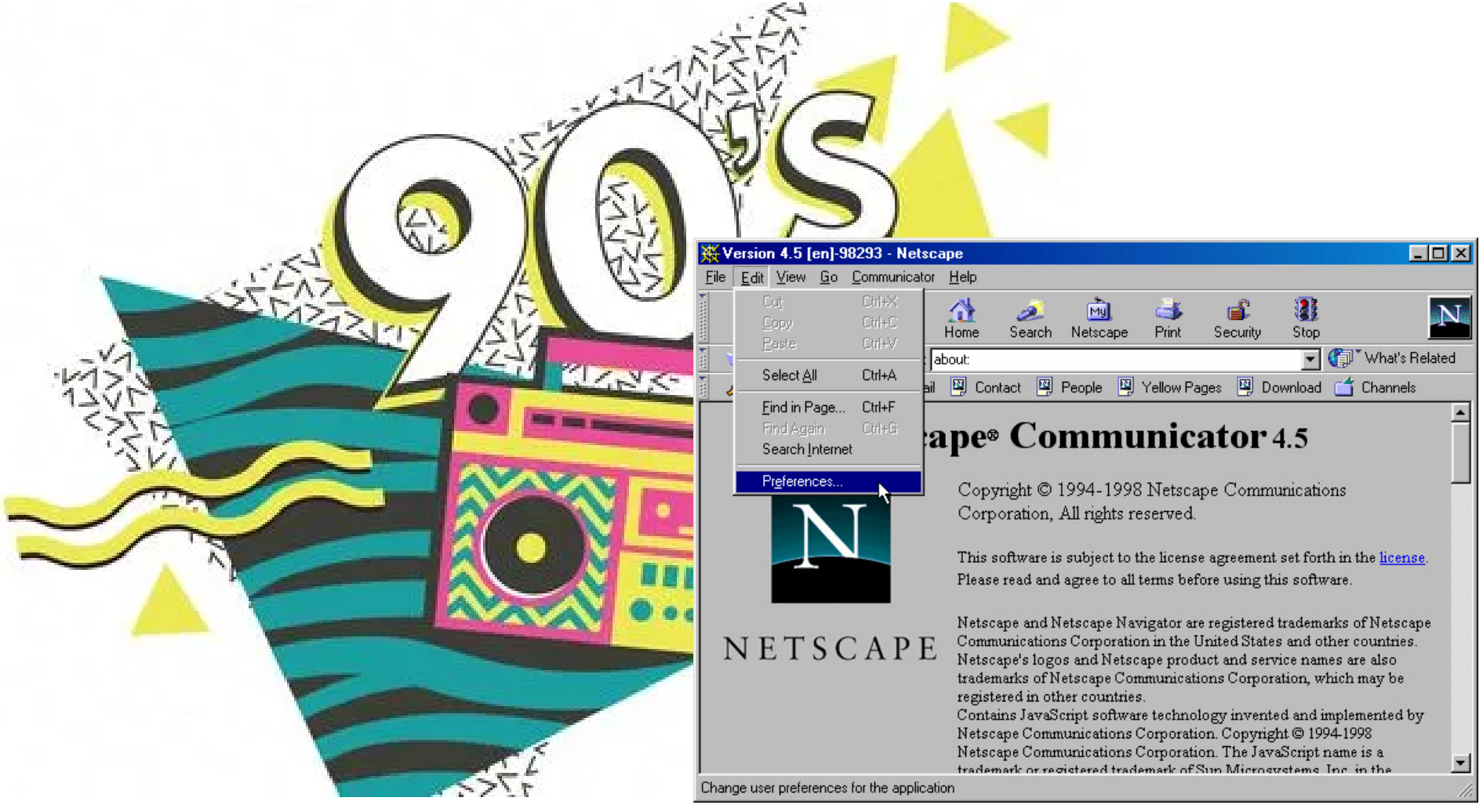




80s : the era of the PC

Morris worm, 1988

90s : the era of the Internet



90s : the era of the Internet

.oO Phrack 49 Oo.

