



# Introduction to Cybersecurity

# Plan

---

- About this course
- Security model and goals
- Security principles
- Introduction to group theory



# About Me:

## Bernardo David

- Research in cryptographic protocols for privacy preserving computation and blockchains
- Office: 4C02
- [beda@itu.dk](mailto:beda@itu.dk)



# Staff

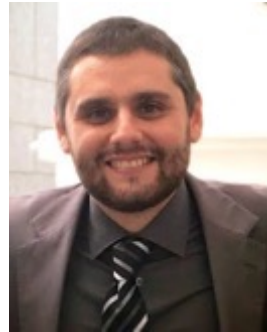
- Bernardo David ([beda@itu.dk](mailto:beda@itu.dk))

Lecturer and course manager. Office: 4C02.



- Rosario Giustolisi ([rosg@itu.dk](mailto:rosg@itu.dk))

Lecturer. Office: 4C16



## Teaching Assistants:

- |                           |                        |
|---------------------------|------------------------|
| - Viktor Máni Mønster     | - Bjarke Brodin Larsen |
| - Emilia Victoria Helsted | - Otto Jacobsen        |

# Contents

The principal security **requirements** and **attacker** models

The fundamental **cryptographic tools** in cybersecurity

Primary **security protocols** and Internet standards (PKI, TLS)

Basic techniques for **penetrating** and **hardening** IT-systems



# What You Will Not Learn



# Book

1. Main Textbook:  
<https://textbook.cs161.org>

## Computer Security

Search Computer Security

CS 161 Dark Mode

- Introduction
- Security Principles
- Memory Safety
- Cryptography
- Web Security
- Network Security
- Glossary

## Computer Security

By David Wagner, Nicholas Weaver, Peyrin Kao, Fuzail Shakir, Andrew Law, and Nicholas Ngai

Additional contributions by Noura Alomar, Sheqi Zhang, and Shomil Jain

This is the textbook for CS 161: Computer Security at UC Berkeley. It provides a brief survey over common topics in computer security including memory safety, cryptography, web security, and network security.


### Corrections

As of the Spring 2023 semester, this textbook is still being actively maintained and updated. Please contact [cs161-staff@berkeley.edu](mailto:cs161-staff@berkeley.edu) for information regarding corrections.

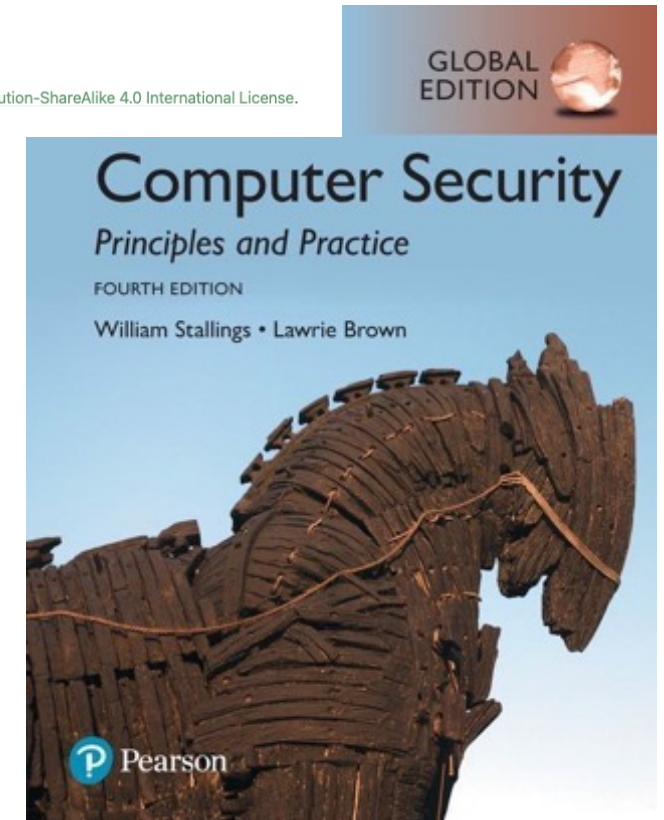
### Source and Changelog

The source for the textbook and a log of all changes is [available on Github](#).

### License

 This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).

2. Exercises (available on LearnIT as PDF):  
Computer Security  
Principles and Practice



# Learning Activities

Lectures

DIY Security on the command-line

Exercises

Hand-ins and Mini-projects  
(protocols, defense, offense)



# Mandatory Activities

You must submit and have approved 2 mandatory hand-ins.

You must submit and have approved 2 mini-projects

You must be present and participate actively in the course conclusion workshop

# Examination

Written, on-premises

Multiple-choice component,  
details will follow

**Prepare by doing exercises  
from Learnit**

## Exercise Sessions

Create groups as you like

**Important: Assign yourself  
(and your study group) to  
an exercise room! Follow  
link on LearnIT.**

# Communication

- Learnit forum: you can post anonymously or reveal your identity
- “No-email” policy: Post to the forum (except for private questions)
- Lectures: Bernardo (most) and Rosario (Penetration Testing)
- Exercises: TA group.

# Schedule

August 28 - Lecture 1 - Introduction and Security Principles

September 4 - Lecture 2 - Network Security

September 11 - Lecture 3 - Symmetric Cryptography

September 18 - Lecture 4 - Asymmetric Cryptography

September 25 - Lecture 5 - Secure Channels

October 2 - Lecture 6 - Privacy and Advanced Cryptography

October 9 - Lecture 7 - Workshop (focusing on Hand-in 2)

October 16 - Autumn Break

October 23 - Lecture 8 - Authentication and Access Control

October 30 - Lecture 9 - Pentesting I

November 6 - Lecture 10 - Workshop (focusing on Pentest project)

November 13 - Lecture 11 - Pentesting II

November 20 - Lecture 12 - Invited Lecture (TBA)

November 27 - Lecture 13 - Mandatory Project Presentations

December 4 - Lecture 14 - Conclusion



Load

7.5 ECTS

12 hr/week

**We want those 12 hours**

# Tips

- Do the work!
- Read the material, do the exercises, ...
- Emphasise learning *and using* the vocabulary
- Adversary, party, resource, trust, ...

# Extra activities

## Ethical hacking club @ITU

<https://www.facebook.com/groups/687644211422418/>

## Cybersecurity talks @ITU

<https://mailman.itu.dk/mailman/listinfo/cybersecurity>

## OWASP events

<https://www.meetup.com/OWASP-Copenhagen-Chapter>

## Hack the Box

<https://www.hackthebox.eu>

# Questions?



WHAT  
DO YOU  
MEAN  
?

**You must provide feedback**

Otherwise we'll just keep doing it wrong.







# Security Model

What is  
Cybersecurity?

What is  
Cybersecurity?

*The protection  
of computer  
systems*

What is  
Cybersecurity?

*The protection of  
computer systems*

*Protection from  
whom?*

What is  
Cybersecurity?

