



Computer Security

Academic Training, May 21-24

Sebastian Lopienski, Stefan Lueders,
Remi Mollon, Antonio Perez Perez
CERN Computer Security Team



Computer Security lectures

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ ***“Introduction to information and computer security”***
Sebastian Lopienski
Monday, May 21st
- ▶ **“Cryptography and authentication”**
Remi Mollon
Tuesday, May 22nd
- ▶ **“Computer security threats, vulnerabilities and attacks”**
Antonio Perez Perez
Wednesday, May 23rd
- ▶ **“Security operations at CERN”**
Stefan Lueders
Thursday, May 24th





Introduction to information security

► How to think about security...

Sebastian Lopienski
(CERN Deputy Computer Security Officer)
CERN Security Lectures, May 21th 2012



Let's learn some Chinese

Computer.Security@cern.ch — Computer Security Lectures 2012

人 person

囚 ?

女 woman

安 ?

Outline

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ **What we care about**
- ▶ **Where the problems (and solutions) are**
- ▶ **How to approach it**
- ▶ **How to prioritize**
- ▶ **Some golden rules**
- ▶ **A global image**





► What we care about

This is about the C.I.A.

▶ **Confidentiality**

▶ **Integrity**

▶ **Availability**

**Not only malicious attacks,
but also incidents,
hardware problems etc.**

Security needs / objectives

Computer.Security@cern.ch — Computer Security Lectures 2012

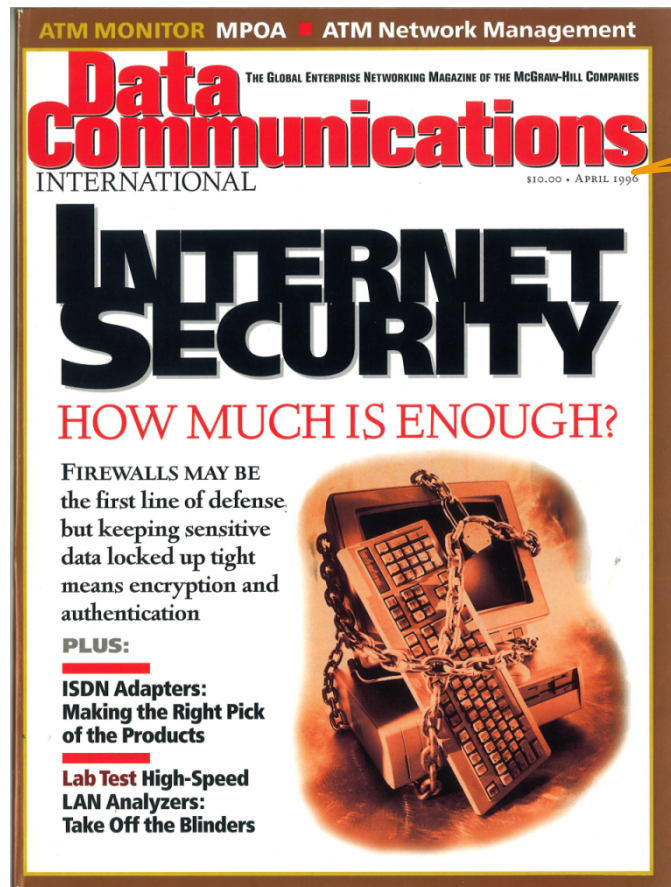
- ▶ **identification** (who is the person, server, software etc.)
- ▶ **authentication** (proving the identity)
- ▶ **authorization** (what is that person allowed to do)
- ▶ **privacy** (controlling one's personal information)
- ▶ **anonymity** (remaining unidentified to others)
- ▶ **non-repudiation** (user can't deny having taken an action)
- ▶ **traceability** (having traces/logs, knowing what happened)



