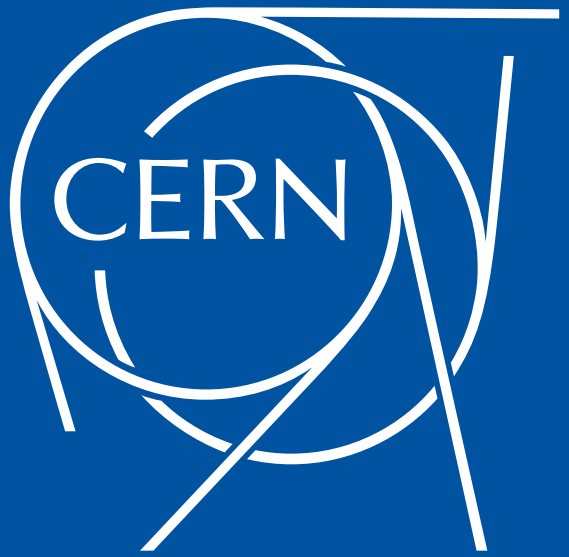

CERN

Introduction to Information Security

CODATA School

Hannah Short (CERN), Sebastian Lopienski (CERN)

Lecturers

These slides have been compiled by members of the CERN Computer Security Team based at CERN, the European Organisation for Nuclear Research.

Hannah Short

Sebastian Lopienski

Why Security?

Data Security Concepts

Security Objectives

Guidelines and Principles

Software Vulnerabilities

Data Privacy

Course Objectives

- Understand why Security is important for you as a Data Scientist
- Familiarise yourself with the basic principles of Information Security

Note:
If the slide title is in red, the slide is considered an advanced topic

Security Measures

Is this a good security measure?

Why Security?

Why Security?

- You are constantly exposed to reputational, financial and even physical risks online
- The aim is to **minimise your exposure to risk** through
  - Secure online activity
  - Secure software design

Safety vs Security

Safety is about protecting from **accidental risks**

- road safety
- air travel safety

Security is about mitigating risks of dangers caused by **intentional, malicious actions**

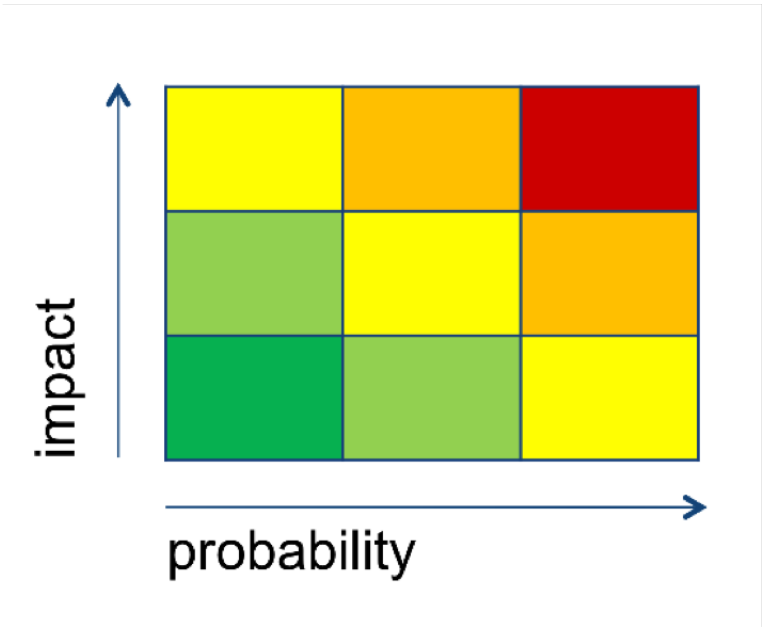
- homeland security
- airport and aircraft security
- information and computer security

Why is security difficult?

Security is as strong as the weakest link. There is no 100% security!

What is risk?

