CS7NS5/CSU44032

Security & privacy

Stephen Farrell

stephen.farrell@cs.tcd.ie

Course materials:

https://down.dsg.cs.tcd.ie/cs7053/

https://github.com/sftcd/cs7053

Slideware + some papers

Computer and Network Security is…

- …a good thing to study (“one born every minute”, and some of those are programmers!)
- …something with more and more impact (scaling factor is about the same as the Internet)
- …a part of risk management

Privacy is…

- …nowhere near as well understood
- …an issue for people and not companies
- …not clearly a part of risk management, but related
  - I’m unsure if risk analysis is a good approach to address privacy

So-called “consent”

- Web sites/services/licensed things (like s/w) impose terms and conditions and require you “consent” to those (maybe via a “click-through”)
  - Legal fiction, everyone knows people do not read T&C documents designed to obfuscate
  - Apparently, that legal fiction is still considered non-fiction by courts
- So-called “consent” is an awful model that only pretends to address privacy
- Good presentation on the so-called “consent” problem, with IMO less good (but worth exploring) ideas on HOWTO fix https://datatracker.ietf.org/meeting/105/materials/slides-105-ietf-sesse-privacy-modern-concerns-steven-m-bellovin-00.pdf

Privacy

- RFC 6973 - In addition to “normal” security threats we need to care about
  - Correllation
  - Identification
  - Secondary use
  - Disclosure
  - Exclusion
  - **Re-identification**

Sidebar: Internet RFCs

- Last slide referred to RFC 6973
  - https://datatracker.ietf.org/doc/html/rfc6973
- The Request For Comment (RFC) series has been an important way to document Internet protocols and practices since 1969 when RFC1 was written
- Most recent one I co-authored: RFC 9446, published July 2023
- RFCs are freely available at many places but you can always get ‘em via the IETF (the main body producing RFCs), e.g. at
  - https://datatracker.ietf.org/doc/rfc9446/
- Not all RFCs specify protocols, some are process things, some are nerdy jokes (April 1st RFCs), and there’re other types too
- There will be some RFCs that you will have to read to pass the exam
- Anyway, I’ll usually not bother with URLs for RFCs as it should be obvious how to get ‘em

Privacy Puzzle

- Emails can contain a Received header field which can contain the mail user agent IP address
  - What consequences?
  - Overall good or bad from a privacy perspective?
- Redacted example:

```
Received: from localhost (ip-xxx-xxx-xxx-
  xxx.bb.vodafone.cz. [xxx.xxx.xxx.xxx])
 by smtp.gmail.com with ESMTPSA id
 h16-yyyyyyyyyy.40.2023.01.30.05.40.25
 (version=TLS1_3 cipher=TLS_AES_256_GCM_SHA384
  bits=256/256);
 Mon, 30 Jan 2023 05:40:25 -0800 (PST)
```

Privacy Processes

- Various have been proposed or are in use, but this is not a mature space (IMO)
- Privacy by design
  - https://en.wikipedia.org/wiki/Privacy_by_design
- Data protection by design and default
  - https://gdpr-info.eu/art-25-gdpr/
- Data protection impact assessments
  - https://dataprotection.ie/en/organisations/know-your-obligations/data-protection-impact-assessments

Irish Covidtracker example

- In 2020 the HSE (with developer Nearform) produced a GAEN covid tracking application
  - https://www.hse.ie/eng/services/news/newsfeatures/covid19-updates/covid-tracker-app/
- Way more information than you need on the Google/Apple Exposure Notification (GAEN) system:
  - https://down.dsg.cs.tcd.ie/tact/
- Original DPIA:
  - https://github.com/HSEIreland/covidtracker-documentation/blob/master/documentation/privacy/Data%20Protection%20Impact%20Assessment%20for%20the%20COVID%20Tracker%20App%20-%2026.06.2020.pdf
- While (I believe) they wanted to do all the right things, in a space that was very "foreign" for the HSE, the end result wasn't wonderful from a privacy perspective:
  - Dependency on Google Play Services (not HSE's fault, inherent in GAEN)
  - Super-cookie: JWT token required for TEK download – most other GAEN apps have no such token
- Note: the above is only about privacy, the efficacy of these apps is another question entirely (spoiler: I don't think they were useful), but that wasn't known (for sure) at that time

