Computer Security Capstone

Project 1:
TLS Connection Hijacking

Chi-Yu Li   (2025 Spring)

Computer Science Department

National Yang Ming Chiao Tung University

Goal

● Understand how to hijack a TLS connection

● You will learn about

  ❑ Establish TLS connections with customized certificates

  ❑ Handle multiple network connections

  ❑ Importance of certificates and identity verification

What is HTTPS?

● Nowadays, HTTPS (HyperText Transfer Protocol Secure) is commonly used to secure HTTP connections between end devices and web servers

● In HTTPS, the communication is encrypted using the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) convention

What is TLS?

- **TLS is the successor to SSL**
  - ❑ It is a security protocol that provides privacy and data integrity for Internet communications

- **Key Features**
  - ❑ Encryption: Protects data transmitted over the network from eavesdropping.
  - ❑ Authentication: Uses digital certificates to verify the identity of parties.
  - ❑ Data Integrity: Ensures that data has not been altered during transmission

TLS Primer: Certificate and CA

● TLS certificates are crucial for establishing secure connections

  ❑ Containing public keys, identity information, and digital signature

  ❑ Facilitating encryption, authentication, and data integrity

● A certificate authority (CA) is a trusted entity that issues certificates

  ❑ Ensuring the authenticity of websites, domains and organizations

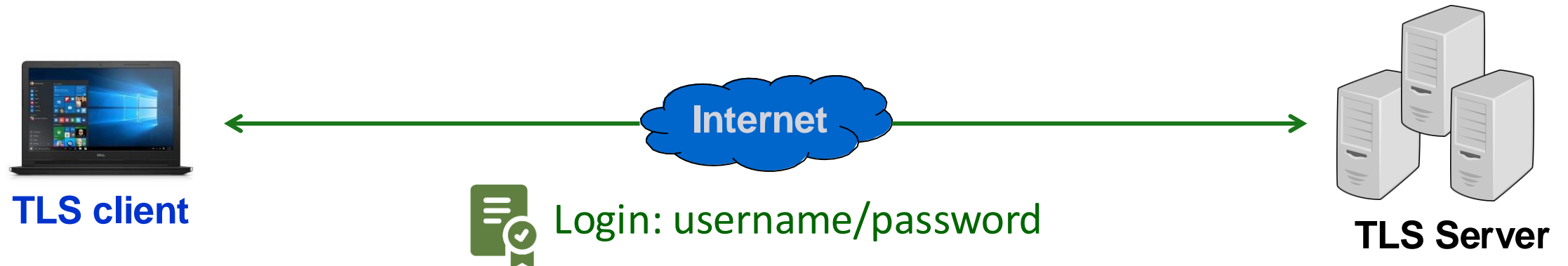 ❑ Help users verify they are connected to an official website, preventing fake or spoofed sites created by attackers

TLS Primer: Cipher Suite

● **Cipher Suites are predefined sets of algorithms that dictate how TLS protects data**

● **Components of a Cipher Suite**

  ❑ Key Exchange Algorithm
  - Securely exchanging cryptographic keys between a client and a server

  ❑ Encryption Algorithm
  - Encrypting the data being transmitted

  ❑ Hashing Algorithm
  - Ensuring the integrity and authenticity of the message
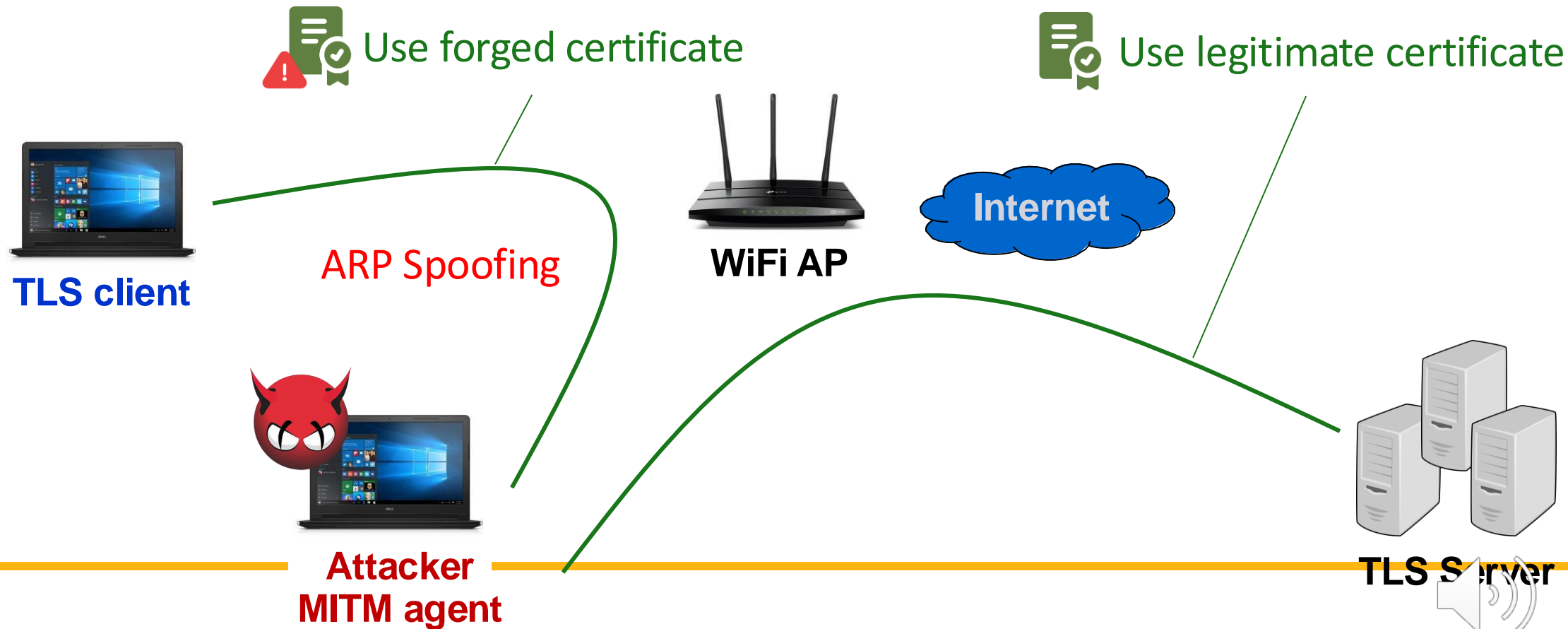
  ❑ E.g. TLS_RSA_WITH_AES_128_GCM_SHA256