► **Where the problems
(and solutions) are**

IT Security in 1996

Computer.Security@cern.ch — Computer Security Lectures 2012

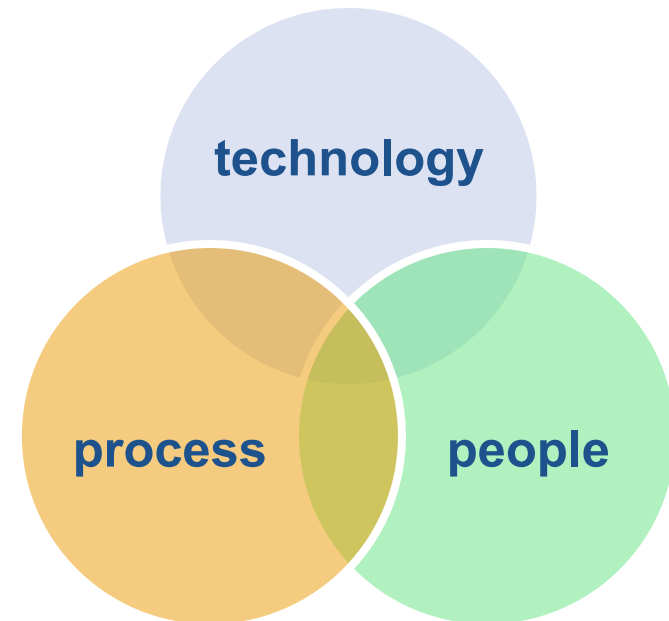


*“**firewalls** may be the first line of defense, but keeping sensitive data locked up tight means **encryption and authentication**”*

IT security - technology only?

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ **SQL Injection** is a technical vulnerability
- ▶ What could help to avoid or prevent it?
 - ▶ secure coding
 - ▶ using secure frameworks
 - ▶ vulnerability scanning
 - ▶ Web application firewalling
 - ▶ code review
 - ▶ security testing
 - ▶ secure development lifecycle
 - ▶ developer training



Technology, process, people

Computer.Security@cern.ch — Computer Security Lectures 2012

Three aspects of security:

▶ **Technology**

▶ **Processes**

▶ **People**



Process and organization

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ “*Security is a process, not a product*” - Bruce Schneier
- ▶ **Just deploying a security measure is not enough:**
 - ▶ antivirus software → virus signature updates
 - ▶ monitoring systems → checking, reacting to alarms
 - ▶ endpoint security → OS and software patching
 - ▶ security policies → updating, enforcing
 - ▶ risk management, vulnerability management, business continuity planning, security development lifecycle etc.
→ **ongoing processes**, not one-off exercises



Security is a process

Computer.Security@cern.ch — Computer Security Lectures 2012



Security solutions often **degrade with time**
– they need to be **verified periodically**

... and it's about people

Computer.Security@cern.ch — Computer Security Lectures 2012



NO PERSON SHALL, ON A FRIDAY, SATURDAY
OR SUNDAY THE DAY PRECEEDING A
PUBLIC HOLIDAY, OR ON A PUBLIC HOLIDAY,
DRIVE OR CAUSE TO BE DRIVEN BETWEEN
THE HOURS OF 6 P.M. AND MIDNIGHT. A
MOTOR VEHICLE WHICH EXCEEDS 10.5 M
IN LENGTH IN ALL MAIN ROADS

Maybe it is correct
– but **can anyone follow it?**

ODDLYSPECIFIC.COM



Last but not least: humans

Computer.Security@cern.ch — Computer Security Lectures 2012

People...

- ▶ have **flawed risk perception**
- ▶ are bad in dealing with **exceptions and rare cases**
- ▶ can't take **correct security decisions**
- ▶ put too much **trust in their computers**
- ▶ easily fall for **social engineering**
- ▶ sometimes **turn malicious**

- ▶ prefer **convenience and bypass security measures**
- ▶ often **make mistakes**

from Schneier „Secrets and Lies”



Flawed risk perception

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ Is flying more dangerous than travelling by car?



- ▶ Is it more likely to get killed by a shark or by a pig or a coconut?



- ▶ Is using 1024 bits key length enough for encryption?

- ▶ “... what is key length???”

Taking security decisions...