- Probability * impact
- Risks should be: Assessed, Prioritised, Mitigated, Avoided and finally Accepted



Typical Threats

But we're Scientists, surely we're not a target...!

Typical Threats

BBC Sign in | News | Sport | Weather | Capital | TV | Radio

NEWS Watch ONE-MINUTE WORLD NEWS

News Front Page
Africa
Americas
Asia-Pacific
Europe
Middle East
South Asia
UK
Business
Health
Science & Environment
Technology
Entertainment
Also in the news

Video and Audio

Page last updated at 11:24 GMT, Monday, 15 September 2008 12:24 UK

E-mail this to a friend Printable version

'Big bang' experiment is hacked

Part of the computer system of the Large Hadron Collider (LHC) was hacked into as the world's most powerful physics experiment got under way.

The CMS detector was not affected by the computer hackers

A group calling itself the "Greek Security Team" hacked into a computer connected to the system last Wednesday.

A spokesman for Cern, the lab that houses the LHC, said the hackers put up a message on the facility's website.

No harm was done but the incident has highlighted the need for security in the LHC's network, the spokesman said.

http://news.bbc.co.uk/2/hi/technology/7616622.stm

Typical Threats

https://www.wired.com/2008/09/hackers-infiltr/

Attackers

criminals

motivation:
profit

hacktivists

motivation:
ideology,
revenge

governments

motivation:
control,
politics

Hacking as a Business

1. Send this:

```
POST live_events_edit_status_ajax?action_delete_event=1
Host: www.youtube.com

event_id: ANY_VIDEO_ID
session_token: YOUR_TOKEN
```

2. Receive this:

```
{
   "success": 1
}
```

3. Report to Google and get $5'000 bounty
   http://kamil.hism.ru/posts/about-vrg-and-delete-any-youtube-video-issue.html

Hacking as a Business

1. Send this:

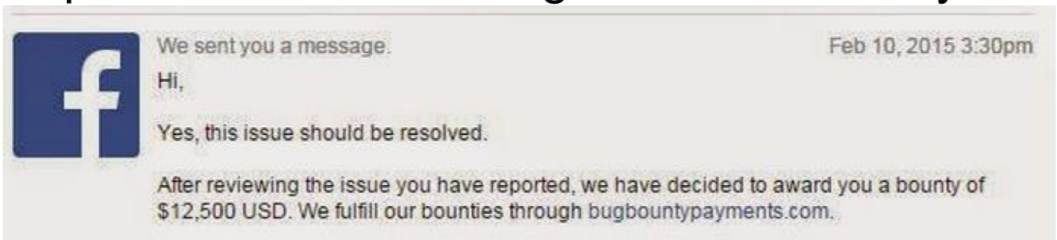
```
DELETE /ANY_PHOTO_ALBUM_ID HTTP/1.1
Host: graph.facebook.com

access_token=YOUR_FACEBOOK_FOR_ANDROID_ACCESS_TOKEN
```

2. Receive this:

```
true
```

3. Report to Facebook and get $12'500 bounty



http://www.7xter.com/2015/02/how-i-hacked-your-facebook-photos.html

3

Why Security - Summary

- Security = mitigating risk of malicious actions
- Science is an interesting target for bad guys/girls

Data Security Concepts

Data Security Concepts

At the heart of Security we have three key components:

- Technology
- Processes
- People

Technology

We will come back to some of this in part 2 of our lecture course :)

Processes

"Security is a process, not a product" - Bruce Schneier

Processes

Processes

Security solutions often degrade with time - they need to be verified periodically!

People

- Have flawed risk perception
- Are bad at dealing with exceptions and rare cases
- Put too much trust in their computers
- Easily fall for social engineering
- Sometimes turn malicious
- Prefer convenience and bypass security measures
- Often make mistakes...

Risk Perception

Is flying more dangerous than traveling by car?

