Application Security (apsi)

Lecture at FHNW

Lecture 15

Arno Wagner, Michael Schläpfer, Rolf Wagner

<arno@wagner.name>, <{michael.schlaepfer,rolf-wagner}@fort-it.ch>

Agenda

- Data Protection (Datenschutz)
  - Motivation: Why is it important?
  - Risks: What can attackers do if Data Protection is inadequate?
  - How to do Data Protection right?
  - Some Specific Aspects of the GDPR (DSGVO)
- Future Trends (A Look into the Crystal Ball)

Some Definitions (mostly from Wikipedia...)

- Human Rights: The "basic rights and freedoms to which all humans are entitled", including civil and political rights, such as the right to life and liberty, freedom of expression, and equality before the law; and economic, social and cultural rights, including the right to participate in culture, the right to food, the right to work, and the right to education.

  Note: This is a newer thing. For most of history, nobody had any rights just because they were human.

  - Universal declaration of Human Rights 1948
  - Idea has been taken serious since around 1600 ("natural rights")
- Privacy (a Human Right):
  Privacy is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively.
- Fear: Fear is a basic emotional sensation and response system ("feeling") initiated by an aversion to some perceived risk or threat.

Some Definitions (2)

- Democracy: Democracy is a form of government in which power ultimately comes from the people who are governed, whether through direct voting or through elected representatives.
  Quote: Democracy is the worst form of government except all those other forms that have been tried. (Winston Churchill)
- Surveillance: Surveillance is the monitoring of the behavior and activities of people for the purpose of influencing, managing, directing, or protecting them.
- Police State: Police state is a term denoting government that exercises power arbitrarily through policing.
- Totalitarianism: A political system where the state recognizes no limits to its authority and strives to regulate every aspect of public and private life wherever feasible.
- Fascism: A from of totalitarianism. Fascists believe a nation is an organic community that requires strong leadership, singular collective identity, and the will and ability to commit violence and wage war in order to keep the nation strong.

Note: In any organized state, there are forces trying to drive things downwards on the above list. They have to be carefully monitored and kept under control.

Reasons for Privacy

- Human Rights: From the "UN Declaration of Human Rights":

  No one shall be subjected to arbitrary interference with his privacy, family, home or correspondence, nor to attacks upon his honor and reputation. Everyone had the right to the protection of the law against such interference or attacks.

This is a moral thing. It is "What kind of world do you want to live in?"

However, rational justifications do exist:

- Modern societies invest a lot in individuals education-wise, and therefore the well-being of the individual has become important for prosperity.
- Modern societies are critically dependent on their best and brightest and these can turn up in surprising places.
- Modern societies found that granting individuals a lot of freedoms leads to better stability with all its benefits. The more room for decisions an individual has and the less control and surveillance, the less incidence of stress, sickness, depression, etc. also at work. People must feel they are valuable as individuals.
  => The more repression and surveillance, the lower the economic power!

Reasons for Privacy (2)

Prevent "Chilling Effects":

- Definition (others exist): Chilling Effects are when individuals self-censor in their expressions, opinions and choices, even if they are perfectly legal, because they fear negative effects.

Examples:

- After the Snowden revelations, Wikipedia articles related to terrorism were visited 30% less often.
- People not joining public protests because they fear losing jobs.
- People not saying what they think in email, because "the state can read them". Also note the global efforts to outlaw secure encryption...

Effect:

- People are less informed and organized, understand less and have less influence

But: Democracy and freedom is defended by the citizens, not by the "authorities"!

"Authorities" have a long history of fighting privacy

- People expressing their opinions are inconvenient to those in power. Now if there only was a way to shut them up…
  - Example: An all-seeing, all knowing and vengeful "God", that punishes people thinking or doing the "wrong" things (defined by those in power…)
  - Example: The "Big Brother" in Orwell's "1984"
- The more control the state has, the easier it becomes to rule and expand control even further
  - Governments not carefully observed and limited in their power by the citizens have a tendency to devolve into police-states and then into totalitarian regimes.
- There is a type of person that cannot stand others thinking differently. These people often seek power to be able impose their views on others by force.
  - Literature: Bob Altemeyer ,"The Authoritarians", free eBook, https://theauthoritarians.org

The Role of Fear

"Fear is the mind-killer…" (Frank Herbert, "Dune").

- Fear has a strong negative impacts decision-making capabilities
- People in fear are not be able to rationally evaluate threats anymore
  => They become easy prey for those claiming to have the answers
  => These "answers" almost universally do not target the problem

  Example: Mass-surveillance is obviously ineffective against terrorism, with countless examples. Yet it is often advertised as the "solution".
- Widespread fear is an invitation for erosion of freedoms and of democracy
  => Fear is created, amplified, spread by those that want more power

This is a fundamental human problem and no good solution is known.
Those in power benefit from fear, so there is little motivation to look into solutions.
"Human rights" are the best we have come up with so far.