Privacy by design (PbD) Principles

7 "foundational principles":

- Proactive not reactive; preventive not remedial
- Privacy as the default setting
- Privacy embedded into design
- Full functionality – positive-sum, not zero-sum
- End-to-end security – full lifecycle protection
- Visibility and transparency – keep it open
- Respect for user privacy – keep it user-centric

- Explained further at: https://privacy.ucsc.edu/resources/privacy-by-design---foundational-principles.pdf
- Not clear (to me) that incentives align here sufficiently that result will be a privacy “win” – just stating an imperative doesn’t make things actually happen

A few more thoughts on privacy

- Who cares? About what?
  - Many governments, marketeers and large corporates do “care deeply” about your (lack of) privacy
- How can designers/implementers improve/protect privacy?
  - Behave as if your entire family will all be users – minimise!
  - Encrypt things in transit and storage
  - Short-lived dynamic identifiers are better than long-lived static identifiers
  - Just don't require identification
  - Your idea here...
- How can you help yourself as a user?
  - Don’t create more accounts
  - Target diversity
  - Your idea here...
  - Also see https://down.dsg.cs.tcd.ie/witidtm/

Risk Management

- Risks (bad things)
  - Disclosure of trade secrets
  - Sabotage (information or hardware)
  - Denial of service
  - Accidents (fire, flooding, earth quakes, …)
- Solutions (not always good things)
  - Security policies and mechanisms
  - Physical security (locks, guards, CCTV, …)
  - Formal specification/verification of software
  - Halon, UPS, off-site backups

Vulnerabilities

- Many risks arise due to the existence of vulnerabilities in computer systems
- All systems have vulnerabilities, our goal is not to remove absolutely all of them, but to control their impact
  - Reducing numbers is good
  - Can also isolate parts of the system (e.g. Firewalling)

Vulnerabilities

- Very common:
  - Scripting user agents
  - Buffer overruns
  - XSS & Injection (e.g. SQL injection)
    - https://owasp.org/www-community/attacks/xss/
  - Insecure default settings
- Uncommon, but interesting:
  - Acoustic side-channel key extraction,
    - Genkin, Shamir & Tromer
  - https://eprint.iacr.org/2013/857.pdf

Figure 6: Parabolic microphone (same as in Figure 5), attached to the portable measurement setup (in a padded briefcase), attacking a target laptop from a distance of 4 meters. Full key extraction is possible in this configuration and distance (see Section 5.4).