*The protection of  
computer systems*

*Protection from whom?*

**Beware the adversary!**







**Vandals**





**Activists**





**Criminals**



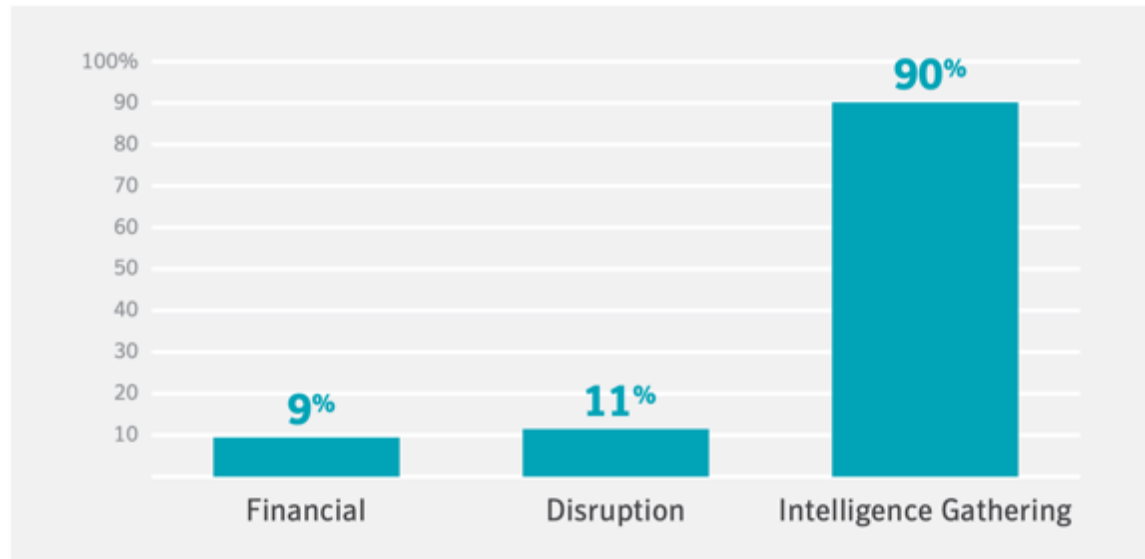


States

# A word on criminal and states

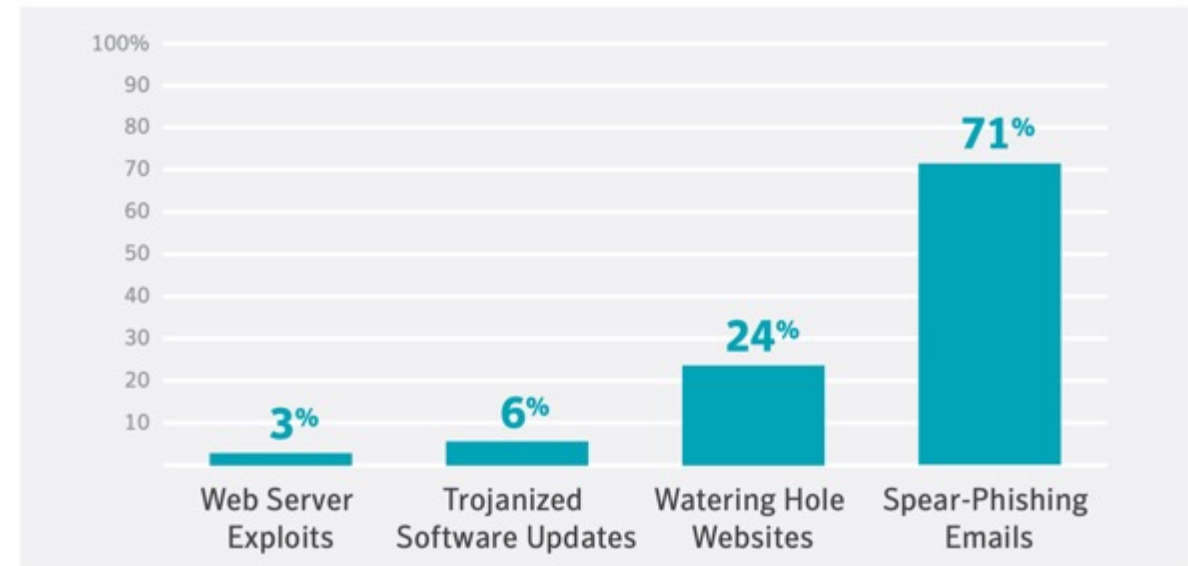
## Targeted attack motives

Known motives of targeted attack groups. The majority of groups are focused on intelligence gathering.



## Targeted attack infection vectors

Known infection vectors used by targeted attack groups. Spear phishing is by far the most popular.





# FE (CFCS) threat assessment

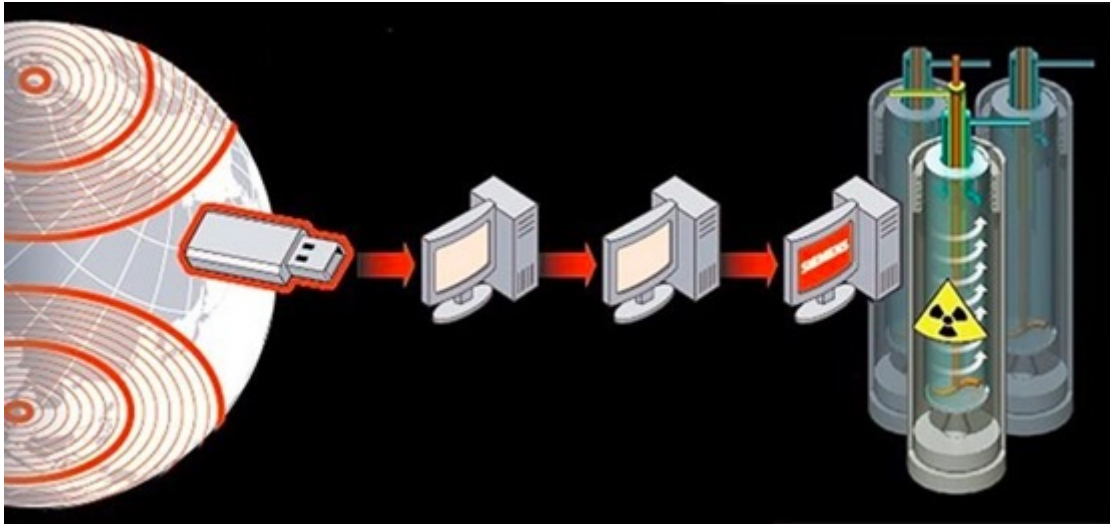
- CFCS vurderer, at truslen fra cyberspionage forsat er **MEGET HØJ**. Det er meget sandsynligt, at danske myndigheder og virksomheder vil blive ramt af forsøg på cyberspionage inden for de næste to år. Det er særligt Rusland og Kina, der benytter cyberangreb til at få viden...
- På samme måde er truslen fra cyberkriminalitet **MEGET HØJ**. Cyberkriminalitet rammer bredt, og vi vurderer, at de økonomisk motiverede cyberkriminelle ofte er velorganiserede og robuste overfor myndighedernes indgriben.
- Truslen fra cyberaktivisme er nu **HØJ** – og altså på det højeste niveau, siden CFCS i 2016 udgav den første årlige trusselsvurdering. Truslen kan kædes direkte sammen med krigen i Ukraine og de mange pro-russiske hackere, der tyer til tastaturet for at vise deres støtte til Rusland...
- Truslen fra destruktive cyberangreb er **LAV**. Vi vurderer fortsat, at det er mindre sandsynligt, at danske myndigheder og virksomheder vil blive mål for et destruktivt cyberangreb...
- Endelig er truslen fra cyberterror fortsat **INGEN**. Cyberterror er alvorlige cyberangreb, der skal opnå samme effekt som konventionelle terrorangreb. Truslen har været **INGEN** flere år i træk, men CFCS følger den tæt, fordi Center for Terroranalyse ved PET vurderer, at truslen fra konventionel terror mod Danmark er i niveauet alvorlig.