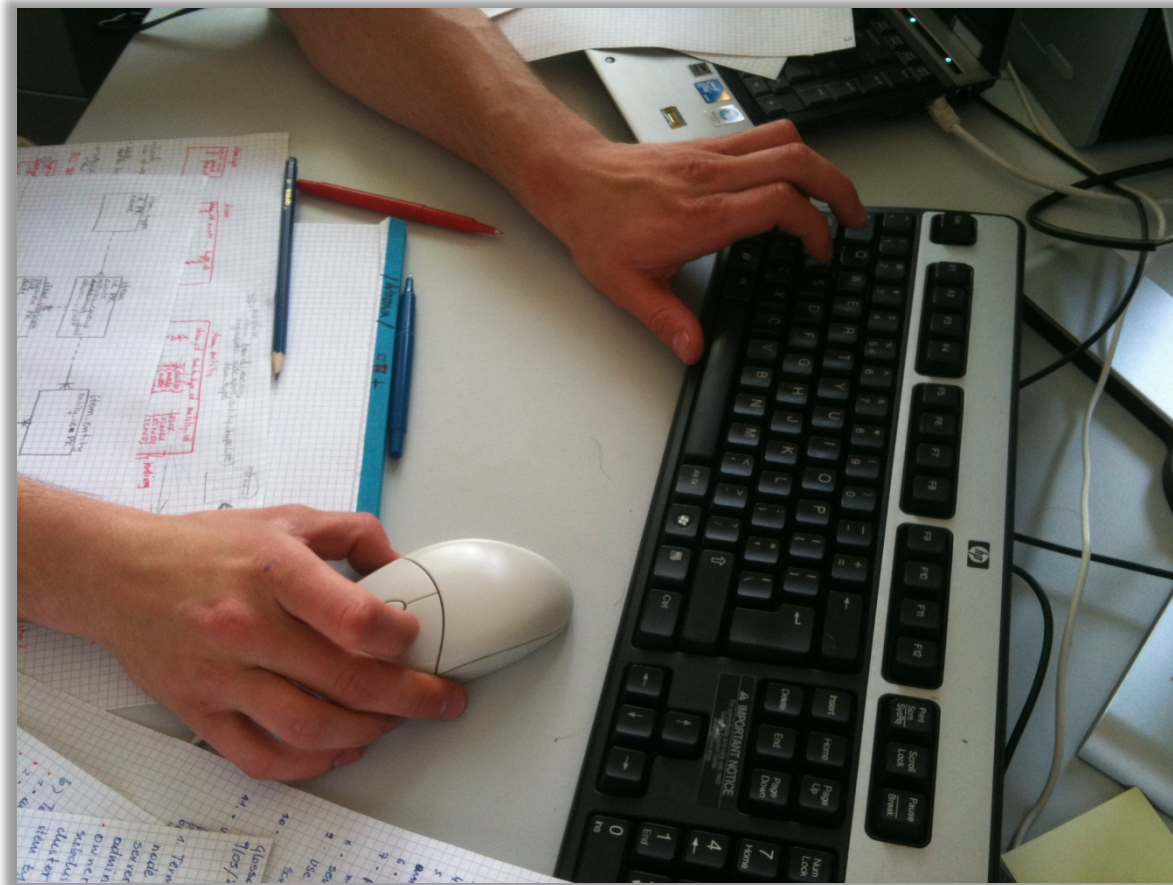
Computer.Security@cern.ch — Computer Security Lectures 2012



People and technology

Computer.Security@cern.ch — Computer Security Lectures 2012

People will always find ways to use technology
not the way it was designed to be used



Social engineering



lectures 2012

Social engineering

Computer.Security@cern.ch — Computer Security Lectures 2012

▶ First step is **information gathering**

- ▶ public and semi-public information: employees' names, what's the hierarchy, who's on a leave, projects names etc.
- ▶ to be used during the attack

▶ A social engineer then:

- ▶ uses influence, persuasion or threat
- ▶ abuses **people's compassion, fear or greed**
- ▶ exploits their **tendency to trust and to help**

in order to deceive them and achieve his goals:

- ▶ to **gain unauthorized access** to systems
- ▶ to obtain more information, data or knowledge

▶ ***“Amateurs hack systems, professionals hack people”***

Social engineering techniques

Computer.Security@cern.ch — Computer Security Lectures 2012

- ▶ **Pretexting** – inventing a scenario to explain a request
- ▶ **Taking believable roles** – to pass for someone else
 - ▶ an employee, a boss, a colleague, ...
- ▶ **Phishing** – to get one's password
- ▶ **Tailgating** – to pass a security gate
 - ▶ following someone closely or joining a legitimate group
- ▶ **Quid pro quo** – something for something
 - ▶ *"I will help you here; can you help me there?"*
- ▶ **Baiting** – leaving a trap to be taken by the victim
 - ▶ e.g. leaving infected USB sticks at victim's location



Safety culture

Computer.Security@cern.ch — Computer Security Lectures 2012



A change in **behaviour**,
but also in **beliefs and risk perception**
... a **culture change**



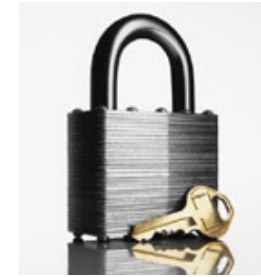
▶ How to approach it

Protection - detection - response

Computer.Security@cern.ch — Computer Security Lectures 2012

Three complementary steps to get more secure:

▶ **Protection/prevention**



▶ **Detection**



▶ **Response**



Make sure not to forget
detection and **response**

► How to prioritize

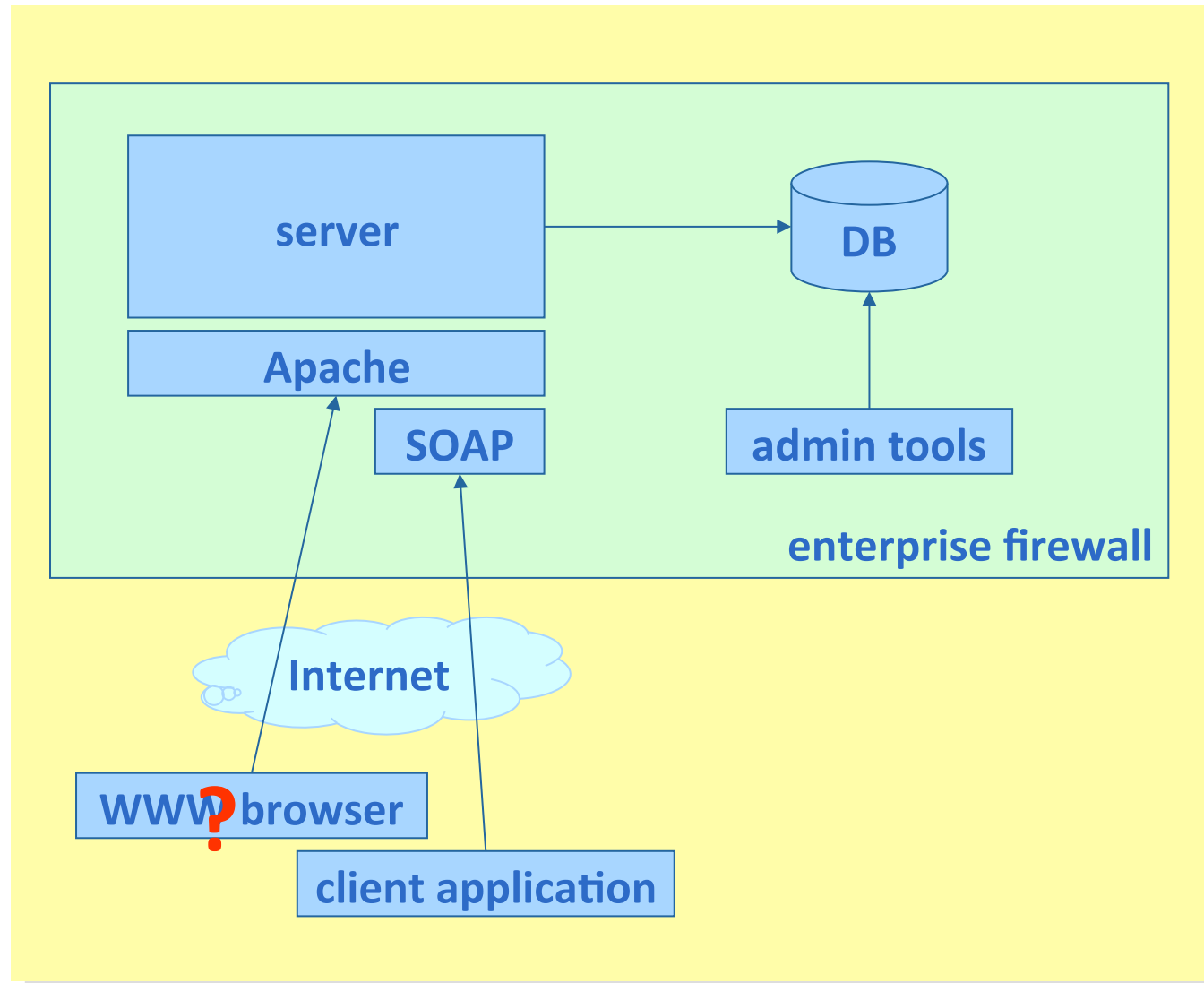
Secure against what and from whom?

What threats will the system face:

- ▶ what assets we want to protect?
- ▶ what could go wrong?
- ▶ who may want to attack and why?
- ▶ what is the exposure area?
- ▶ how is it protected?
- ▶ how could this be broken, made to fail, or bypassed?
- ▶ what are possible attack paths?

Threat modeling

Computer.Security@cern.ch — Computer Security Lectures 2012



Things to avoid

Computer.Security@cern.ch — Computer Security Lectures 2012

FAIL



Security solutions
that do not cover the
whole exposure area

From threat and asset to risk

Computer.Security@cern.ch — Computer Security Lectures 2012

threat – attack – vulnerability – asset
(or incident)



risk

e.g.