Democracy and Freedoms need to be fought for

- If not, they vanish (There are countless historic and current examples.)
- It often is a slippery slope
  - First unconditional data-retention
  - Then storage of all emails in plain and mandatory backdoors in cryptography
  - Then retention of full web-browsing history
  - Then a surveillance software right on your computer
    (China already has this in part...)
- And at some time it is too late to do anything, because anybody that tries can be jailed or killed "lawfully" for thought-crimes.

This is not a fight that can be "won":

"Eternal vigilance is the price of freedom" (multiple sources).

Why is this relevant to APSI?

- Today, the possibilities for electronic data collection and correlation are unprecedented in human history
- Most of it is done by IT experts in private or government employment
  - Many of you will be faced with relevant choices in your career
  - You should have some understanding of what is involved
    (The choices are yours. This lecture can only give some basis to make them on.)
- Quite a bit of data-collection looks harmless, but is not.
- Data can be retained for a long, long time and even impact descendants.
- Governments have already taken to electronic attacks to obtain data of individuals and buy data from private suppliers.
- The risk of Data Privacy violations can be reduced, because it also is a technical security problem.

What Can be Done With Data?

Data can be used to coerce and destroy people

- Data recorded earlier in a person's life can be used to control or eliminate them when they have or are about to obtain a position of power
  - Those that have the data (companies, secret agencies) can subvert Democracy and the rule of law. History shows they will use such possibilities.
- Most people do things in their life that will not look good to the public, often when they were young.
- Medical data is always critical.
- Data can be used to identify and define groups of "undesirable" persons.
  - These can then be conveniently blamed for all problems and can take the fall…
- Data can be used to disadvantage people in sharing the wealth of society
  - No or limited access to healthcare (because of known risk factors)
  - No or limited access to education, certain jobs, etc.

How Critical Can (Seemingly Harmless) Data Be?

Some examples:

- Assume the buyer records for food. These look pretty harmless…

  Now, assume some people form <insert country here> do something "bad".

  Just find anybody that bought food from that country or in the style of that country as a potential sympathizer! (Will only have low reliability, but together with other data-sources, that changes…)

- Assume the mouse-movements of a person playing a game. Looks pretty harmless, right?

  Now, you can determine intoxication-levels from them and levels of tremor. This can be used to increase cost of health-insurance, for example.

Privacy-Relevant Data

We have seen that personal data can be misused.

What data is relevant here?

*** All data that pertains to an individual or a group of people! ***

There really is no limit. As long as it refers to people, it is critical with regards to Data Privacy!

Note: Medical data is especially critical.

Question: Does Cumulus/Supercard data contain medical data?

General Principles of Data Privacy

- Always perform a proper risk assessment before deploying systems handling privacy-relevant data
- Ask for and store the least amount of data necessary
- Delete data as soon as it is not needed anymore
- Make sure that recorded data is accurate
- Make sure data is properly secured against unauthorized access
- Respect Human Rights and local laws
- Inform people when their data is being recorded or given away and ask for permission

Note: Data Privacy includes Data Security!

Note: There is a new field called "Privacy Engineering"

Privacy by Design (from Systems Engineering)

Privacy by Design is based on 7 "foundational principles"

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

These are vague and not very helpful, but things have to start somewhere...

Role of Encryption

Encryption only protects data that is not accessed

- In transfer
- On backups and long-term storage
- Not: Currently being processed or readily accessible

However, encryption can help with Data Privacy

- Encrypt data in transfer and long-term storage (also backups)
- Decrypt only for the data owner (dependent on password or only in an application on the data owner's side)
  => This means each data owner need to be attacked individually
- Use cryptographic measures to secure log-in credentials

Data Anonymization

Why do it? Statistics!

This is a lot more difficult than it looks...

- In many cases, de-anonymization is very easy
- The more data an adversary already has, the simpler de-anonymization becomes
- 100% accuracy is often not required (hang the innocent with the guilty…)
  - Generating suspicion is often enough
  - If a statistical measure is the aim, errors can be tolerated without problem
- Simplistic approaches do not work
  Examples:
  - Delete first digits of a number, for example the phone number
  - Only keep decade of day-of-birth
  - Only keep last name

Data Anonymization (2)

So, what can be done?

- Anonymize so that individuals can only be identified within a large group
- Anonymize, derive a secondary result and then delete the anonymized data
- Get consent!
- Work with a relatively small random sample and use different samples for different aspects to be correlated (may give bad results)
- Use actual data to train a classifier (neural network, for example) and then work with that
- Create synthetic data that matches the characteristics of the original data (this is a current research topic…)

In all cases, try really hard to de-anonymize and have some other competent people do the same. This gives an indicator of quality.

