香港城市大學
City University of Hong Kong

CS 6290

Privacy-enhancing Technologies

Department of Computer Science

Slides credit in part from courses: CSE484: Computer Security at UW, 6.5660: Computer Systems Security at MIT, CS155: Security Principles and OS Security.

Tutorial 1 – Threat Modelling

Yufei CHEN

CS Department
City University of Hong Kong

About Me 👋

- Yufei CHEN (陈宇飞)
- Postdoc from CS Dept
- Email: yufeichen8@cityu.edu.hk
- URL: https://yfchen1994.github.io

Goal of the Tutorial

- Help you better understand the key concepts from the lectures
- Provide practical insights and hands-on experience
- Create an open space where you can ask questions, share ideas, and explore topics in-depth

Format of the Tutorial

- Real-world case studies
- Lightweight programming
- Mathematical derivations

“Spirit of Openness”

- Openness
- Sharing
- Collaboration

Consider to contribute your feedbacks/questions/findings!

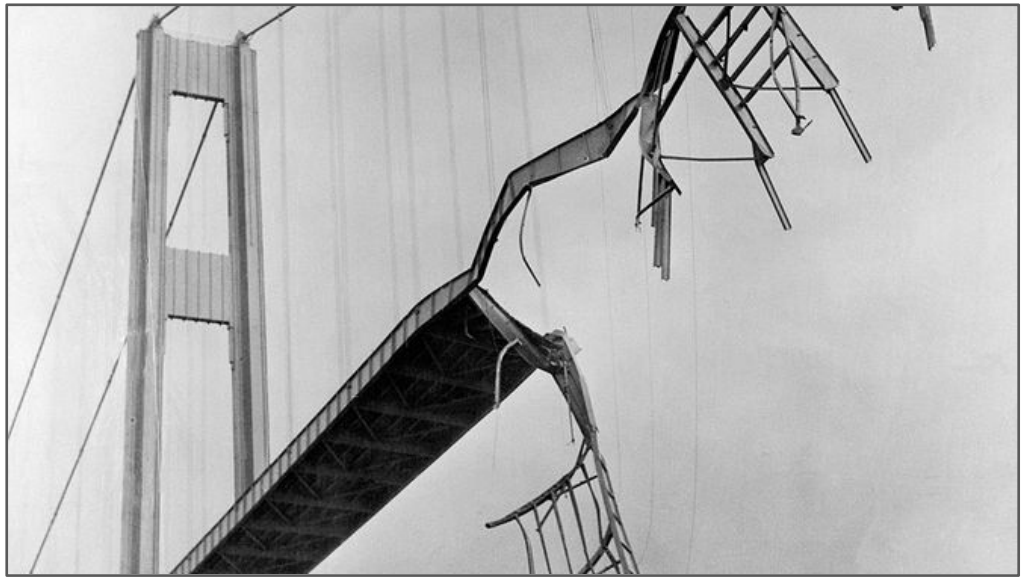
You can be the superhero!

What is threat modelling?

What is threat modelling?

- An approach for analyzing the security of a computer system
- Examine the potential vulnerabilities and risks of the system, and how attackers might approach it
  - *What are we protecting?*
  - *What does an attacker have to gain?*
  - *How would an attacker try to exploit the system?*

What's the difference?