**hardware problem with disks – disks with data X failed –
data X wrongly backed-up – availability of data X**



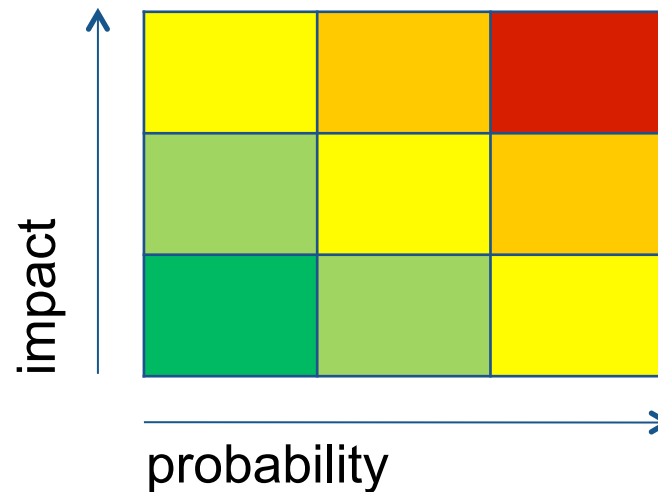
data X lost

This is just one
risk scenario
for that risk

Risk management

Computer.Security@cern.ch — Computer Security Lectures 2012

risk = probability * impact



► **assess them, prioritize, mitigate, avoid, and finally accept**

The right balance

Computer.Security@cern.ch — Computer Security Lectures 2012

How much security?

too little



vs.

too much



It's a **trade-off** between security, usability and cost

▶ Some golden rules

- ▶ deny by default
- ▶ defence in depth
- ▶ complex means insecure
- ▶ least privilege principle
- ▶ security, not obscurity