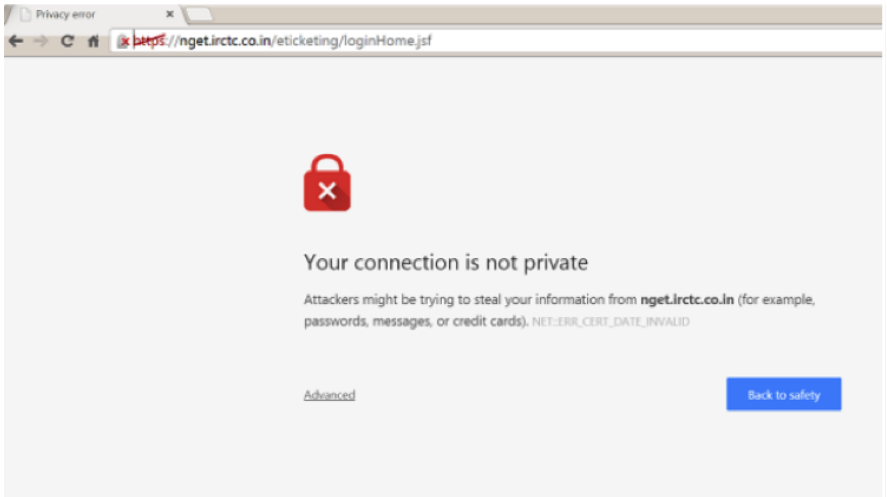
Are you more likely to be killed by a shark, a pig or a coconut?

Social Engineering

https://www.smbc-comics.com

Taking security decisions

Users typically make poor security choices despite systems trying to protect them!



And sometimes it's just plain difficult

Which links point to eBay?

- secure-ebay.com
- www.ebay.com\cgi-bin\login?ds=1%204324@%31%32%34.%31%33%36%2e%31%30%2e%32%30%33/p?uh3f223d
- www.ebay.com/ws/eBayISAPI.dll?SignIn
- scgi.ebay.com/ws/eBayISAPI.dll?RegisterEnterInfo&siteid=0&co_partnerid=2&usage=0&ru=http%3A%2F%2Fwww.ebay.com&rafId=0&encRafId=default

...

Data Security Concepts - Summary

- Processes must be ongoing, security degrades with time
- People often provide the easiest way for an attacker to compromise the system
- Security is only as strong as the weakest link - don't lock the front door but leave the back door open!

Security Objectives

Security Objectives

Computer Security aims to meet these objectives:

- Confidentiality
- Integrity
- Availability

We will start with a quick look at Identity, as this is essential for meeting security objectives!

Identity

Online Identity is really no different from your real life Identity! Your Identity is the answer to the question: **“who are you?”**

- It could be a username for a website
- It could be a government ID
- It could be a digital certificate

Authentication and Authorisation

Authentication = How can I prove my Identity?

Authorisation = What am I able to do?

Multifactor Authentication

Factor	Description	Example
1	Something you know	Password, pin
2	Something you have	Phone, Yubikey
3	Something you are	Fingerprint, iris scan

Which is most secure?

Security Objectives

- **Confidentiality**
- Integrity
- Availability

Can the correct people access the data at the correct time?

Security Tip: Pay attention to where your data is stored and how it is shared!

Confidentiality

- Your online identity is as valuable as your passport
- Your authorisation may be misused if it falls into the wrong hands

Security Tip: Store your secrets safely, not in the public domain, e.g. github

The Register®
Biting the hand that feeds IT

Security

Dev put AWS keys on Github. Then BAD THINGS happened

Fertile fields for Bitcoin yields - with a nasty financial sting

By Darren Pauli 6 Jan 2015 at 13:02 25 SHARE ▼

Bots are crawling all over GitHub seeking secret keys, a developer served with a $2,375 Bitcoin mining bill found.

DevFactor founder Andrew Hoffman said he used Figaro to secure Rails apps which published his Amazon S3 keys to his GitHub account.

He noticed the blunder and pulled the keys within five minutes, but that was enough for a bot to pounce and spin up instances for Bitcoin mining.