Reliability does not equal **Security**

泊车场景更精准

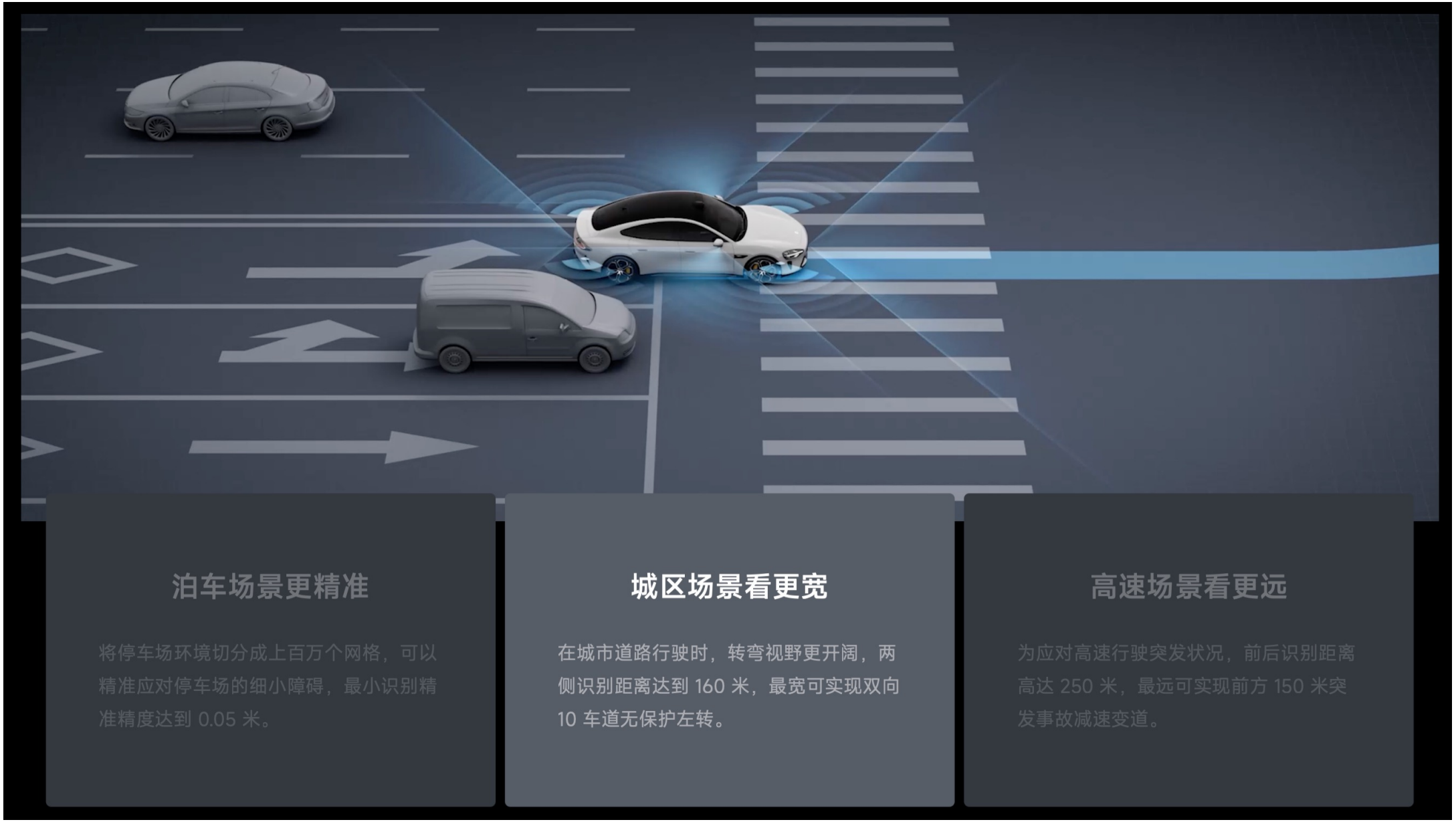
将停车场环境切分成上百万个网格，可以精准应对停车场的细小障碍，最小识别精准精度达到 0.05 米。

城区场景看更宽

在城市道路行驶时，转弯视野更开阔，两侧识别距离达到 160 米，最宽可实现双向 10 车道无保护左转。

高速场景看更远

为应对高速行驶突发状况，前后识别距离高达 250 米，最远可实现前方 150 米突发事故减速变道。


泊车场景更精准
将停车场环境切分成上百万个网格，可以精准应对停车场的细小障碍，最小识别精准精度达到 0.05 米。
城区场景看更宽
在城市道路行驶时，转弯视野更开阔，两侧识别距离达到 160 米，最宽可实现双向 10 车道无保护左转。
高速场景看更远
为应对高速行驶突发状况，前后识别距离高达 250 米，最远可实现前方 150 米突发事故减速变道。


泊车场景更精准
将停车场环境切分成上百万个网格，可以精准应对停车场的细小障碍，最小识别精准精度达到 0.05 米。
城区场景看更宽
在城市道路行驶时，转弯视野更开阔，两侧识别距离达到 160 米，最宽可实现双向 10 车道无保护左转。
高速场景看更远
为应对高速行驶突发状况，前后识别距离高达 250 米，最远可实现前方 150 米突发事故减速变道。

Meet the Adversary

"Computer security studies how systems behave in the presence of an **adversary**."