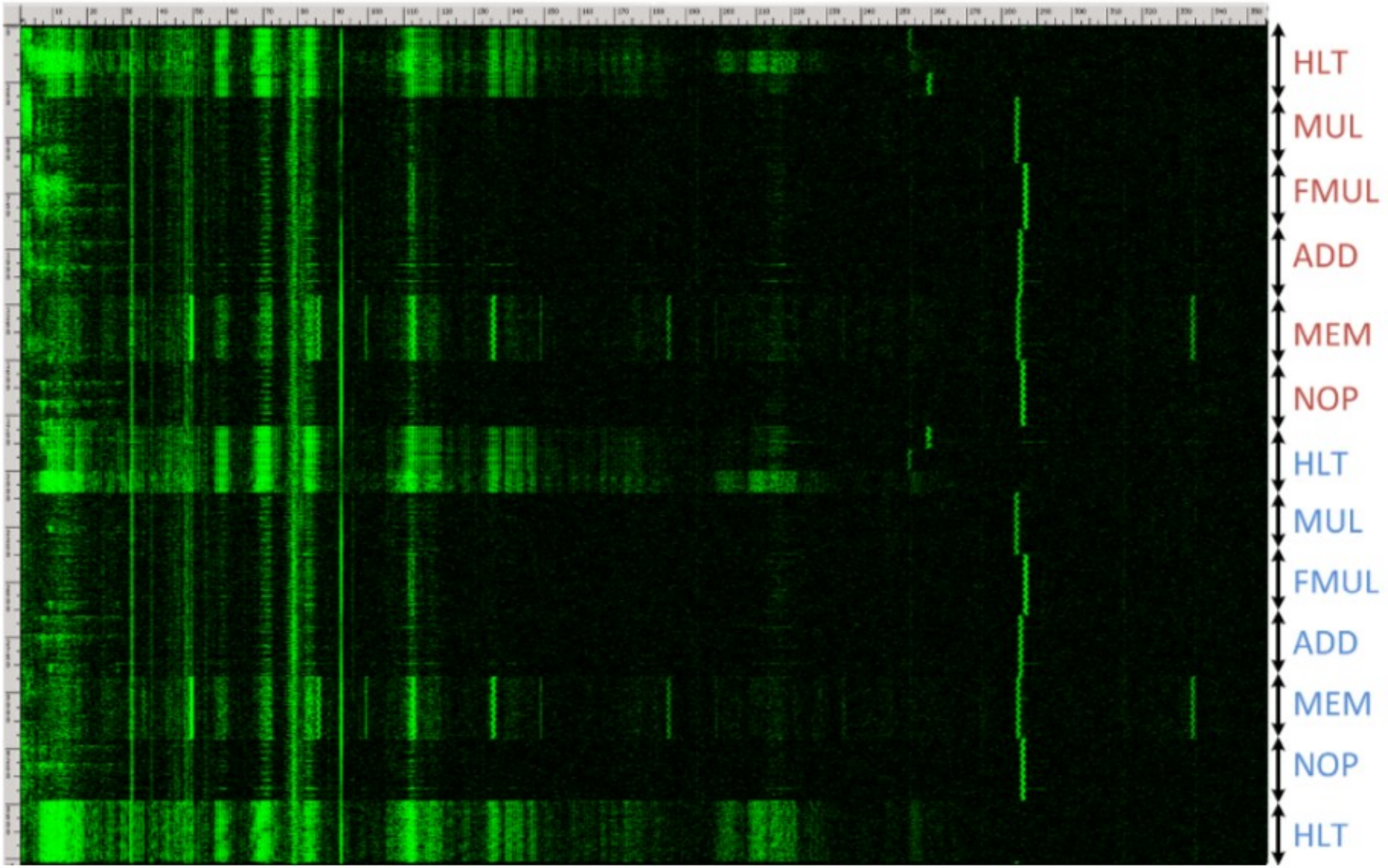

Figure 7: Acoustic measurement frequency spectrogram of a recording of different CPU operations using the Brüel&Kjær 4939 microphone capsule. The horizontal axis is frequency (0–310 kHz), the vertical axis is time (3.7 sec), and intensity is proportional to the instantaneous energy in that frequency band.

Good/Bad Actors

- Systems have users
  - Normal, administrative, “root”
- Networks have nodes
  - “Inside”, “outside”, trusted…
- Attackers
  - Can be one of the above, **or not**…
  - Hijacked ISP router, compromised SIM card factory, bot etc.

Possible Bad Actors

- Disgruntled employees (*plenty*)
- Crackers (*hackers*)
- Script-Kiddies (*cracker wannabes*)
- Spies (*industrial and military*)
- Criminals (*thieves, organized crime*)
- Terrorists
- Governments
- Bait’n’switchers (service providers who change business model)

Possible Exploits

- Force legitimate user to reveal passwords
- Social engineering
- Recruit legitimate user
- Sabotage (*fire, electricity, …*)
- Sifting through garbage
- Attacking the network (*network threats*)
- Install malware

Active/Passive Attacks

- Active attacks
  - Fabrication, modification, deletion, replay of messages
- Passive attacks
  - Eavesdropping/traffic analysis
  - Can be off-line (e.g. if weak encryption)
- Different protocol mechanisms are used to counter these

Risk Analysis Process

Many variations exist, mostly they resemble:

- Identify assets
- Identify risks and vulnerabilities
- Consider probabilities
- Consider consequent costs/losses
- Rank risks
- Develop mitigation(s) for highest ranked risk(s)
- Iterate, until effort exhausted or time up
  - All the time recording what you've done

Summing up risk

- Risk is a function of the cost of threats and their probability of occurrence
  - Which function can be debated
  - High/Medium/Low
    - For both costs and probabilities
- Threats occur when a vulnerability is exploited

Exams eh?

The two previous slides are important for exam purposes – look back at old exams and you'll see why