"When I woke up the next morning, I had four emails and a missed phone call from Amazon AWS - something about 140 servers running on my AWS account," Hoffman said.

"I only had S3 keys on my GitHub and they were gone within five minutes!"

"As it turns out, through the S3 API you can actually spin up EC2 instances, and my key had been spotted by a bot that continually searches GitHub for API keys."

Most read


Fork it! Google fined €4.34bn over Android, has 90 days to behave


Official: The shape of the smartphone is changing forever


Trump wants to work with Russia on infosec. Security experts: lol no


Boss helped sysadmin take down horrible client with swift kick to the nether regions


British Airways' latest Total Inability To Support Upwardness of Planes* caused by Amadeus system outage

How bad can it be?

- 5 minutes exposure
- $2,375
- Plus it could have been avoided, Amazon has a service (IAM) to manage keys securely...

`https://www.theregister.co.uk/2015/01/06/dev_blunder_shows_github_crawling_with_keyslurping_bots/`

Security Objectives

- Confidentiality
- **Integrity**
- Availability

Can we be sure that the data is reliable and hasn't been altered?

Security Tip: Reduce the risk of impersonation, enable multi-factor authentication wherever possible!

Security Objectives

- Confidentiality
- Integrity
- **Availability**

Is the data available? Are our systems reliable?

Security Tip: Keep backups!

Security Objectives - Summary

- Key objectives: Confidentiality, Integrity and Availability
- Consider disaster scenarios and plan for them
- Online Identity is critical to meeting security objectives

Guidelines and Principles

Security Measures

Is this a good security measure?

Security Measures

- What problem is it trying to solve?
- Does it help?
- Does it introduce new problems?
- What are the costs?

Security Measures

How much security?

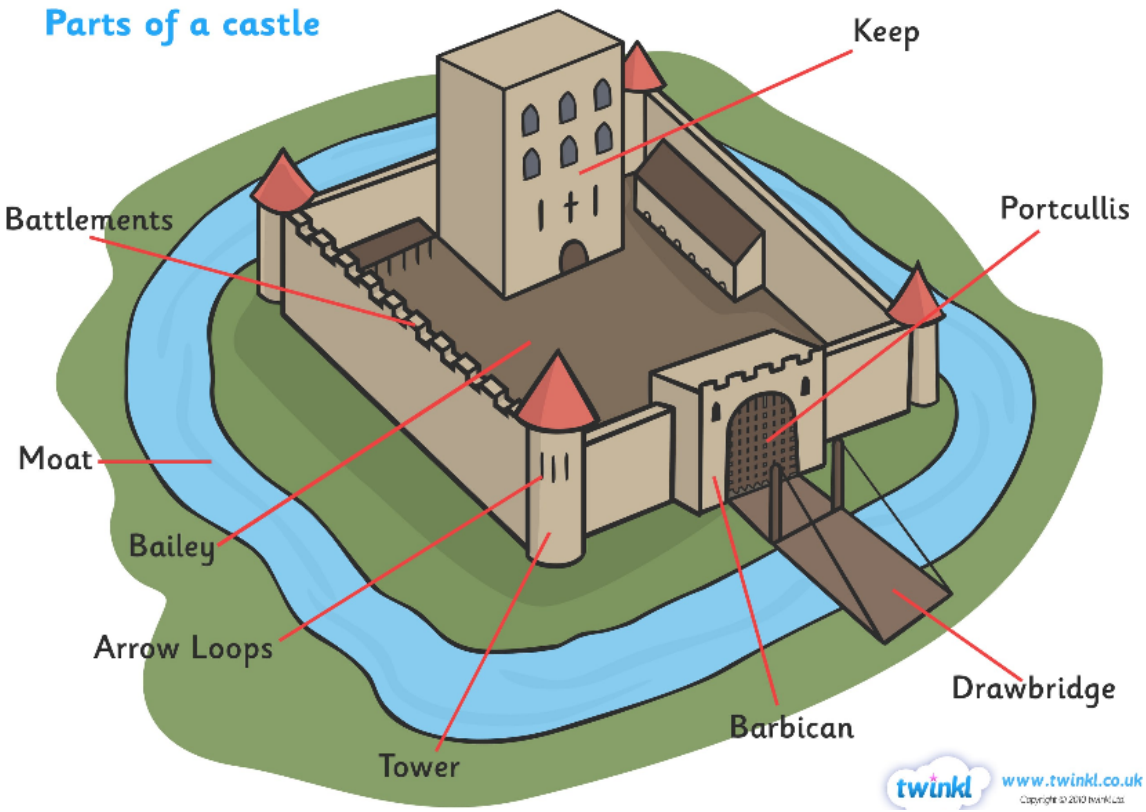
It's a balance of risk, usability and cost