# Security Assumptions

Each (technical/non-technical) assumption is a potential vulnerability

# Security Assumptions

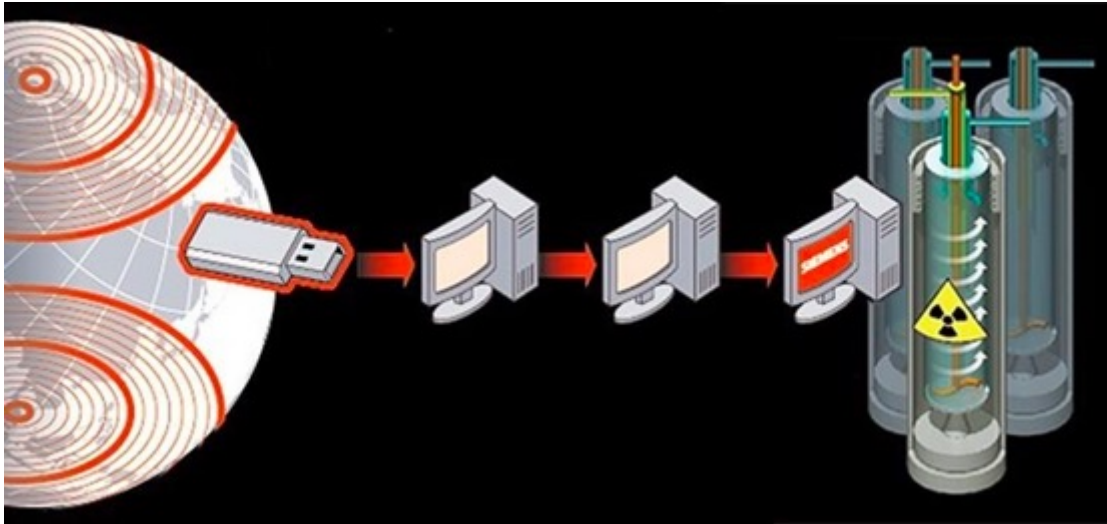
Each (**technical**/non-technical) assumption is a potential vulnerability



Stuxnet was a 500 Kb worm that exploited four zero-day flaws to destroy 984 uranium enriching centrifuges in Iran.

# Security Assumptions

Each (technical/**non-technical**) assumption is a potential vulnerability



Stuxnet was a 500 Kb worm that exploited four zero-day flaws to destroy 984 uranium enriching centrifuges in Iran.



# Security Assumptions

Password must be at least 16 characters

Password must have:

- English uppercase characters (A - Z)
- English lowercase characters (a - z)
- Punctuation (eg. !, \$, #, %)
- Numbers (0 - 9)

Password must be changed every 3 months

Password must not be copied and pasted

...

# Security Assumptions

Pa

Pa

•

•

•

•

Pa

Pa

...



Assumptions will be broken!



# Security Goals

# Confidentiality

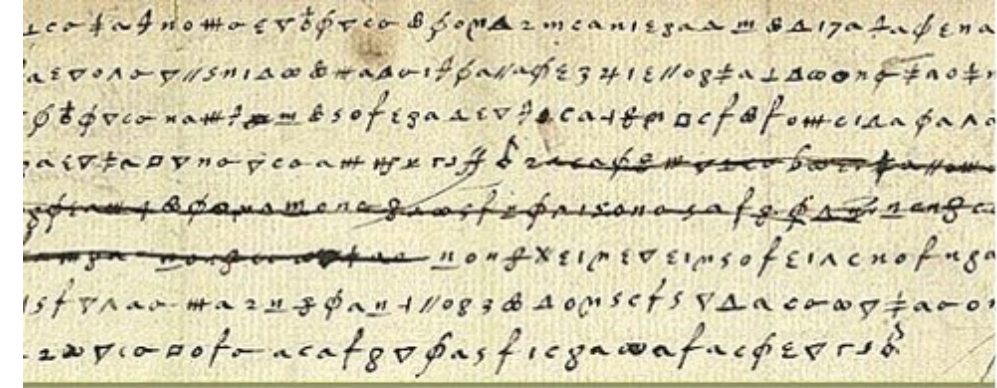
- Attacks: *eavesdropping, man-in-the-middle*