Volume Seven, Issue Forty-Nine

File 14 of 16

BugTraq, r00t, and Underground.Org
bring you

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Smashing The Stack For Fun And Profit
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

by Aleph One
aleph1@underground.org

`smash the stack` [C programming] n. On many C implementations it is possible to corrupt the execution stack by writing past the end of an array declared auto in a routine. Code that does this is said to smash the stack, and can cause return from the routine to jump to a random address. This can produce some of the most insidious data-dependent bugs known to mankind. Variants include trash the stack, scribble the stack, mangle the stack; the term mung the stack is not used, as this is never done intentionally. See spam; see also alias bug, fandango on core, memory leak, precedence lossage, overrun screw.

00s : the era of the Web

Samy worm, 2005

Privacy


amazon.fr
ebay
facebook
Whom we know
TARGET
What we buy
IMDb
Google
twitter
What we like
NETFLIX
Where we go
RATP

Snowden leaks, 2013

TOP SECRET//SI//ORCON//NOFORN

(TS//SI//NF) PRISM Collection Details

Current Providers

- Microsoft (Hotmail, etc.)
- Google
- Yahoo!
- Facebook
- PalTalk
- YouTube
- Skype
- AOL
- Apple

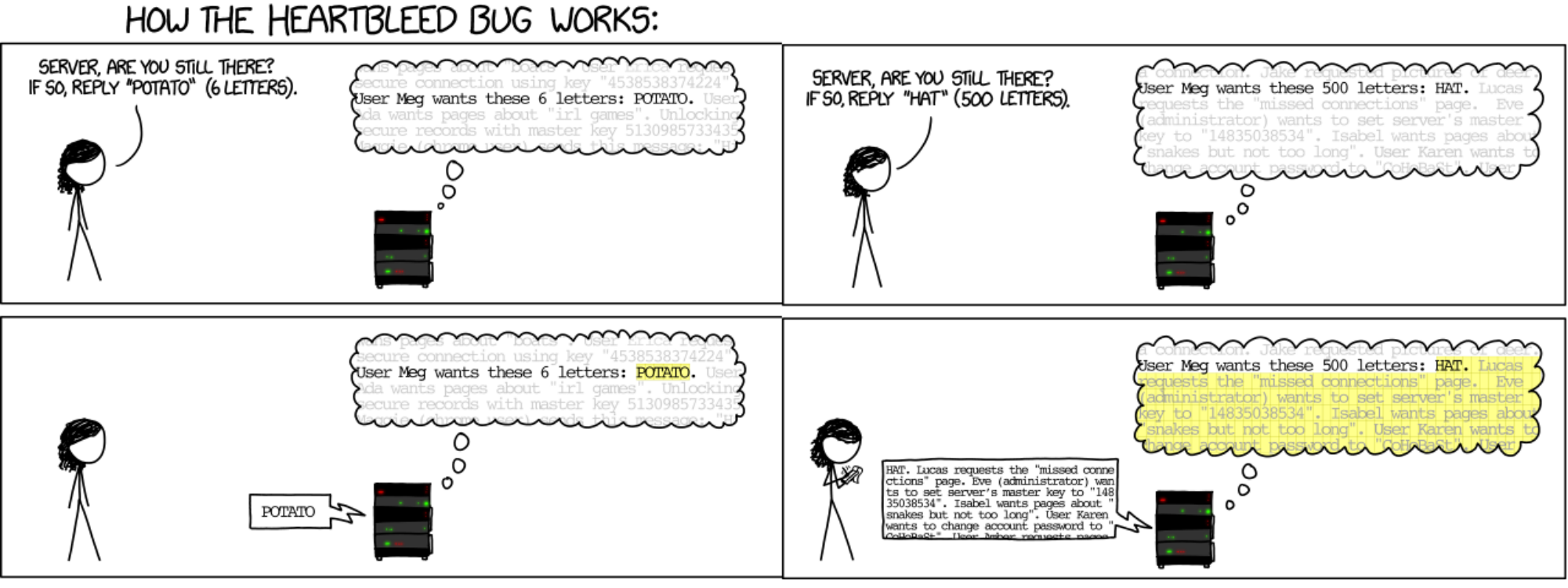
What Will You Receive in Collection (Surveillance and Stored Comms)?
It varies by provider. In general:

- E-mail
- Chat – video, voice
- Videos
- Photos
- Stored data
- VoIP
- File transfers
- Video Conferencing
- Notifications of target activity – login
- Online Social Networking details
- **Special Requests**