Security Design Principles

- Defense in depth
- Deny by default
- Least privilege principle
- Complex = insecure
- Security, not obscurity

Defense in depth

How can you avoid a single point of failure? Where should you keep your assets?



Deny by default

Use whitelisting rather than blacklisting

```
def isAllowed(user):
  allowed = true
  try:
    if (!inFile(user, "admins.xml")): allowed = false
  except IOError: allowed = false
  except: pass
  return allowed
```

No!

What if XMLError is thrown instead?

```
def isAllowed(user):
  allowed = false
  try:
    if (inFile(user, "admins.xml")): allowed = true
  except: pass
  return allowed
```

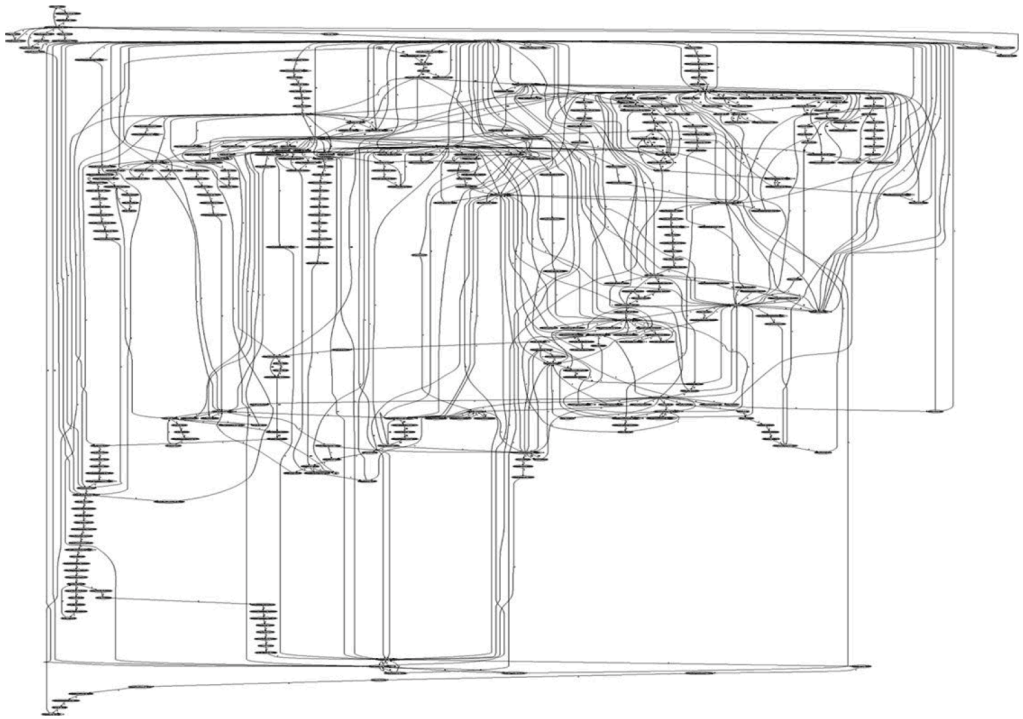
Yes

Least privilege principle

"Need to know" basis: require, grant and use only the privileges that are really needed