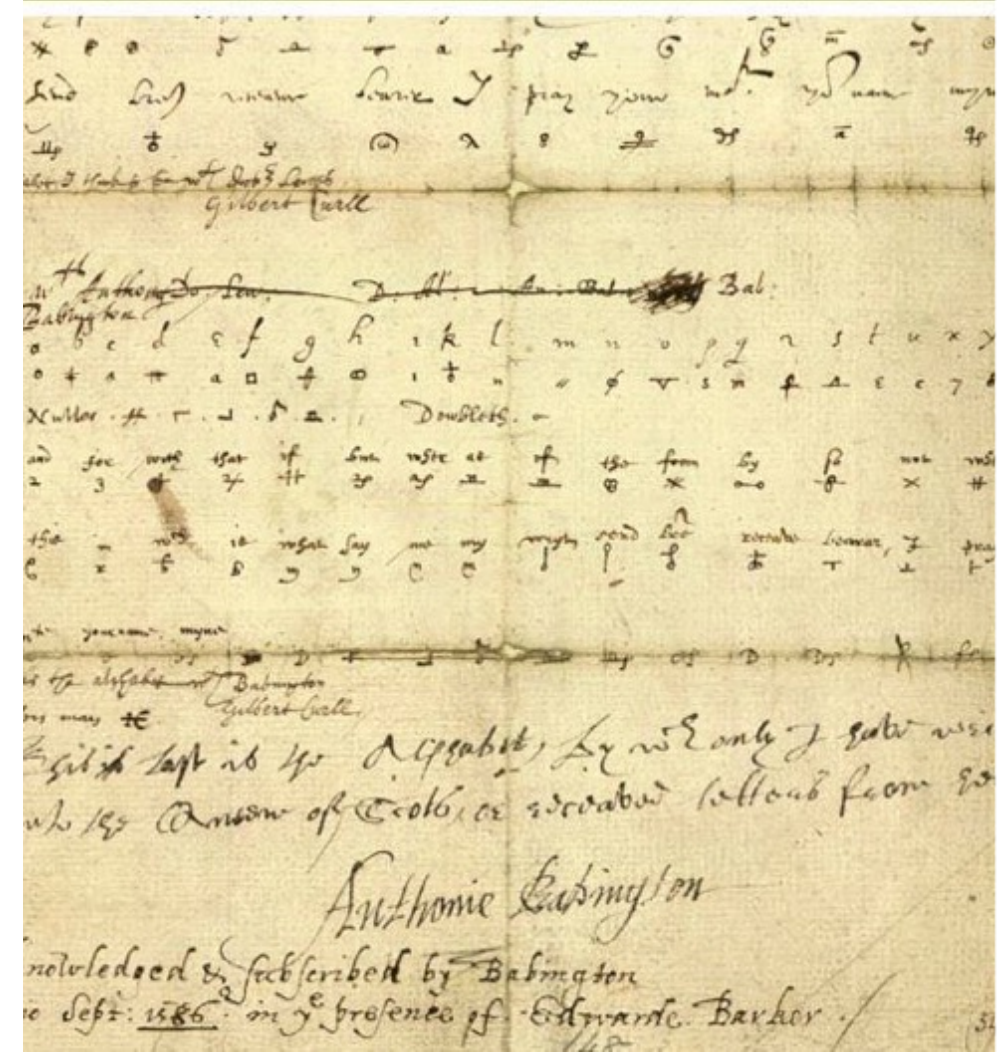


# Integrity

- Attacks: *masquerading, message tampering, replaying*
- July 17, 1586: Thomas Phelippes confounds the Babington plot to murder Queen Elisabeth and install Queen Mary as regent.
- He intercepted and decrypted a letter, then added:
- “I would be glad to know the names and qualities of the six gentlemen which are to accomplish the [deed], ...”



Handwritten text in a cipher, likely the Babington Plot letter, showing several lines of text in a non-standard script.

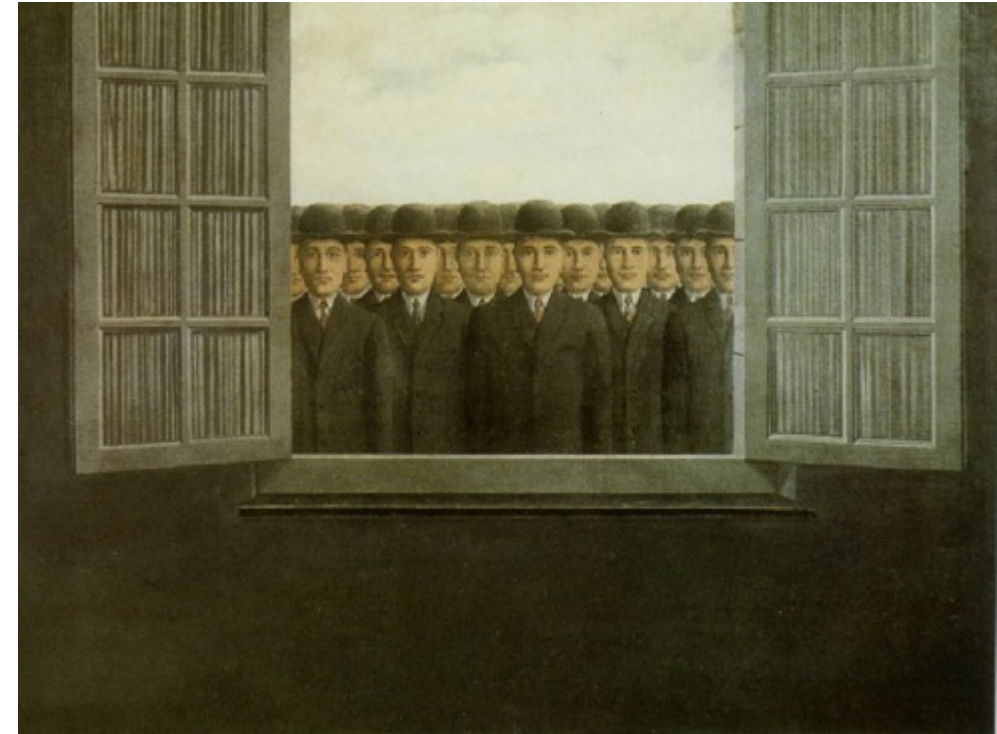


Handwritten text in a cipher, likely the Babington Plot letter, showing several lines of text in a non-standard script. The text is written in a cursive hand and includes a signature at the bottom.

# Availability

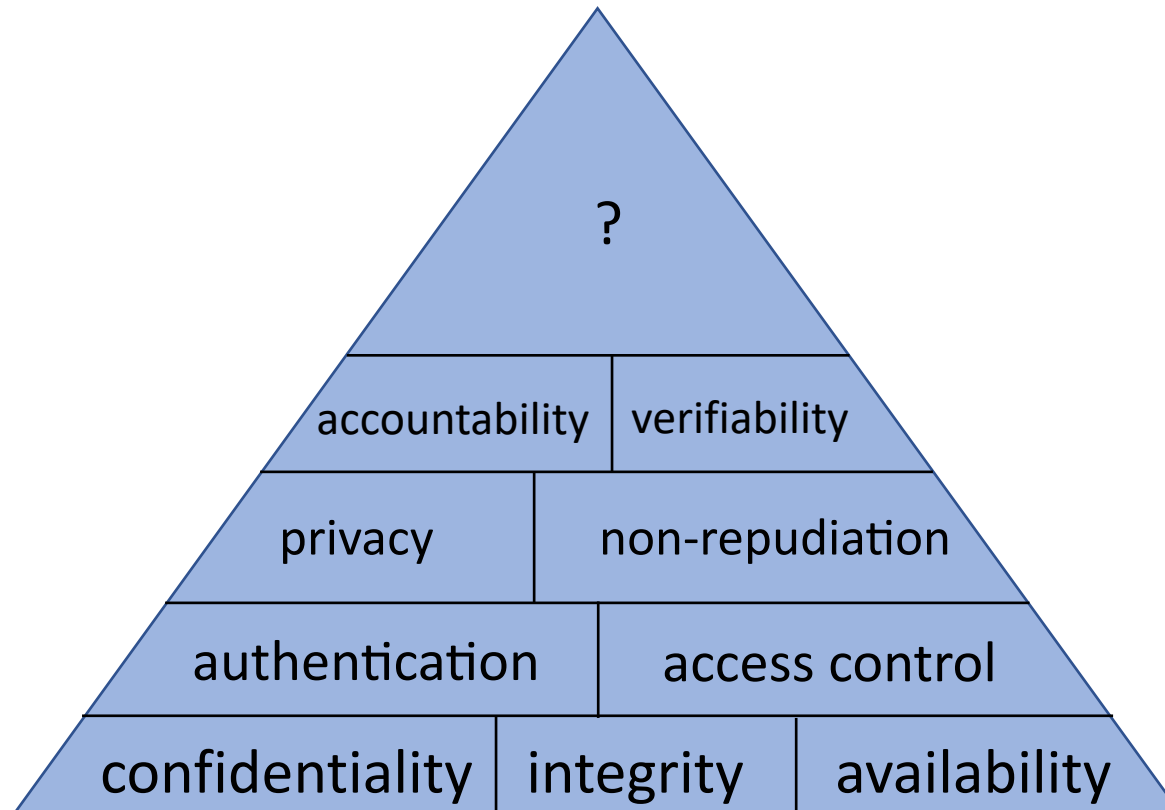
Attacks: *Denial of Service, distributed denial of service*

- December 19-21, 2018: Gatwick drone disruption cost easyJet nearly \$20 million



# Security Goals

- What does it mean that a system is secure?