Deny by default

Computer.Security@cern.ch — Computer Security Lectures 2012

► Deny by default



Use **whitelisting** rather than blacklisting



Deny by default

Computer.Security@cern.ch — Computer Security Lectures 2012

```
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



Defense in depth

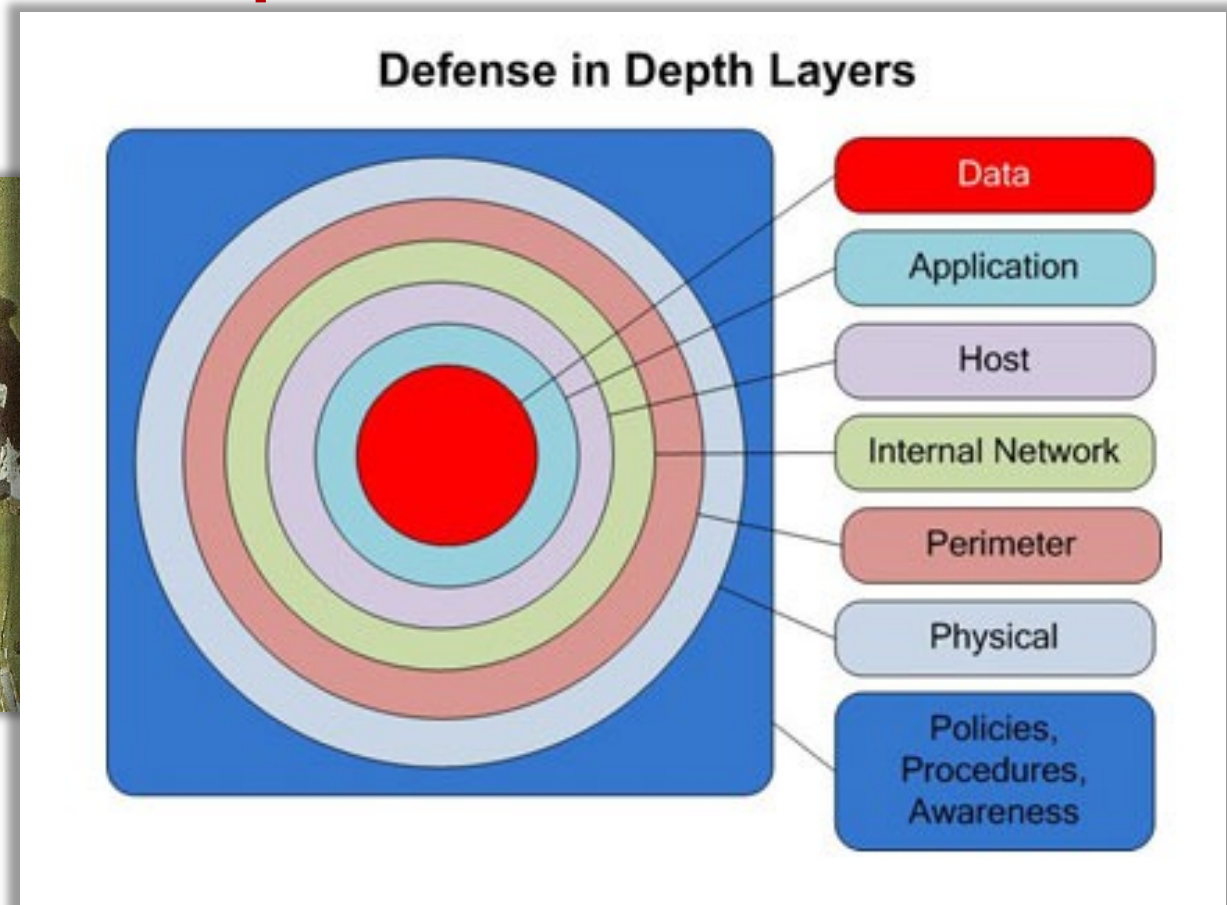
Computer.Security@cern.ch — Computer Security Lectures 2012

► Defense in depth

XXI century



XIII century



Use multiple layers of defense

Things to avoid

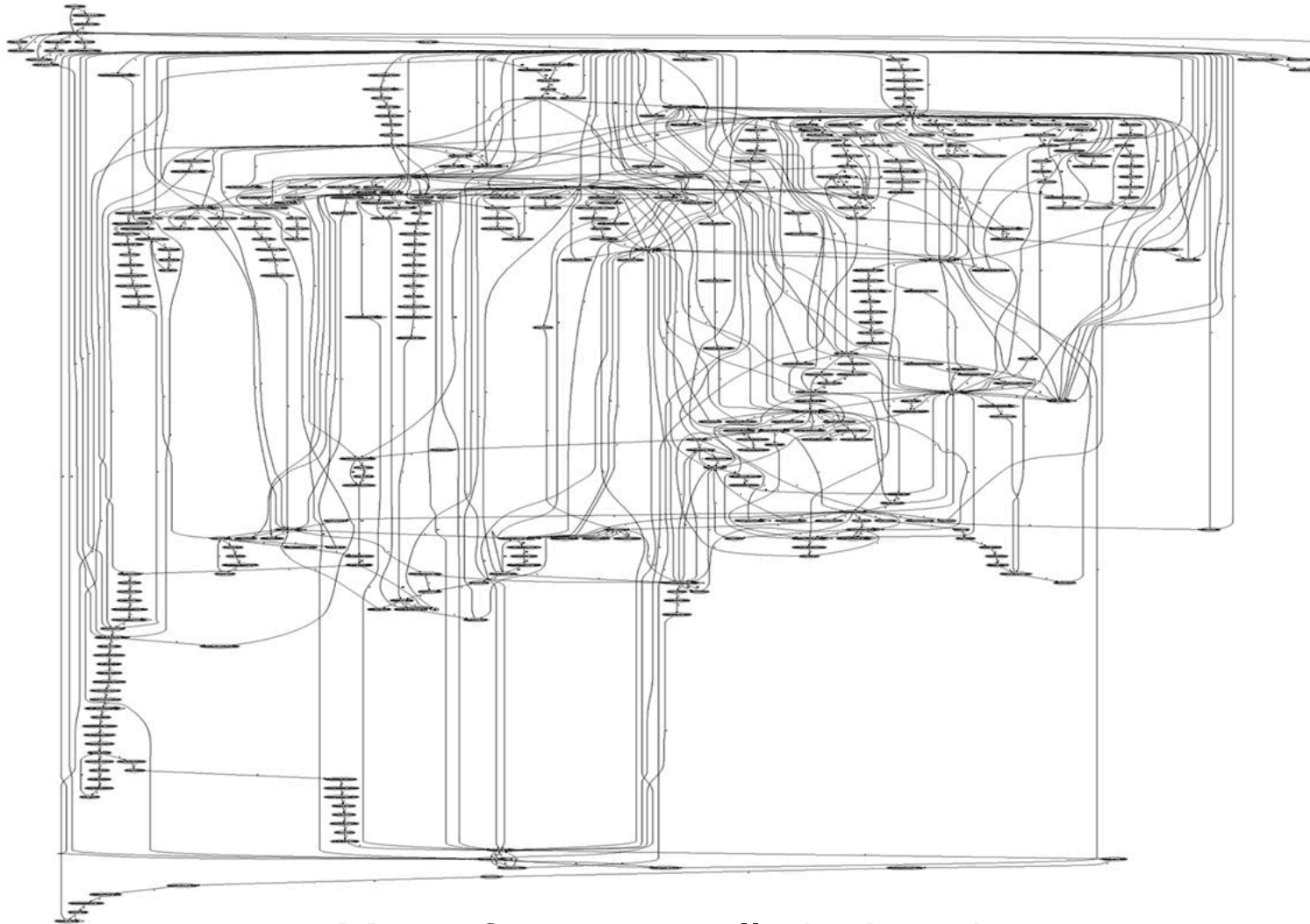
Computer.Security@cern.ch — Computer Security Lectures 2012

Situations that can
easily turn
disasterous



Complex means insecure

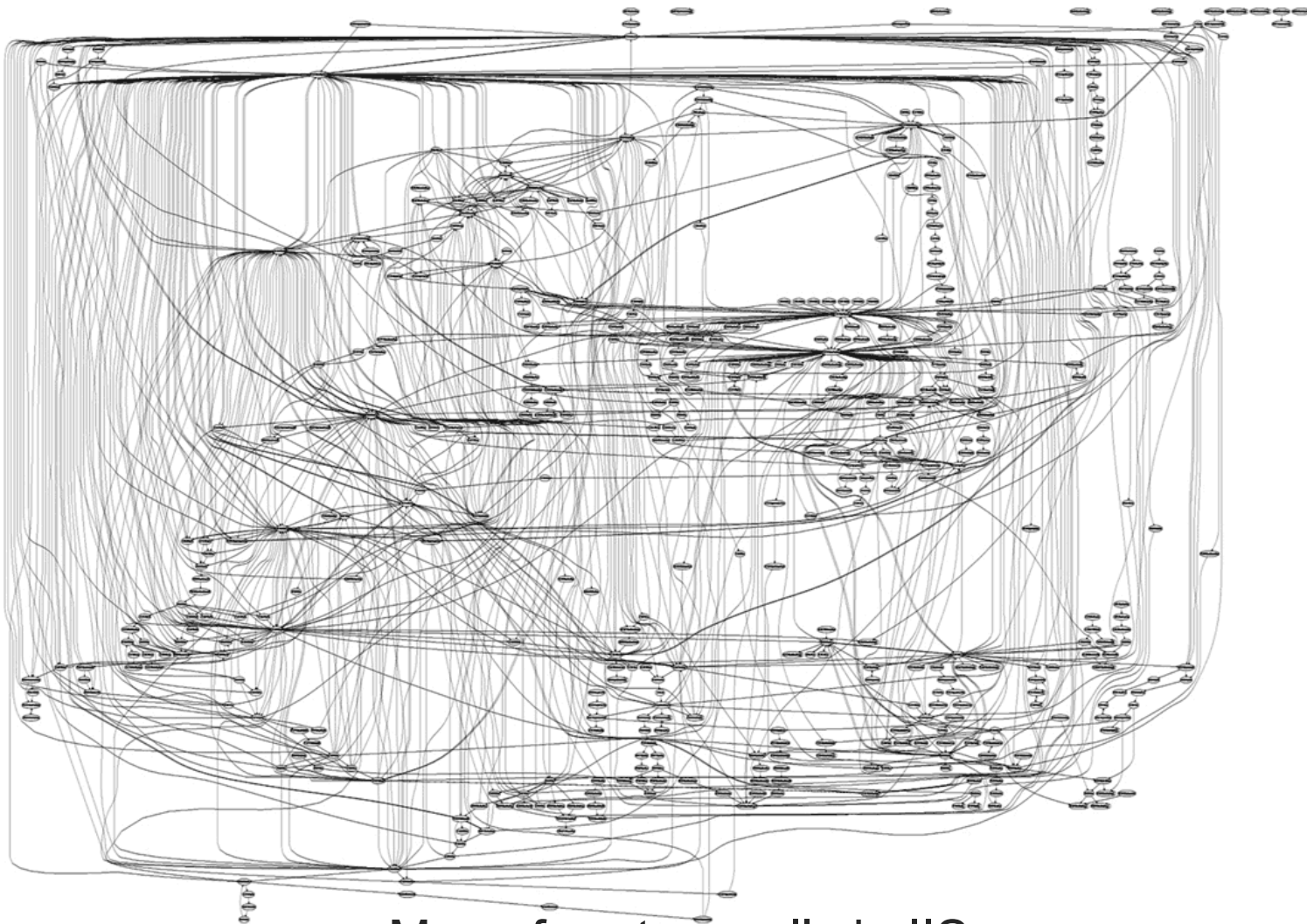
Computer.Security@cern.ch — Computer Security Lectures 2012



