50.020 Security
Lecture 1: Introduction

About 50.020 – Admin

- Instructors: Zhaohui TANG, Andrei Bytes (TA)
- Part of ISTD Security and Communications track
- Pre-requisite: Computer System Engineering, Networks
- 2 Lectures + 1 lab/ week. 12 credits total.
- Grade: 10% in-class participation, 40% labs (including a 3-week team project), 20% midterm exam, 30% final exam
- Main contents:
    - Cryptography (Public/Privat key, Hashing, Analysis)
    - Protocols (Key establishment, Secure Channels)
    - System Security (Buffer overflows, web security)
    - Network Security (Transport Layer Security)

About 50.020 – Format

- Most exercise grading and in-class tests using eDimension.
- Everyone should be signed up.
- All important communication through eDimension.
- Please always backup your exercise submissions in case of any system issue at eDimension
- Lecture slides may have two versions: student and instructor. Difference: no solution in student version for some questions/in-class activities.

Important notes on labs

1. Two lab sessions concurrently: LEET (1.612) + Think Tank 11 Room 1.503 (if you are willing to work with your own laptop).
2. Most of the labs can be done with your own laptop. Please install: Lubuntu 16.04 (all labs work on Linux), Python 3.6.
3. A separate session (with different timing) will be provided if needed. For this, we need to find a timing available for students, instructors and LEET lab.

Important notes on labs

1. Some labs can ONLY be done within an isolated network environment. For these labs, you are strictly prohibited from experimenting within SUTD network or any other public network. Please read lab instructions carefully. For ethics related matters, please refer to "Disclaim" and "Ethics" pages (to mention soon).
2. For every lab, there is 10 mins' briefing (1-1:10pm) in LEET lab. You can shift to Room 1.503 after briefing.
3. All labs are graded.
4. (Heads-up) For the 3-week project CTF (Capture The Flag), you many want to utilize free resources from AWS. Check your student AWS account.
5. Avoid using LEET lab on the following days/time:
   Tuesday : 6:00pm - 10:00pm; Saturday: 10:00am - 2:00pm

Content of the class

Learning Objectives

1. List basic security solutions and models; explain concepts for confidentiality, integrity and availability.
2. Classify and describe common attacks and countermeasures for host, network and web security.
3. Apply known attacks to vulnerable cryptographic primitives.
4. Model, analyze, and apply cryptographic primitives used for encryption, secure hashing, and digital signatures.
5. Design security solutions to achieve specific security goals in a system.
6. Apply protocols used for key establishment, network encryption, and authentication to secure a system.
7. Evaluate the security of existing networked systems.

Class Calendar

Refer to classCalendar.pdf in eDimension.

Example for exercise content

1. Ciphers: write your own, attack others
2. Rainbow table attacks and countermeasures
3. Security assessment of web servers
4. SQL injection attacks
5. Return-to-libc attacks
6. . . .

Further reading

- If you are looking for further reading, consider these:
  - Understanding Cryptography: A Textbook for Students and Practitioners, C. Paar and J. Petzel, `http://www.crypto-textbook.com/`
  - Security Engineering, R. Anderson, `https://www.cl.cam.ac.uk/~rja14/book.html`
  - If you are looking for a fun book on crypto, try The code book : the science of secrecy from ancient Egypt to quantum cryptography, S. Singh

Before we start – Disclaimer

This class will NOT:

- Teach you how to "hack any system"
- Teach you how to break any encryption
- Teach you how to make your system 100% secure

All three things are impossible

- Also: "Don't try this at home"
  - Don't use tools and knowledge from this class to attack third parties!
  - Considered a serious offense. It's easy to detect (beginners)
  - Don't attack SUTD network or lab

What we will do

- Learn about common security problems, and how to find them
- Learn why these are common, how to fix them
- How to design systems which are 99% secure[1]
- Techniques are not restricted to computer systems, can be applied in many scenarios

[1] your mileage may vary

Ethics - Origin of hacking

- The term *hacking* refers to creative use of computer systems
    - Coined in academia (MIT) in the 60s
    - Not necessarily security related
- In security, different hacking flavors exist:
    - White-hat (benign), grey-hat, black-hat (malicious)
- In this lecture: attacker vs. defender
    - We don't discuss politics or morals

Ethics - Responsible hacking

What to do if you identify security problems?

- On your new iPhone, you find a way to bypass unlocking screen
- You find an easy way to bypass MS Office password protection
- What are possible paths of action?

Ethics - Responsible hacking

What to do if you identify security problems?

- On your new iPhone, you find a way to bypass unlocking screen
- You find an easy way to bypass MS Office password protection
- What are possible paths of action?

- Options:
  1. Full disclosure: Notify the internet/media immediately
  2. No disclosure: Notify the vendor, but no-one else
  3. Sell the vulnerability on the (black) market

Which one should you take?

Ethics - Responsible Disclosure

- Security professionals often choose a fourth option: *responsible disclosure*
- Notify producer of vulnerability, but provide a deadline for patches
    - After deadline passes, go to the public
- This gives producer time to react, but forces him to action
- Nowadays, some companies also start to pay for vulnerabilities (Google/Facebook)

What is security?

Security? Where do we need it?

- On the Internet
- In your phone
- In your car
- In the MRT, ferry, airplane
- In the hospital, power grid, ATM
- With the *Internet of Things*, everywhere!

"The [yearly] cost of cybercrime and cyber espionage to the global economy is [..] hundreds of billions of dollars." – McAfee

What is security?

Origin of the word

"Security" derived from latin "securus", meaning "free from care"

[2] In practice: there is at most a negligible chance of success

What is security?