# Security is impossibly hard

- You must defend against **all** possible attacks
- Adversary needs to find just **one** attack that works
- No perfect security
  - (...all possible attacks)
- Security is measured in the resources required of the adversary

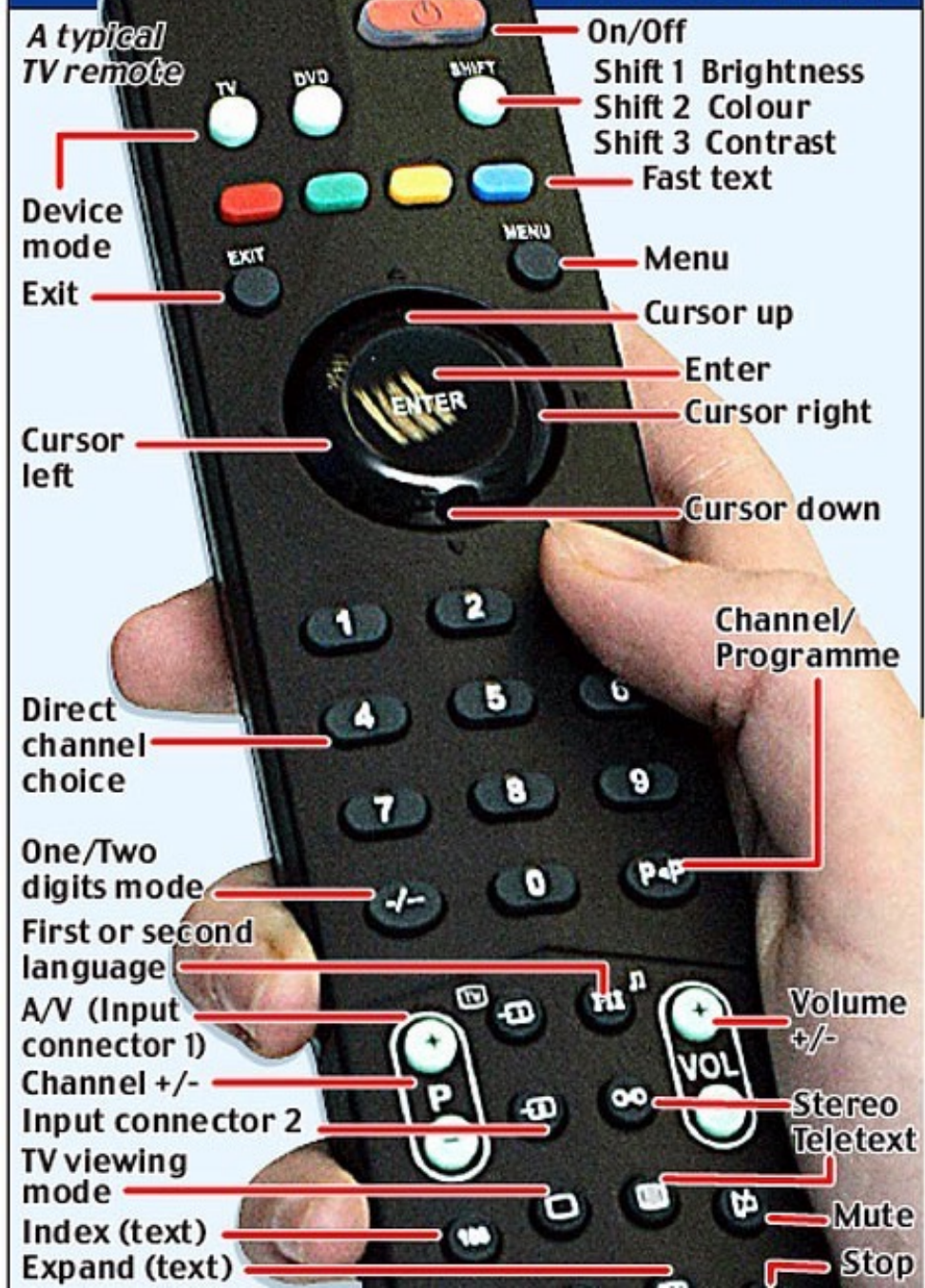




# Security Principles



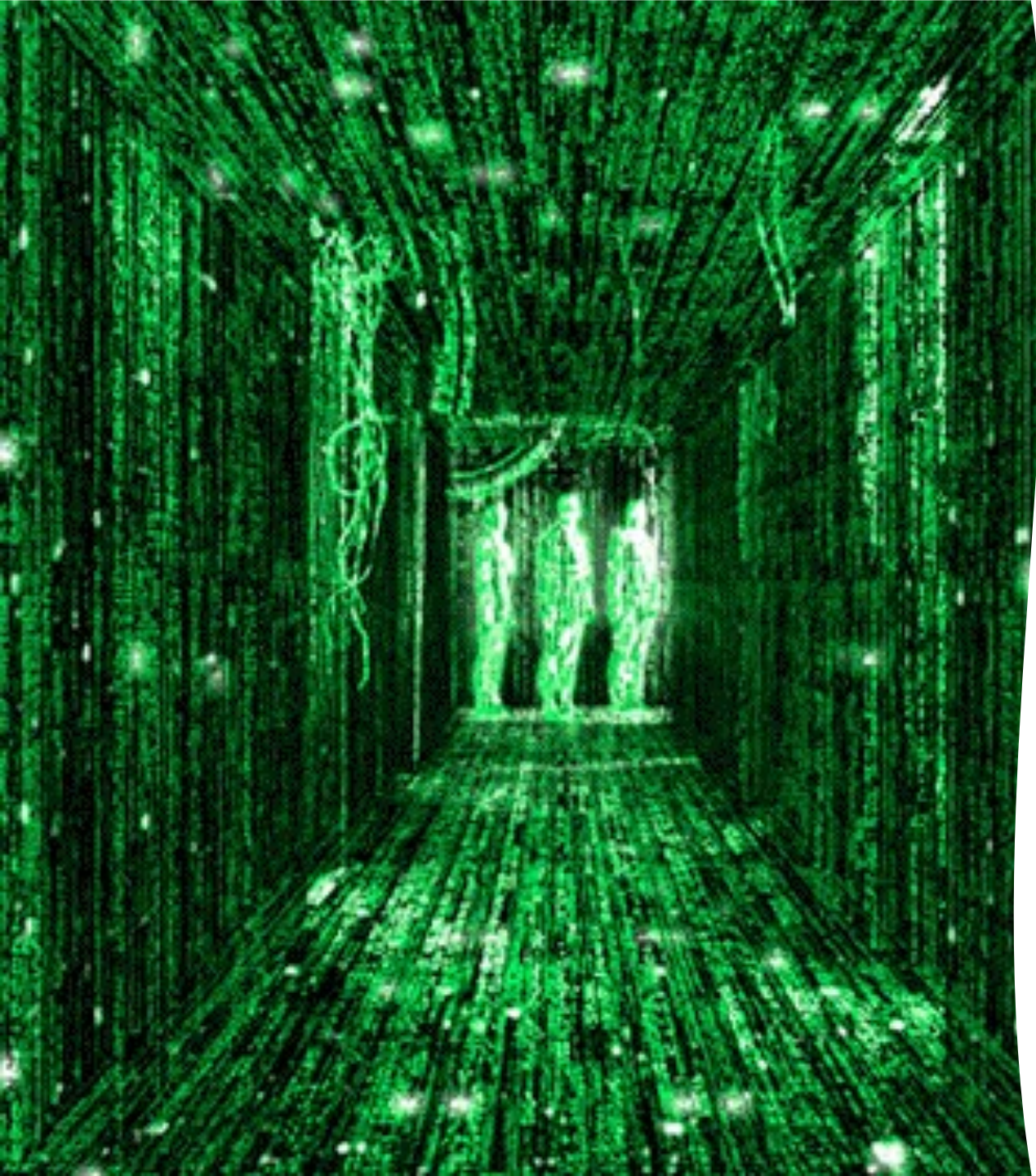
## 43 BAFFLING BUTTONS



## Economy of Mechanism

- “Keep it simple.”
- aka. “simplicity”
- General engineering principle:  
Complex designs yields complex failure analysis.