What about the Cloud?

Simple: Do you trust the cloud-operator? No? Question answered.

- Do not store non-encrypted privacy-relevant data in the cloud
- Do not process privacy-relevant data in the cloud

What would be needed to create this trust?

- The cloud provider must be forbidden to share data except by an individual court order (no mass-orders) and before that happens, the affected individuals must be informed and be allowed to fight that order.
- Collaboration with secret agencies must be absolutely forbidden.
- The cloud provider must be absolutely forbidden to keep unauthorized copies or backups.

If these conditions are violated, both the cloud provider and the violating agency must be held criminally liable (individuals get punished) and the data gained must not be used in any way and must be deleted.

The GDPR (DSGVO): Some Principles

This is very incomplete!

- Continued storage and processing of personal data only with a valid reason
- No storage of personal data without consent
  → This includes personalized cookies and other tracking-data
- Time-dependent data retention and data deletion obligations
- On request, people must be told what data is stored about them
- People can request deletion of data under some circumstances

And most important: There are real penalties possible on violation!

A Tricky Problem: Long-Term Backups and Archives

This is a real-world example…

Situation: A company offers services for money. The relevant personal customer data needs to be kept 10 years and deleted after that.

Assume:

1) The life-system is backed up to tape every month
2) Every year, the tapes from January and July are turned into archives
3) Archive tapes are deleted after 10 years
4) The data is deleted from the life-system 10 years after the last transaction

Question: How long after the last transaction is customer data deleted?

Long-Term Backups and Archives (2)

So, how long is the data being kept after the last transaction?

20 Years!

→ Deleted from the life system after 10 years and the last archive tape from before that deletion is kept another 10 years.

That is a violation of the rules…

So, why is the data kept on the life system at all?

- New transactions may happen
- If somebody asks, you need to be able to find all their data
  → Difficult to do on the archive
- If somebody requests it, you may have to erase all data
  → You need to find it first for that!

Right-to-be Forgotten vs. Archives

In theory, you need to be able to erase each individual from the archive

- How do you erase just one person's data from a tape?
  - What do you do if the original software is not available/compatible anymore?
  - What do you do with binary or in-transparent formats?

- How do you even know what is on an archive tape?
  → Data may have been erased from the life-system for some reason

If you lie about this, the fines can be huge!

Right-to-be Forgotten vs. Archives

So assume you tell people that you cannot really delete from the archive, but that all their data is gone from the life-system. (Apparently you may get away with this for a while yet.)

This allows a nice attack:

- A person requests data deletion → Nothing left on life-system
- A year later the same person requests information about all the data stored about them
- Because there is no data in the life-system, you tell them "none"
- They then file a legal complaint, because they know they are still on the archive tapes...

Possible solutions for GDPR-Conform Archives

This is speculative!

1) Separate all archived data by person it relates to
   - The applications need to provide that type of data backup capability!
   - The "section" relevant to an individual needs to be easily identifiable and erasable
     → Special tools must not be required (open plain-text formats)
   - The archive needs to be relatively easy to manipulate
   - But: It may need to be revision-proof (not changeable) by regulatory requirement
   - → If so, this does not work.
     Otherwise use an online archive that can be written under special circumstances
2) Same as above, but encrypt every section with a person-specific key.
   → To erase data for one person, just erase that key (kept separately)
3) Some sort of trusted 3rd party ("Treuhaender") keeping the archive and only allowing access under specific circumstances.
4) ???

Backups are Easier

But not trivial...

Restoring backups needs to be adjusted:

- Keep a list of all persons that had their data erased during backup lifetime
  Note: You do not need to erase backups for "right-to-be forgotten" immediately. It can be done within a reasonable time-frame (e.g. 3 months) and the person requesting deletion can be informed accordingly.
- BUT: After a restore (DR case), you must make sure they get erased again!

Additional Reading

- Read the Wikipedia-Article on China’s social scoring program:
  https://digitalcourage.de/blog/2018/social-scoring-china-das-sozial-kredit-system
- Watch David Kriesel's talk on “Spiegelmining” at the 2016 Chaos Communication Congress:
  https://www.youtube.com/watch?time_continue=5&v=-YpwsdRKt8Q&feature=emb_logo

Future Trends

- Sandboxes
  - Example: Web-Assembler
- Programming Languages
  - Example: Rust
- Artificial Intelligence
  - What can we expect in the next decade(s)?