Complex = insecure

Maintenance of complex code leads to vulnerabilities



System calls in Apache

Security by obscurity

What is it? Hiding design or implementation details to gain security:

- e.g. hiding a DB server under a name different from “db”, etc.
- e.g. keeping the encryption algorithm secret, instead of the key

Security by obscurity

The idea doesn't work

- It's difficult to keep secrets (e.g. source code gets stolen, Google indexes hidden pages...)
- If security of a system depends on a secret that's revealed, the whole system is compromised
- Secret algorithms, protocols etc. will not get reviewed, flaws won't be spotted and fixed, less security

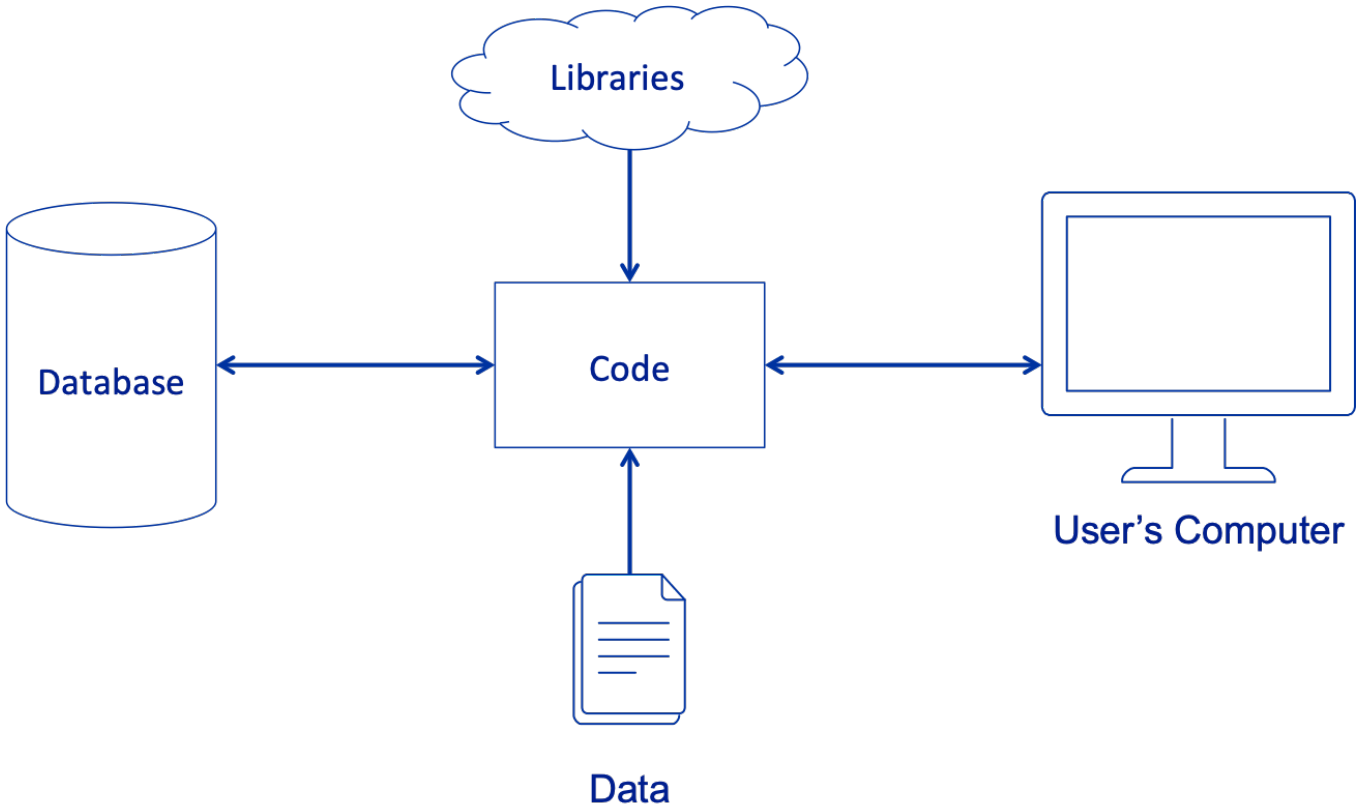
Systems should be secure by design, not by obfuscation!