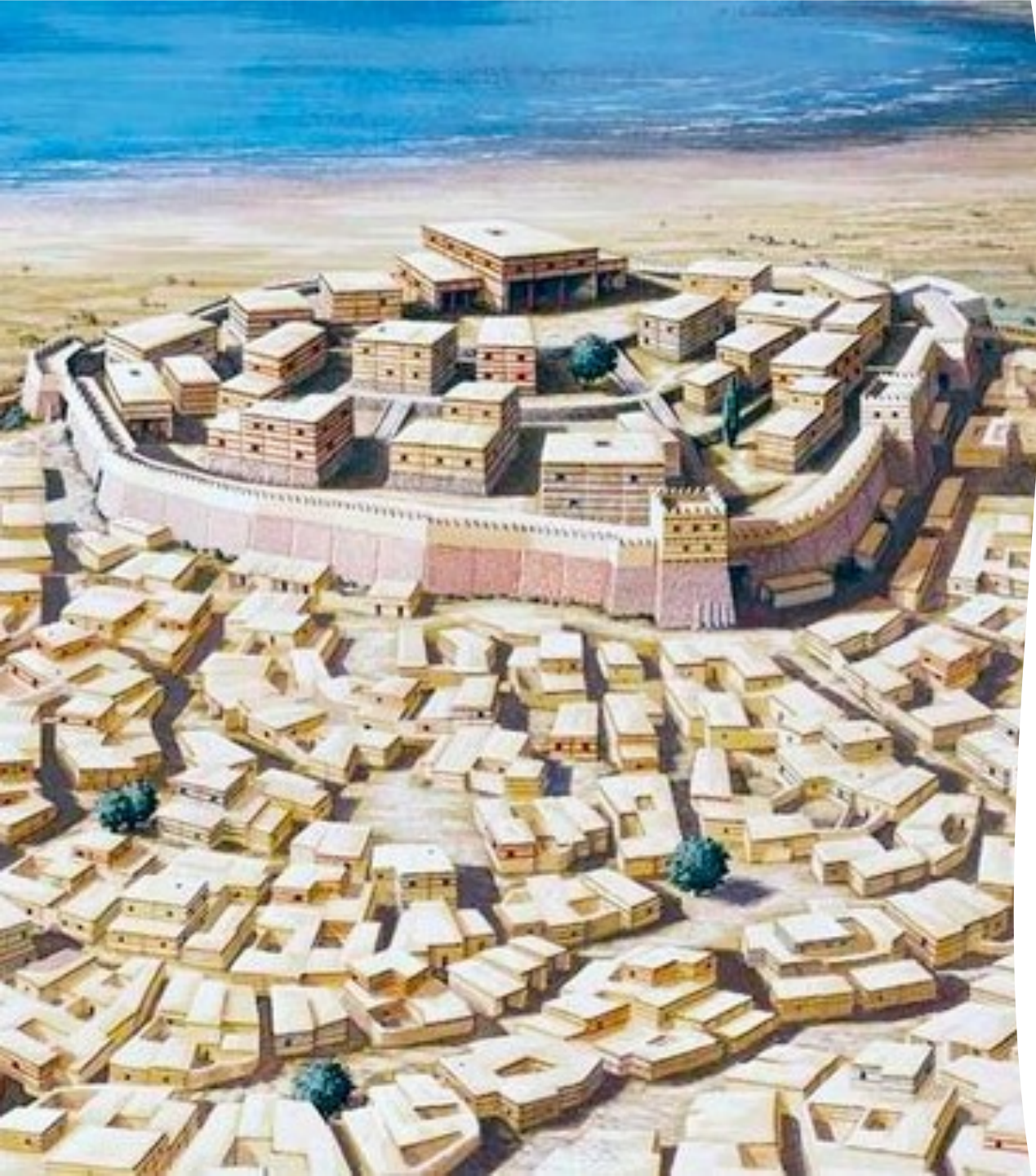




# Open design

- “The security of a system should not depend on the secrecy of its protection mechanisms.”
- aka “Kerckhoff’s principle”
- The adversary knows the system (Claude Shannon).
- Systems are hard to build—more scrutiny, less defects.
- Hard case: DRM. The user has the device. Sony compromises(!) consumers machines in 2005.





# Minimum exposure

- “Minimise the attack surface a system presents to the adversary.”
- Reduce external interfaces  
(If you don’t need it, turn it off.)
- Limit information
- Limit window of opportunity.





# Least privilege

- “Any component should operate using the least set of privileges necessary.”
- I don't have access to ITU mail servers.
- PowerPoint does not run as root.