Sandboxes

- Encapsulate and isolate
- Examples:
  - Fully virtualized environments (e.g. Virtual Box, VmWare, Xen, …)
    Problems: heavy-weight, need a full OS installation
  - General virtual machines for a language (e.g. JavaVM)
  - Problems: not general, often be slow and/or memory intensive
  - Browser-encapsulated languages (e.g. JavaScript)
  - Similar problems as general language VMs
  - Virtual "sandboxes" done by compilation process

A new Approach: WebAssembly ("Wasm")

Sandboxed assembly execution of applications in web-browsers

- Stack-machine-based assembler variant (very portable)
  - Captures what a typical modern 32 or 64 bit CPU can do
  - Hioghe performance, regardless of language
  - High-level languages get compiled to it
  - Missing features of a CPU can be added by JIT or exceptions
- Binary format
- System API, libraries: Like JavaScript ("web-environment" model)
- Callability JavaScript ↔ WebAssembly
- Non-web: ~~Planned~~ Beginning to become available (e.g. "Wasmer")

References: http://webassembly.org/
https://en.wikipedia.org/wiki/WebAssembly

Modern System Programming Languages

Aim to be as fast as C and have some of the advantages of scripting

- Imperative, OO and functional features
- Static type-safety as much as possible
- Memory-safety as much as possible
- Avoidance of races (the bane of all concurrent system programming and source of many security problems…) as much as possible
- Syntax and compiler that make common mistakes less likely

Examples: Rust, Go

These are currently in an experimental phase
→ Nobody knows how well they really work in the long run
→ Nobody knows whether they will have long-term support

Some excessively optimistic promises are made (again…)
→ These almost universally do not pan out

Some Features of Rust

- C-like performance for most things
- Guaranteed memory safety:
  - There is no way to overflow buffers, write to arbitrary memory, etc.
- Threads without data races
  (General race-conditions cannot really be avoided...)
- Move semantics:
  - Automatic invalidation of source in assignments with "let"
    Example: "let a = b; let c = b;" → 2nd assignment fails.
    Note: can be done manually, e.g. in C as "a = b; b = NULL;"

But: For some things you need to go "unsafe" or use C code
→ You lose all those nice assurances…and you do not have experience with that!

Note: The Rust "culture" is infected by the idea of "safe-spaces"...

What About AI (Here: AGI) ?

We hear things like

- "In 10 years, 80% of all code will be written by AI"

Is this realistic?

No, it is not. This has failed before…

- 5GL languages are "*programming languages based on solving constraints given to the program, rather than using an algorithm written by a programmer*" (i.e. you input the spec, code falls out "magically", around 1980)
  → Complete failure, never really produced anything
- An AI that can code would need to understand coding, algorithms and the problem being solved
  → This is currently not even theoretically possible.
  Machines understand absolutely nothing at this time.
  For very limited problem spaces, they sometimes can fake understanding.
  → That means it is at least 50 years in the future and may be impossible...

What are the Problems with AI?

- Nobody knows what "intelligence" is and how it works
  - Does intelligence require "self-awareness"?
    → Intelligence is only observable in connection with it. That is a rather strong hint.
    → Nobody knows what "self-awareness" is...
  - Some people (mostly "physicalists") think self-awareness is an illusion.
    → If so, who or what has that illusion? This argument is circular.
  - Does intelligence require "free will"?

    → Intelligence is currently only observable in connection with it...
  - Some people think that "free will" is an illusion…
- There are no known ways to implement AI that would work in this universe.
  - The only credible thing (automated theorem proving) does not scale to what a smart human being can do, not enough matter, energy and time available
    → Hugely useful for assisted proof verification though!
- It is completely unknown whether intelligent machines can be created
- It is completely unknown what intelligent machines would look like

The Future of AI

My prediction:

- We will not see it anytime soon
- We may never see it

But what about IBM Watson, self-driving cars, …?

- Watson is a NLP-driven expert system
  - Not AI (IBM incidentally does not claim so to expert audiences…)
  - Data put in needs no pre-conditioning, natural language documents work
  - → Huge advantage when using scientific literature, crawling the web, spying on emails, chats and phone-calls, etc. (surveillance is probably main use...)
    → It makes some things a lot cheaper!
- Self-driving: "Automation", not AI. The problem is not that complicated.
  Example: No understanding of abstract concepts needed.
  (Otherwise many people could not learn to drive…)

References

- Bruce Schneier thinks differently about AI and code vulnerabilities:
  "The problem of finding software vulnerabilities seems well-suited for ML systems." (ML = Machine Learning, a sub-area of AI)
  I doubt that, many other people do as well. Also refer to the comments.
  For example user "Impossible Stupid" has a good comment.

  Link:
  https://www.schneier.com/blog/archives/2019/01/machine_learnin.html

That is it with APSI for this time….

Thank you for the attention!

I hope APSI provided you with some new insights

Questions regarding the exam: After this or by email