Guidelines and Principles - Summary

- Security is a balance of risk, usability and cost
- The Security Design Principles discussed will help you prioritise security
- Ensure Security Design Principles are included from the very beginning of a software project

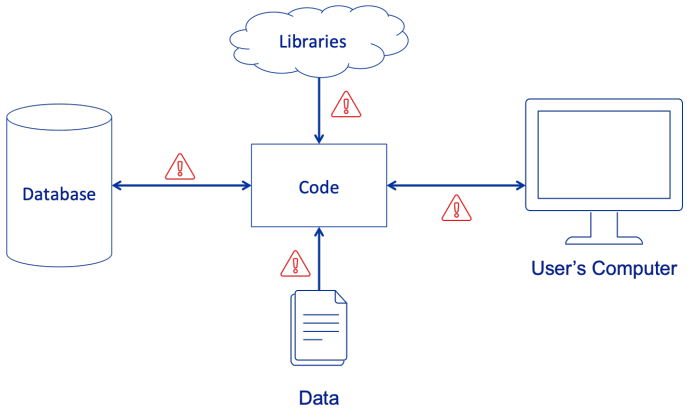
Software Vulnerabilities

Software Vulnerabilities



Software Vulnerabilities

- Malicious libraries
- Unexpected code hidden in data files
- Unchecked input data from users (SQL or command injection)
- ...



Software Vulnerabilities - Injection

```python
#BAD
import subprocess

def transcode_file():
    filename = raw_input('Please provide the path for the file to transcode: ')
    command = 'ffmpeg -i "{}" output_file.mpg'.format(filename)
    subprocess.call(command, shell=True)  # a bad idea!
```

https://www.kevinlondon.com/2015/07/26/dangerous-python-functions.html

What if the file were called ”; rm -rf /”

Software Vulnerabilities - Injection

```python
#OK
import subprocess
from shlex import quote

def transcode_file():
    filename = raw_input('Please provide the path for the file to transcode: ')
    command = 'ffmpeg -i "{}" output_file.mpg'.format(quote(filename)) # quote!
    subprocess.call(command, shell=False)  # do not allow the default shell!
```

Software Vulnerabilities - Libraries

- When was the library last modified?
- How many people use the library?
- Is it hosted securely?
- Is the name correct? E.g. "djago", "diango", "dajngo", impersonated the real library "django"

Software Vulnerabilities - Summary

- Validate user input and data very carefully
- Vet third party libraries before using and keep updated
- Look up major security flaws for the languages you use, e.g. "Best Practices for Using R Securely"
- Use tools to help you (e.g. static analysis tools, dependency checkers etc.)

Data Privacy

Data Protection

As a Data Scientist, you may be collecting Personal Information. If this data is not treated according to the law, you may be liable for significant fines.

- Many countries have their own Data Protection laws
- The EU General Data Protection Regulation is applicable to anyone physically located in the EU
- Certain research communities require approval from ethics boards for data collection