Origin of the word

"Security" derived from latin "securus", meaning "free from care"

So. . .

⇒ I am secure if I don't care?

[2] In practice: there is at most a negligible chance of success

What is security?

Origin of the word

"Security" derived from latin "securus", meaning "free from care"

So...

⇒ I am secure if I don't care?

Correct interpretation:

- I am secure if I know there cannot[2] be any successful attack

[2] In practice: there is at most a negligible chance of success

Basic terminology: Properties

"Classic" C,I,A properties:

- C onfidentiality
  - Attacker cannot obtain secret data of victim
- I ntegrity
  - Attacker cannot change data of victim undetected
- A vailability
  - Attacker cannot stop services provided by victim (Denial of Service/DoS)

Additional properties

- Non-repudiation
  - Attacker cannot deny having taken certain actions
- Privacy
  - An attacker cannot learn *private* information of victim
- Authenticity, . . .

Measuring security

How can security be specified?

- Attribute of a system
  - But depending on the attacker?
- Boolean value (secure/insecure) or real number?
  - What about probabilistic attacks?
- Metrics for security?
  - "Your system is 78% secure", "Your security is level 7"

Measuring security in practice

- Metrics such as number of bugs found in software
- Attack surface metrics: *how many entry points?*
- Practical time-to-compromise for experts
- In general: estimates based on complexity and cost

Effort/time estimates based on brute force key exploration:

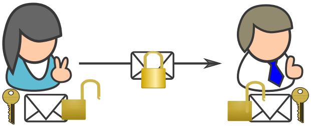
Key length	Attempts	Time to brute force attack
32	2^{32}	Realtime
64	2^{64}	Few days or less
128	2^{128}	Decades
256	2^{256}	Long term secure

Numbers for symmetric keys. See also:
`https://www.keylength.com/en/3/`

Basic terminology: Alice, Bob, and Eve

- Who are they?
  - Commonly used in security research to explain protocol interactions
  - Names sometimes change (Mallory, Charly, etc)
  - Just a convenient way to identify parties (e.g., servers, users)
  - Alice usually initiates communication
- Part of our fundamental attacker and system model (more later)

Basic terminology: Cryptography



- Alice wants to send a secret *message m* to Bob
- The original message m is the *plaintext*
- Alice has shared *key* k and symmetric encryption function E(m,k)
- Alice encrypts the plaintext to obtain a *ciphertext* c=E(m,k)
- Bob receives the ciphertext, and applies key and D(c,k) to *decrypt*, resulting in the plaintext m=D(c,k)

System and attacker model, requirements

A system can only be secure wrt well-defined assumptions/models

- A *system model* that describes the involved legitimate parties, their actions and behaviour
- An *attacker model* that provides an exhaustive description of the attacker
- A list of requirements for the operation of the system, and the security requirements

Kerckhoffs' design principles (1883)

Kerckhoff wrote on military ciphers:

1. The system must be practically, if not mathematically, indecipherable;
2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
3. Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
4. It must be applicable to telegraphic correspondence;

(2 more principles omitted here)

Takeaway from Kerckhoffs

- Implicit description of the user (system)
  - User is applying cipher by hand, must remember key, transmits by telegraph
- Implicit attacker model
  - Attacker's goal: learn the plaintext
  - Attacker is able to use math for cryptanalysis
  - Attacker can learn about the algorithm
- Implicit requirements for the system
  - must be usable and memorable
  - enemy must not be able to derive plaintext

3D printing and Security

- Here at SUTD we do lots of 3D printing
- Have you considered *security* impact of 3D-printing?

Physical keys

System model:

- Locks protect doors or start devices
- Physical possession of key grants access

Attacker model:

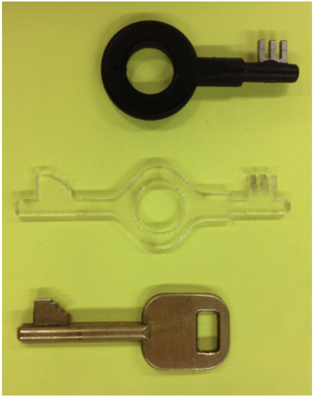
- Does not have legitimate key, but access to lock
- Wants to open lock (or start engine)

Security requirements:

- With legitimate key, lock can always be opened
- Without key, lock cannot be opened

Assumption: keys are impossible to replicate without original

3D-copied keys



- 3D printed copies of security keys
  - For security doors
  - Universal keys for handcuffs
  - Keys for infrastructure (voting machines)

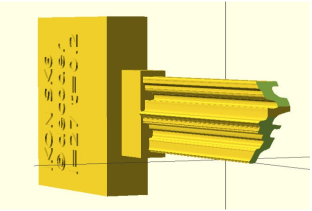
How to get data for the keys?

- Often it's easy: pictures will be posted by manufacturer
- These keys are for American Diebold voting machines!
- Based on pictures, correct key share can be derived
- Voting machines can now be manipulated (e.g. "updated")

But security keys are. . . secure?

- Simple keys were always easy to copy
- 3D printing also allows easy replication of ”security” keys
- MIT students released a tool to generate CAD models

But getting a picture of the key is hard?



- For some keys, a picture of the *lock* is enough
- The displayed key can be used to *bump* security locks
- *Key impressioning* is another technique to create key copy

Conclusion

- Security is important and will become ubiquitous
- Often hard to define and measure
- Depends on System model, Security Requirements and Attacker model
- Lock-picking is a physical example of security attacks and defenses

Interesting videos about security:

- DefCon conference (YouTube)
- Blackhat conference (YouTube)
- Lock-picking (on YouTube)