Map of system calls in Apache

Complex means insecure

Computer.Security@cern.ch — Computer Security Lectures 2012



Map of system calls in IIS

Least privilege principle

Computer.Security@cern.ch — Computer Security Lectures 2012

► Follow the **Least privilege** principle



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

Security, not obscurity

Computer.Security@cern.ch — Computer Security Lectures 2012

▶ *Security through obscurity?*

(hiding design or implementation details to gain security)



NO!

▶ Systems must be secure **by design**, not by obfuscation

▶ Security **AND** obscurity is OK



▶ A global image (an attempt)

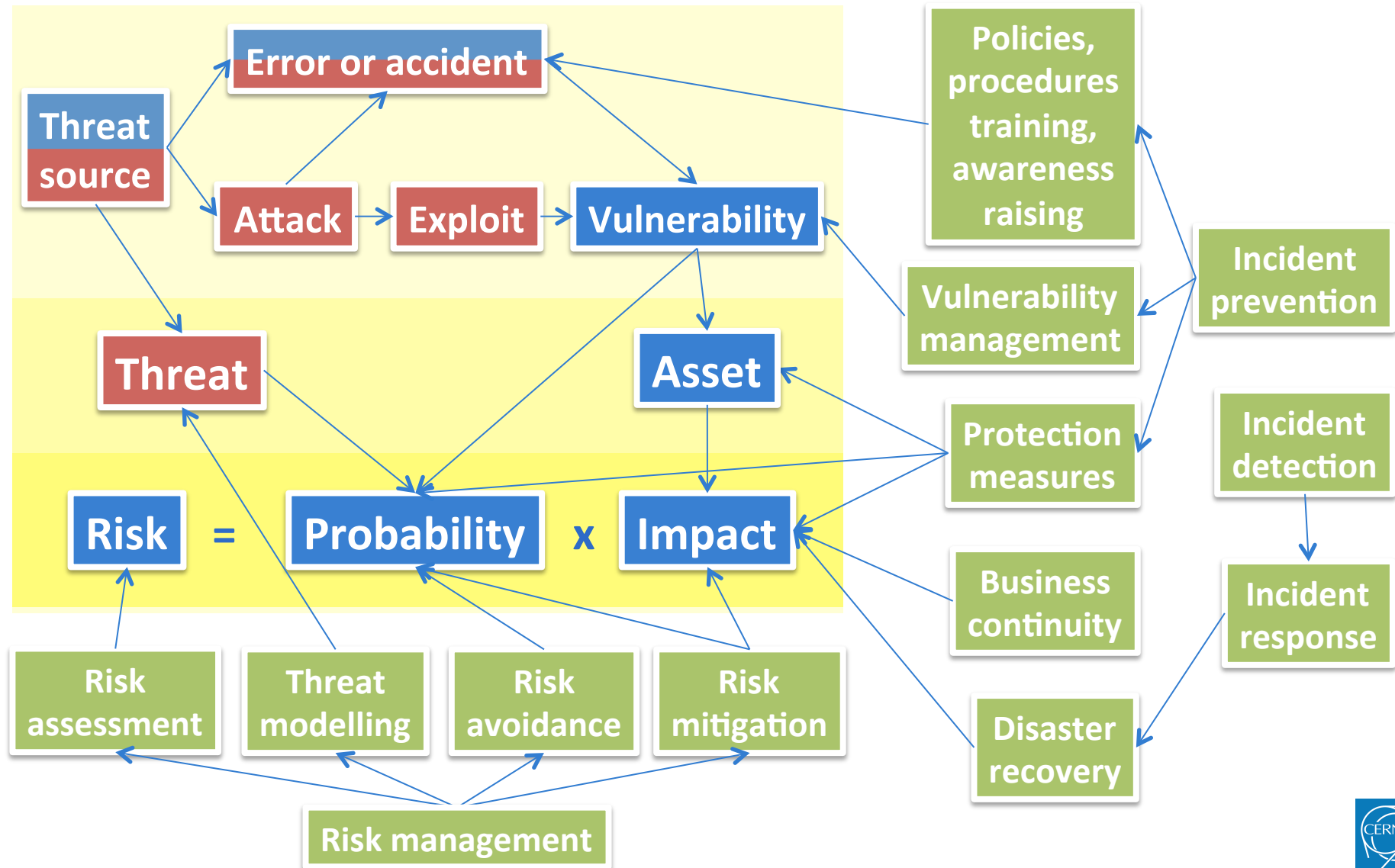
we may control

we cannot control

we do

Info security

Computer.Security@cern.ch — Computer Security Lectures 2012



© Sebastian Lopienski





► Summary

Summary

Computer.Security@cern.ch — Computer Security Lectures 2012

confidentiality – integrity – availability

technology – process – people

protection – detection – response

defense in depth deny by default

least privilege principle

complex = insecure security, not obscurity

risk = probability * impact

And get the balance right

Further reading

Computer.Security@cern.ch — Computer Security Lectures 2012

Bruce Schneier
*Secrets and Lies:
Digital Security
in a Networked World*



Let's learn some Chinese

Computer.Security@cern.ch — Computer Security Lectures 2012

人

person

囚

prisoner

(person in a box)

女

woman

安

secure

(woman under a roof)

Thank you

Computer.Security@cern.ch — Computer Security Lectures 2012



Any questions?