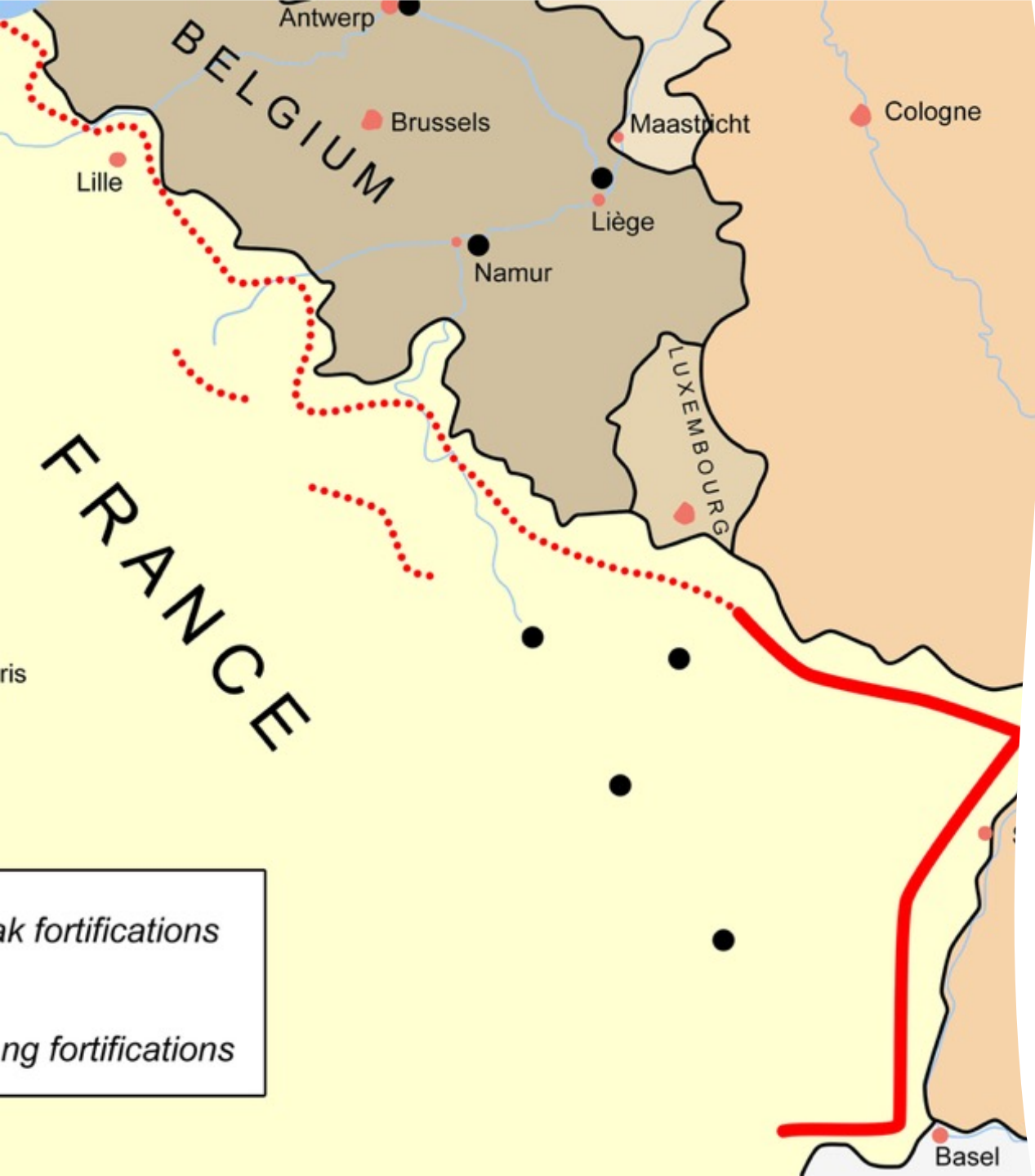


High above  
the city of L.A.  
a team of terrorists  
has seized a building,  
taken hostages, and  
declared war.  
One man has managed to escape...  
An off-duty cop hiding somewhere inside.  
He's alone, tired...  
and the only chance anyone has got.

BRUCE WILLIS  
**DIE HARD**

# Fail-safe defaults

- “The system should start in and return to a secure state in the event of a failure.”
- Allowed list, blocked list.
- If you lost connectivity to the authentication server, don't let anyone in while it's down.
- E.g., allowed lists of ports for firewalls



# Complete mediation

- “Access to any object must be monitored and controlled.”
- The Maginot-line: strong fortifications not extending all the way did not help.
- E.g., OS access control to files can be circumvented if you have access to the physical disk. (Use crypto, then.)



# No single point of failure

- “Build redundant security mechanisms whenever feasible.”
- aka “defence in depth.”
- Key technique: separation of duty





# Psychological acceptability

- “Design usable security mechanisms”
- ...let users circumvent them
- Help the user to make the *right* choice

## Wrapping up

- Adversary: **state**-sponsored cyber attacks
- Assumptions: as **few** as possible
- Goals: as **many** as possible
- Principles: as **many** as possible

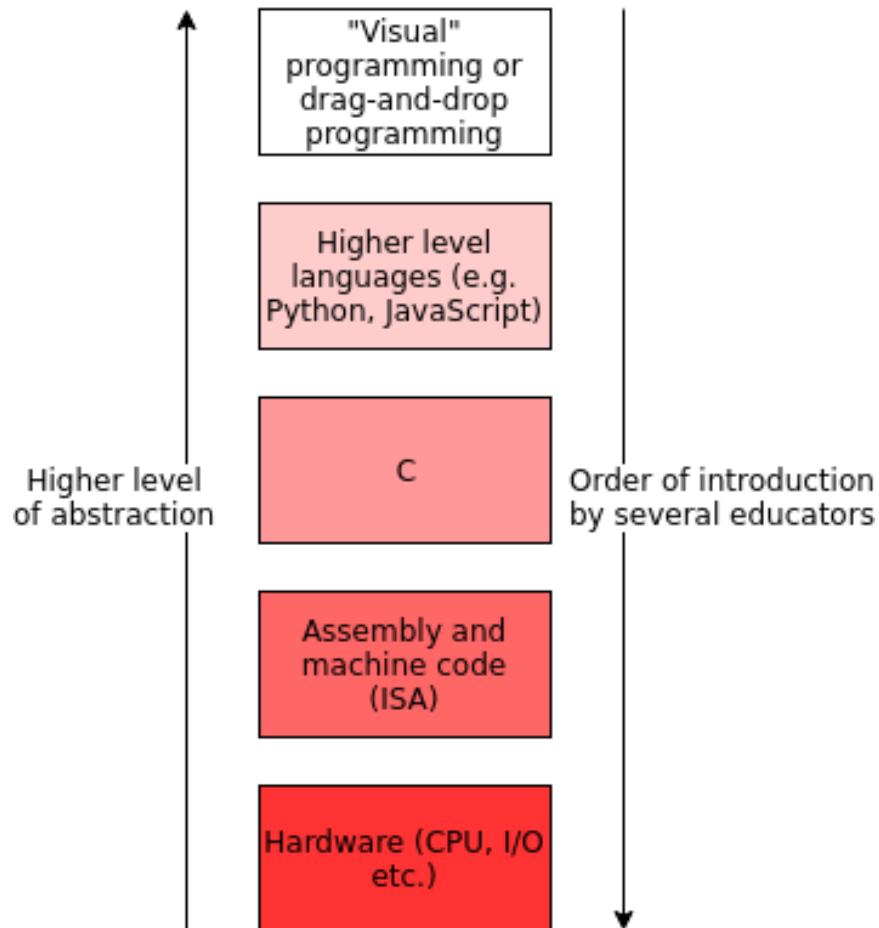




# Group Theory (To Be Continued in Asymmetric Crypto)

# Group Theory: Why?

- Abstraction makes our lives easier!



Abstract group (many operations):

$$g \cdot g = g^2, g^a \cdot g^b = g^{a+b}, h = g^t, h^a = g^{t \cdot a}$$

$$g^m \cdot h^r = c, c = g^{m+t \cdot r}$$

Group Representation for an elliptic curve point (secp256k1):

$$g = (x, y) \text{ such that } y^2 = x^3 + 7 \pmod{115792089237316195423570985008687907853269984665640564039457584007908834671663}$$

Concrete Numbers for an element:

$x =$

(55066263022277343669578718895168534326250603453777594175500187360389116729240)

$y =$

(32670510020758816978083085130507043184471273380659243275938904335757337482424)