- The adversary…
- a.k.a. the attacker
- a.k.a. the bad guy
- An intelligence that actively tries to cause the system to misbehave.

Trapping a car with a bottle of salt.

J. Bridle. https://jamesbridle.com/works/autonomous-trap-001

Trapping a car with a bottle of salt.

J. Bridle. https://jamesbridle.com/works/autonomous-trap-001

Know thine Enemy

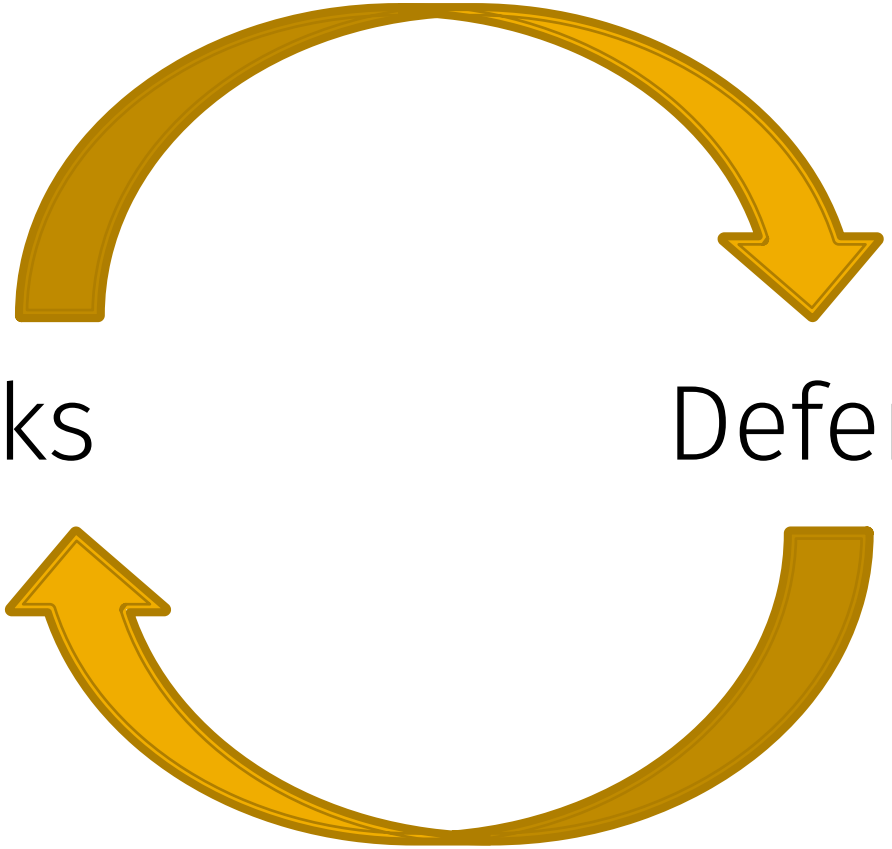
- Motives?
  - Disruption
  - Espionage
  - Money
- Capabilities?
  - Denial of service
  - Code execution
- Degree of access?
  - Physical access
  - Root privileges

The Security Mindset

- Thinking like a defender
  - Know what you're defending, and against whom
  - Weigh benefits vs. costs:
    No system is ever completely secure.
  - Embrace "rational paranoia"
- Thinking like an attacker
  - Understand techniques for circumventing security
  - Look for ways security can break,
    not reasons why it won't

High-level Approaches

Attacks



Defenses

Thinking like an Attacker

- Look for the weakest links
  - What is easiest to attack
- Identify assumptions that the security depends on
  - Are any assumptions false?
  - Can you render them false?
- **Think outside the box!**
  - Don't be constrained by the system designer's worldview

Practice thinking like an attacker:

For every system you interact with, think about what it means for it to be secure, and **imagine how it could be exploited**

Thinking as a Defender

- Security policy
  - What are we trying to protect?
  - What properties are we trying to enforce?
- Threat model
  - Who are the attackers? Capabilities? Motivations?
  - What kind of attack are we trying to prevent?
- Risk assessment
  - What are the weaknesses of the system?
  - What will successful attacks cost us?
- How likely?
  - Countermeasures
  - Costs vs. benefits?
  - Technical vs. nontechnical?