Complete list and details on PRISM web page:
Go PRISMFAA

TOP SECRET//SI//(

Heartbleed, 2014



Cambdridge Analytica scandal, 2018

Spectre / meltdown, 2018

Security Engineering

We want to build systems satisfying

- Confidentiality
- Integrity
- Availability

We want to build systems satisfying

- Confidentiality
- Integrity
- Availability

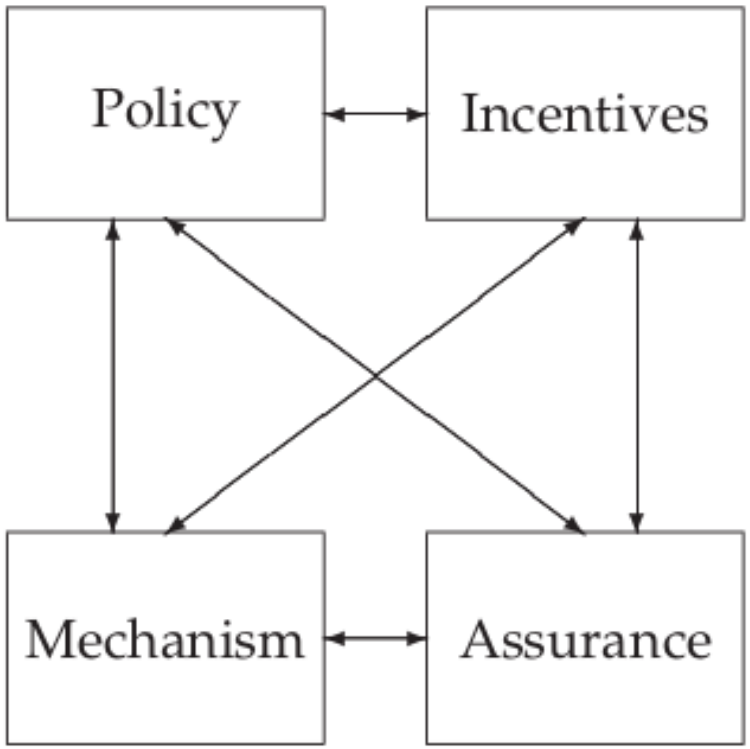
  but also...

- Authenticity
- Accountability / non-repudation
- Anonymity
- Privacy
- …

How?

- Prevention
  - eg. encrypt, validate inputs, …
- Detection
  - eg. check logs, monitor network activity, …
- Reaction
  - eg. update firewall rules

Security engineering



Σκοπός του μαθήματος

- να μελετήσουμε πως μπορούμε να αναπτύσσουμε ασφαλή συστήματα και εφαρμογές
- να μάθουμε συνηθισμένες αδυναμίες και επιθέσεις
- να αναλύσουμε διάφορες μεθόδους ανίχνευσης ευπαθειών και μηχανισμούς προστασίας
- να δούμε μερικά βασικά κρυπτογραφικά εργαλεία για να πραγματοποιούν ασφαλείς συναλλαγές.

Συμπεριφορά

- Το ότι κάποιος άφησε ανοικτή την πόρτα του ανοιχτή δεν σημαίνει ότι έχουμε το δικαίωμα να μπούμε μέσα
- Οποιοσδήποτε εφαρμόσει τεχνικές που παρουσιάστηκαν στο μάθημα (ή και εκτός αυτού) για την πραγματοποίηση επιθέσεων μηδενίζεται αυτομάτως (το οποίο πιθανότατα να είναι και ασήμαντο πρόβλημα σε σχέση με άλλες νομικές συνέπειες μιας τέτοιας πράξης)

References

- Ross Anderson, Security Engineering, Chapters 1-2
- `https://bitcoin.org/en/alert/2013-08-11-android`
- Wired: How Apple and Amazon Security Flaws Led to My Epic Hacking
- `http://en.wikipedia.org/wiki/Sarah_Palin_email_hack`
- `https://medium.com/p/24eb09e026dd`
- `https://github.com/hackappcom/ibrute`
- Trustwave issued a man-in-the-middle certificate

References

- https://limn.it/articles/the-morris-worm/
- https://samy.pl/myspace/
- http://heartbleed.com/
- https://meltdownattack.com/
- The Guardian: Cambridge analytica files