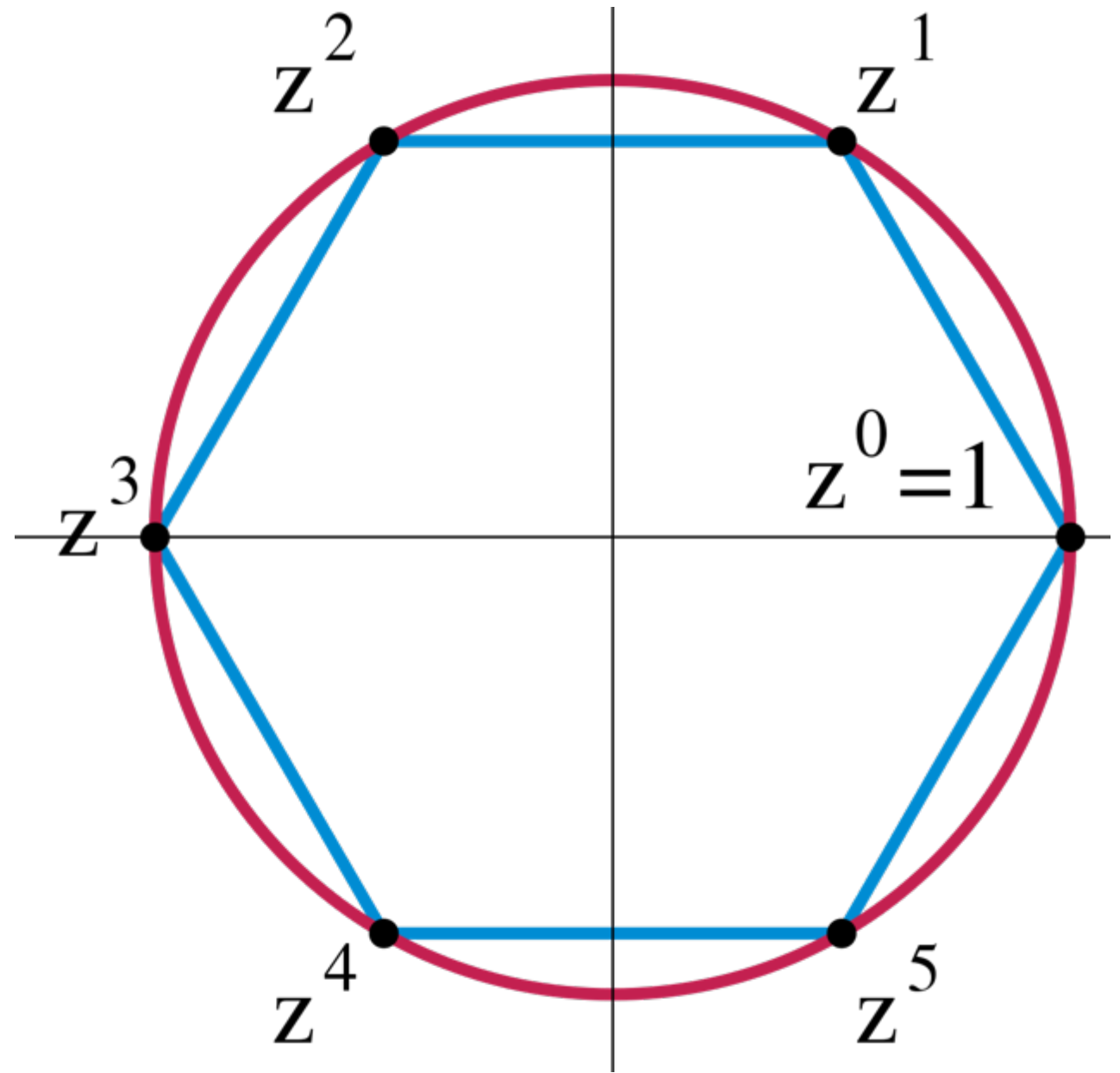
# (Abelian) Groups

- An abelian *group* is a pair  $(G, \circ)$  where  $G$  is a set and a binary operation  $\circ$  defined on  $G$  such that:
  - (Closure) For all  $g, h \in G$ ,  $g \circ h$  is in  $G$
  - There is an identity  $e \in G$  such that  $e \circ g = g$  for  $g \in G$
  - Every  $g \in G$  has an inverse  $h \in G$  such that  $h \circ g = e$
  - (Associativity) For all  $f, g, h \in G$ ,  $f \circ (g \circ h) = (f \circ g) \circ h$
  - (**Commutativity**) For all  $g, h \in G$ ,  $g \circ h = h \circ g$
- The *order* of a finite group  $G$  is the number of elements in  $G$

# Group Operation

- The group operation can be written *additively* or *multiplicatively*
  - I.e., instead of  $g \circ h$ , write  $g+h$  or  $g \cdot h=gh$
  - Does *not* mean that the group operation corresponds to (integer) addition or multiplication
- Identity denoted by 0 or 1, respectively
- Inverse of  $g$  denoted by  $-g$  or  $g^{-1}$ , respectively
- Group exponentiation:  $m \cdot a$  or  $a^m$ , respectively
- In multiplicative groups, identity can be written  $g^0$

# Cyclic groups



# Cyclic Groups

- Let  $(G, \cdot)$  be a finite group of order  $q$  (written multiplicatively).
- Let  $g$  be some element of  $G$ .
- Consider the set  $\langle g \rangle = \{g^0, g^1, \dots\}$ .
- We know  $g^q = 1 = g^0$ , (Fermat's little theorem) so the set has  $\leq q$  elements.
- If the set  $\langle g \rangle$  has  $q$  elements (all elements in the group), then we say  $g$  is a generator of  $(G, \cdot)$ .
- A generator  $g$  “generates” all elements in the group when the group operation is applied to itself multiple times.
- If a group has a generator, then we say this is a *cyclic group*.

# Example: Multiplicative Group over positive non-zero Integers modulo a prime $p$ ( $\mathbb{Z}_p^*, \cdot$ )

- $\mathbb{Z}_p^*$  is the set of positive non-zero integers modulo  $p$ :  $\{1, \dots, p-1\}$ 
  - Remember that “ $a \bmod b$ ” means the remainder of the division of  $a$  by  $b$ .
- The operation in this group is multiplication mod  $p$ , denoted by  $\cdot$ .
- Concrete example (but useless for cryptography) ( $\mathbb{Z}_5^*, \cdot$ ) :
  - $\mathbb{Z}_5^* = \{1, 2, 3, 4\}$
  - Closure:  $1 \cdot 1 = 1, 1 \cdot 2 = 2, 1 \cdot 3 = 3, 1 \cdot 4 = 4, 2 \cdot 2 = 4, 2 \cdot 3 = 1, 2 \cdot 4 = 3, 3 \cdot 3 = 4, 3 \cdot 4 = 2, 4 \cdot 4 = 1$
  - Identity: 1 (property of integer multiplication), or  $1^0 = 1, 2^0 = 1, 3^0 = 1, 4^0 = 1$
  - Inverse:  $1 \cdot 1 = 1, 2 \cdot 3 = 1, 4 \cdot 4 = 1$ , also expressed as  $1^{-1} = 1, 2^{-1} = 3, 3^{-1} = 2, 4^{-1} = 4$
  - Associativity and commutativity: properties of integer multiplication
  - Generator: 2 (notice that  $2^1 = 2, 2^2 = 2 \cdot 2 = 4, 2^3 = 2 \cdot 2 \cdot 2 = 3, 2^4 = 2 \cdot 2 \cdot 2 \cdot 2 = 1$ )



# Summary

- About this course: use the forum, give us feedback
- Security model and goals: many adversary flavours, no perfect security.
- Security principles: make adversary life harder
- Group theory: in math we trust

[illegible]