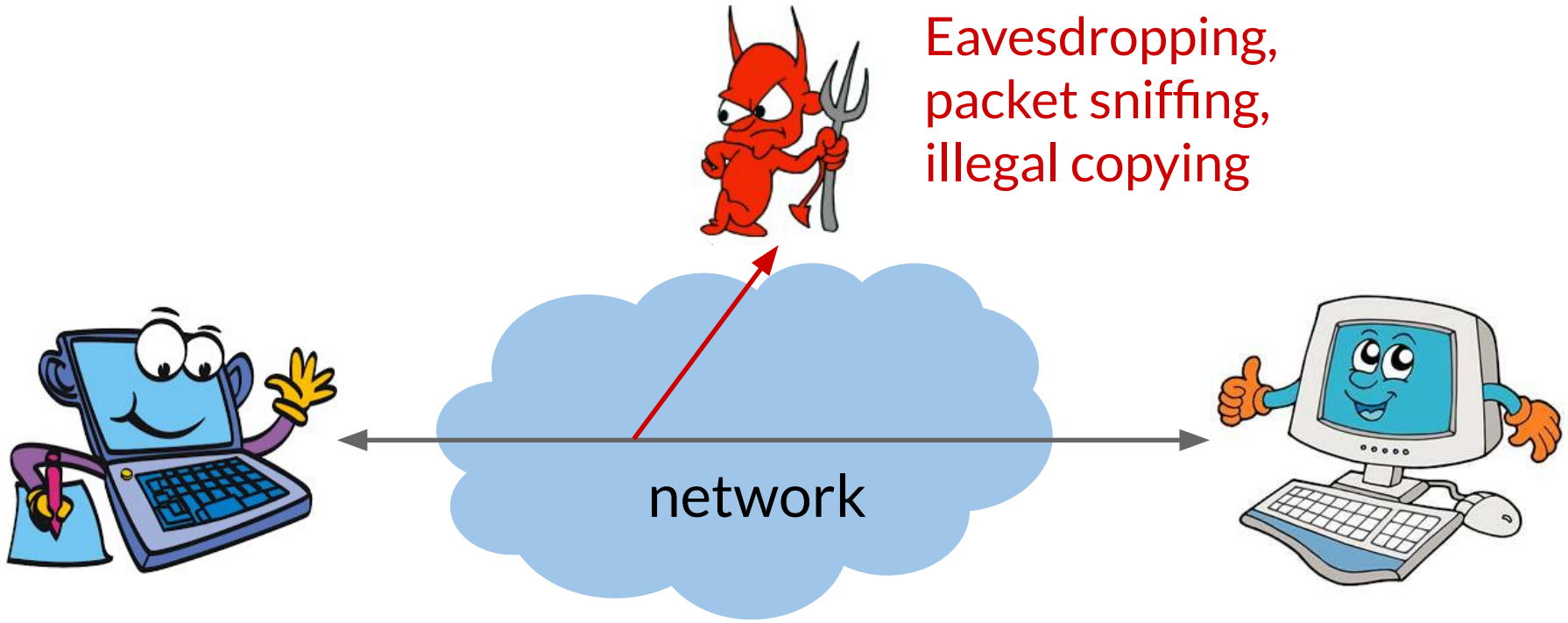
The challenge is to think rationally and rigorously about risk.

Rational paranoia.