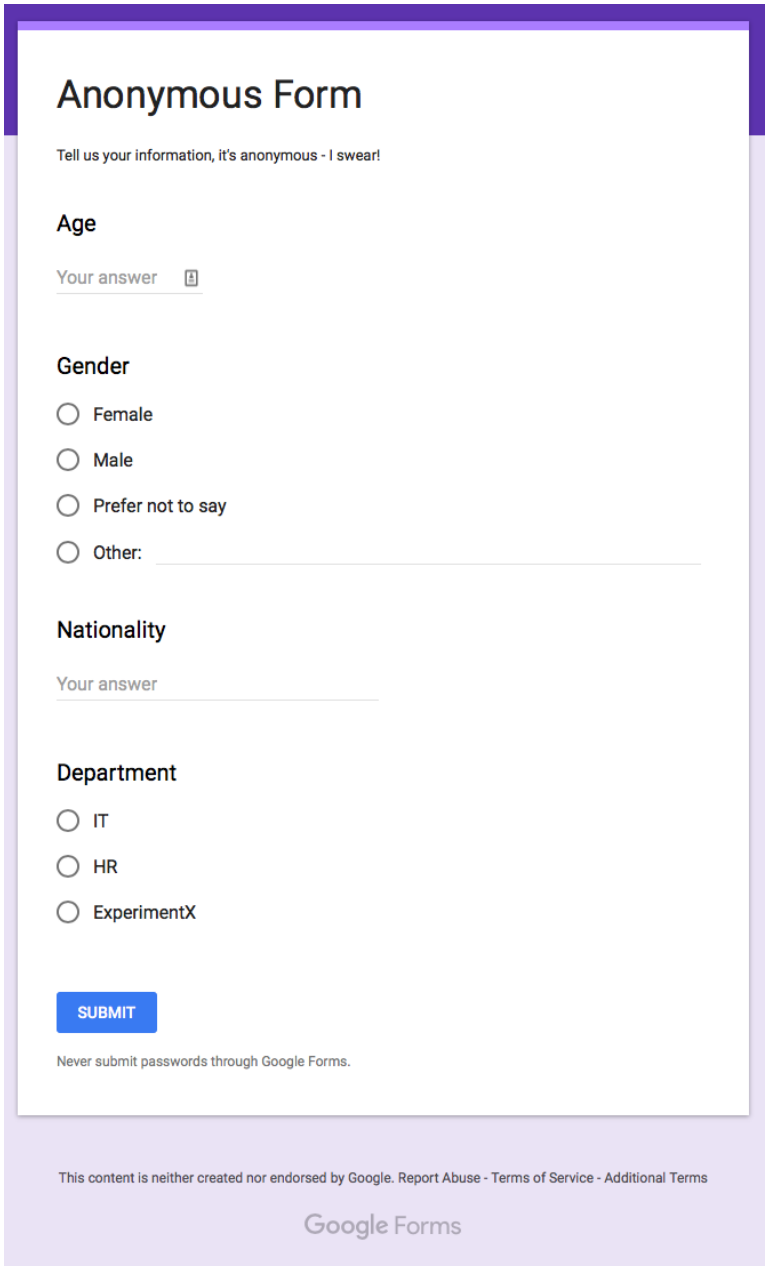
Data Protection

Best Practices

- **Minimise** Data Collection
- Be **transparent**; why are you collecting the data? Which data are you collecting? How will you share it? How long will you keep it?
- Treat the data with **respect**; store it securely, anonymise it when possible
- Make it clear how data owners can **retrieve** their data, or request **modification** or **deletion**

Anonymisation

- Even if you anonymise the name, are individuals still identifiable from the data?
- If you convert names to anonymous strings, can you get back to the name?



Anonymous Form

Tell us your information, it's anonymous - I swear!

Age

Your answer

Gender

- ○ Female
- ○ Male
- ○ Prefer not to say
- ○ Other:

Nationality

Your answer

Department

- ○ IT
- ○ HR
- ○ ExperimentX

SUBMIT

Never submit passwords through Google Forms.

Data Privacy - Summary

- Minimise the collection of privacy impacting data
- Be transparent about data processing and transfer

Questions?

- Ask now
- Find us during the break
- You are welcome to contact us after the school

Credits

- Sebastian Lopienski (CERN IT) for security principles
- Stefan Lueders (CERN IT) for threats
- Hannah Short (CERN IT) for identity aspects


CERN