Security goals to keep in mind…

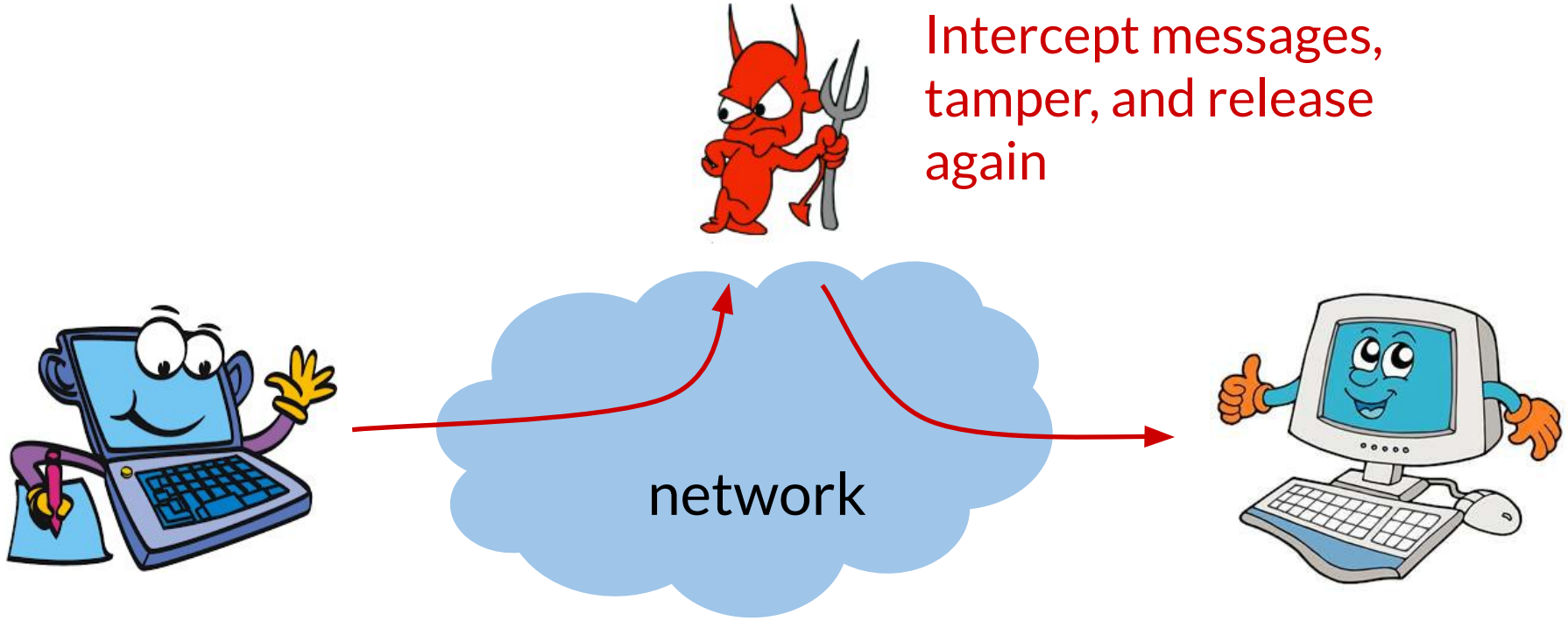
Confidentiality

Confidentiality is the *concealment of information*



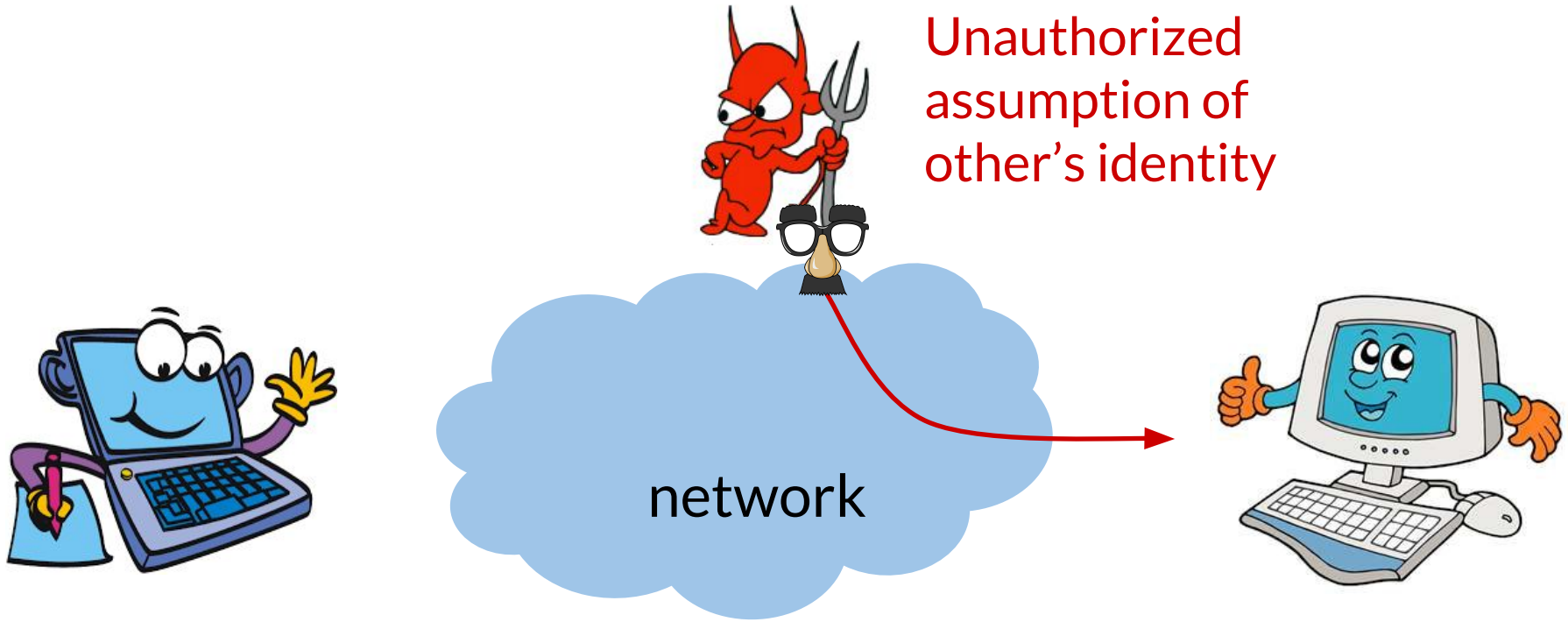
Integrity

Integrity is the *prevention of unauthorized changes*



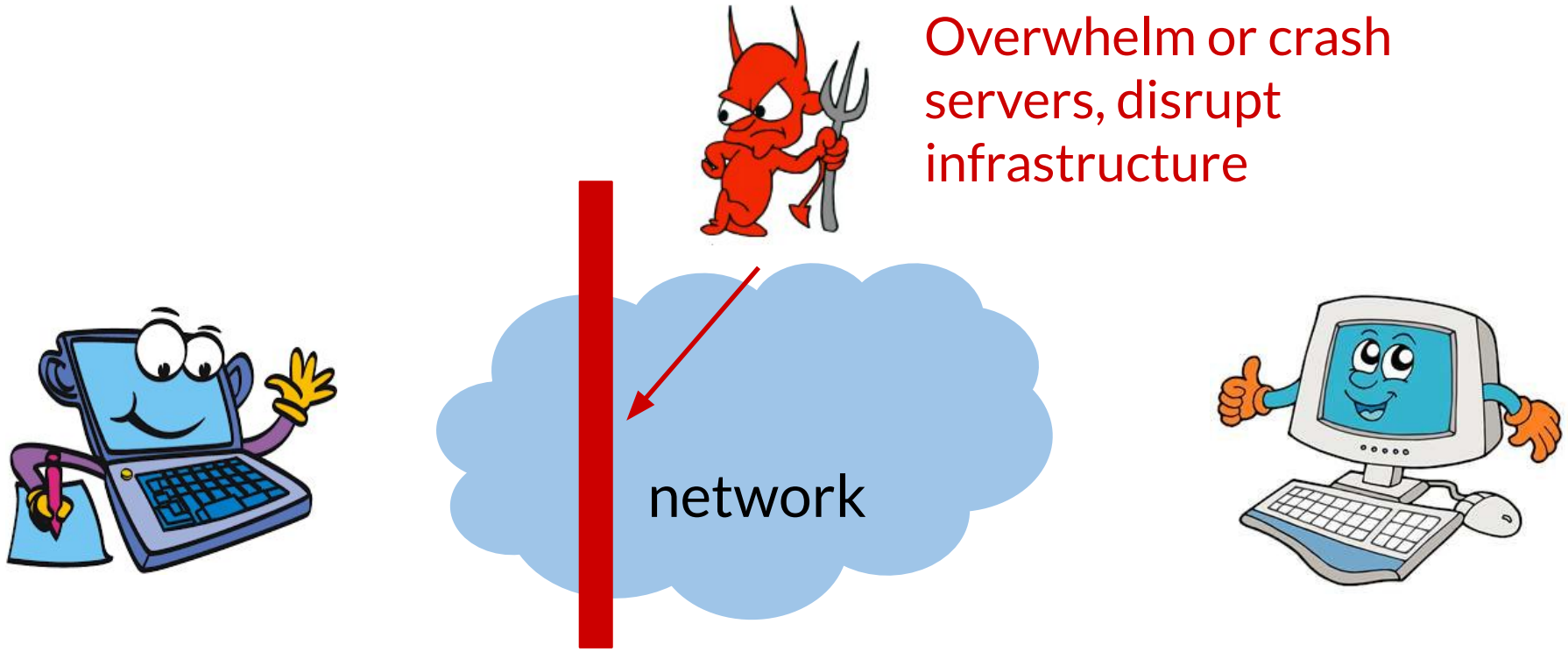
Authenticity

Authenticity is *knowing who you're talking to*



Availability

Availability is the *ability to use information or resources*



Threat Models

- Who are our adversaries?
  - Motives?
  - Capabilities?
  - Level of access?
- What kinds of attacks must we prevent?
  - Think like the attacker!
- Limits: kinds of attacks we should ignore?
  - Unrealistic versus unlikely

Security via Obscurity?

- Many organization think their data are private because they perturb the data and make the parameters of perturbation secret.

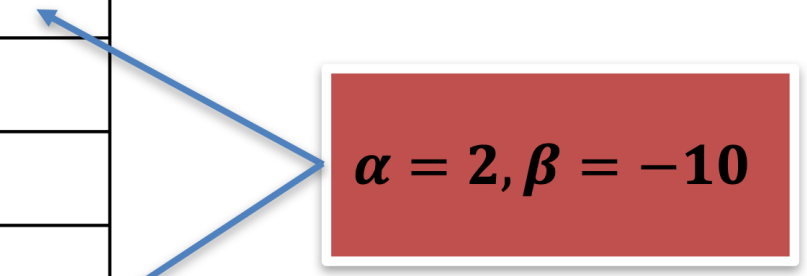
Icebreaker (~5 min)

- Say hello to your surrounding classmates:
  - Your name, major, previous experiences with computer security and privacy?
- Discuss on this case with your (new) friends!

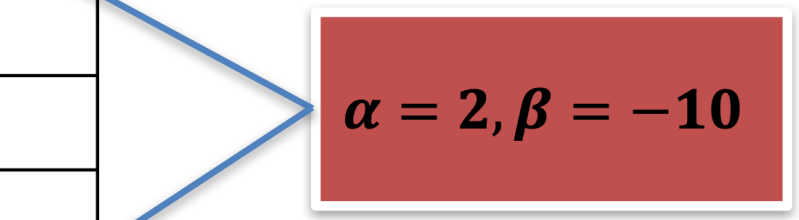
- The email service provider also released perturbed records as per a linear function, but with secret parameters. What can Alice and Cathy deduce now?

Node ID	Age (perturbed)	True Age
1 (Alice)	40	25
2 (Ed)	34	
3 (Bob)	52	
4	28	
5 (Cathy)	48	29
6	22	
7	92	

Node ID	Name	Age ($\alpha x + \beta$)	True Age
1	Alice	40	25
2	Ed	34	
3	Bob	52	
4		28	
5	Cathy	48	29
6		22	
7		92	



Node ID	Name	Age ($\alpha x + \beta$)	True Age
1	Alice	40	25
2	Ed	34	22
3	Bob	52	31
4		28	19
5	Cathy	48	29
6		22	16
7		92	51



Kerckhoff's Principle

"a crypto system should be secure even if everything about the system, except the key, is public knowledge."

- Auguste Kerckhoff

Open Design

“The security of a mechanism should not depend on the secrecy of its design or implementation.”

If the details of the mechanism leaks (through reverse engineering, dumpster diving or social engineering), then it is a catastrophic failure for all the users at once.

If the secrets are abstracted from the mechanism, e.g., inside a key, then leakage of a key only affects one user.

Security/Privacy has costs!

Which one you need?

天猫 年货节 新品升级虎牌保险柜
家用小型45/60/70CM指纹保...

保险箱热卖榜·第1名

¥188.00

全网低价 超级单品 跨店每200减30

回头客5万

德国司库进口保险柜箱家用防火盗
爆奢华转表器大型双门非宾利集宝

单门 全国联保

¥2588000.00

包邮 公益宝贝

Weak Passwords

- RockYou hack
  - "Social gaming" company
  - Database with 32 million user passwords from partner social networks
  - Passwords stored in the clear
  - December 2009: entire database hacked using an SQL injection attack and posted on the Internet
  - One of many such examples!

Weak Passwords

Password Popularity – Top 20

Rank	Password	Number of Users with Password (absolute)
1	123456	290731
2	12345	79078
3	123456789	76790
4	Password	61958
5	iloveyou	51622
6	princess	35231
7	rockyou	22588
8	1234567	21726
9	12345678	20553
10	abc123	17542

Rank	Password	Number of Users with Password (absolute)
11	Nicole	17168
12	Daniel	16409
13	babygirl	16094
14	monkey	15294
15	Jessica	15162
16	Lovely	14950
17	michael	14898
18	Ashley	14329
19	654321	13984
20	Qwerty	13856

[Inglesant and Sasse, "The True Cost of Unusable Password Policies"]

Password Policies

- Old recommendation:
  - 7 or 8 characters, at least 3 out of {digits, upper-case, lower-case, non-alphanumeric}, no dictionary words, change every 4 months, password may not be similar to previous 12 passwords…

Image from http://www.interactivetools.com/staff/dave/damons_office/

[Inglesant and Sasse, "The True Cost of Unusable Password Policies"]

Password Policies

- Old recommendation:
  - 7 or 8 characters, at least 3 out of {digits, upper-case, lower-case, non-alphanumeric}, no dictionary words, change every 4 months, password may not be similar to previous 12 passwords…
- **But …** results in frustrated users and less security
  - Burdens of devising, learning, forgetting passwords
  - Users construct passwords insecurely, write them down
    - Can't use their favorite password construction techniques (small changes to old passwords, etc.)
  - Heavy password re-use across systems
  - **(Password managers can help)**

Costs of Security

- **No security mechanism is free**
- Direct costs:
  - Design, implementation, enforcement, false positives
- Indirect costs:
  - Lost productivity, added complexity, time to market
- Challenge is to rationally weigh costs vs. risk
  - Human psychology makes reasoning about high cost, low probability events very difficult

Assessing Risk

- Remember: *Rational* paranoia

- What would security breaches cost us?
  - Direct: money, intellectual property, safety
  - Indirect: reputation, future business, well being

- How likely are these costs?
  - Probability of attacks?
  - Probability of success?

Countermeasures

- Technical countermeasures
  - Bug fixes, more crypto, re-architecting, etc.
- Nontechnical countermeasures
  - Law, policy (government, institutional)
  - Procedures, training, auditing, incentives, etc.

Where to Focus Defenses

- Trusted components (aka Trusted Computing Base)
  - Parts that must function correctly for the system to be secure.
- Attack surface
  - Parts of the system exposed to the attacker
- **Complexity versus security** are inversely related

Other Principles

- Defense-in-Depth
  - Multiple layers of safeguards
  - Physical, technical, administrative
- Diversity
  - More moving parts = harder to attack
  - Conversely, harder to secure
- Maintainability
  - Minimize maintainer workload
  - Make fixes easy/fast to deploy