Normal TLS connection

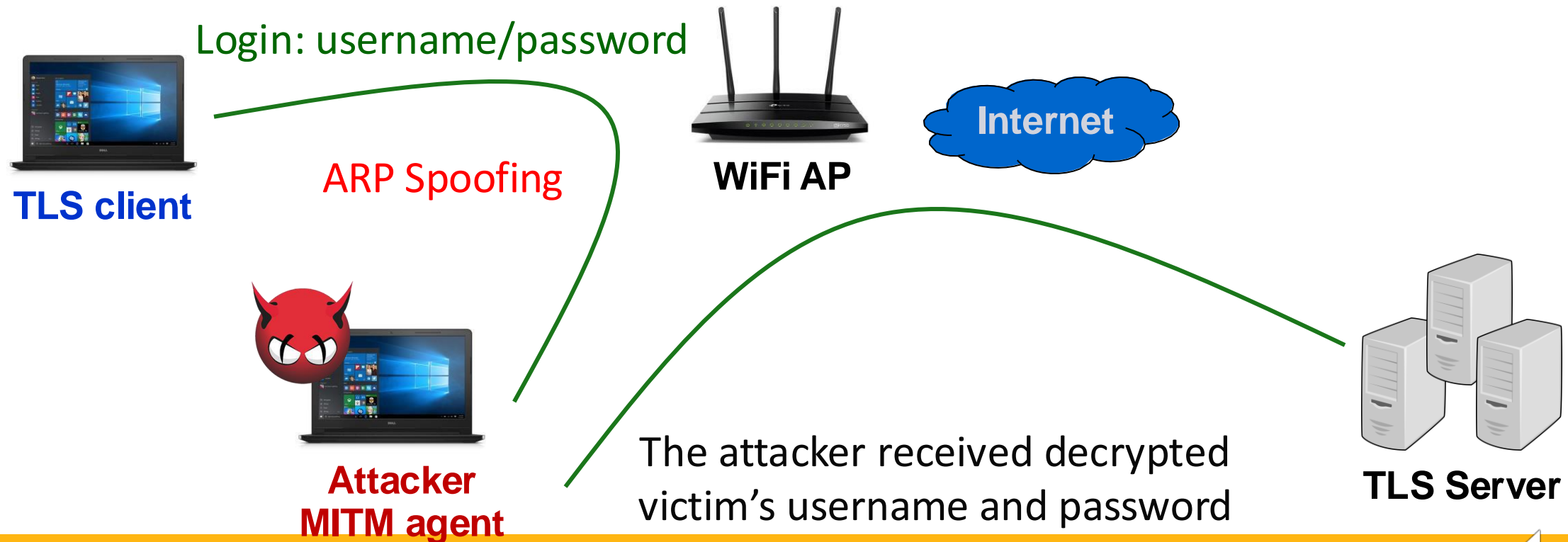● Establish a secure connection with a legitimate certificate

TLS client   Login: username/password   **TLS Server**

Attack Scenario

● How can Attacker steal Victim's user credential?

Use forged certificate

Use legitimate certificate

TLS client

ARP Spoofing

WiFi AP

Internet

**Attacker
MITM agent**

TLS Server

Attack Scenario

● How can Attacker steal Victim's user credential?

Login: username/password

TLS client

ARP Spoofing

WiFi AP

Internet

Attacker MITM agent

The attacker received decrypted victim's username and password

TLS Server

Major Ideas

● **Redirect Victim's traffic to Attacker**

    ❑ Man-in-the-middle based on ARP spoofing

● **Dual Connection Establishment**

    ❑ What you need to implement in this project

TLS Client   forged certificate   **Attacker MITM Agent**   legitimate certificate   **TLS Server**
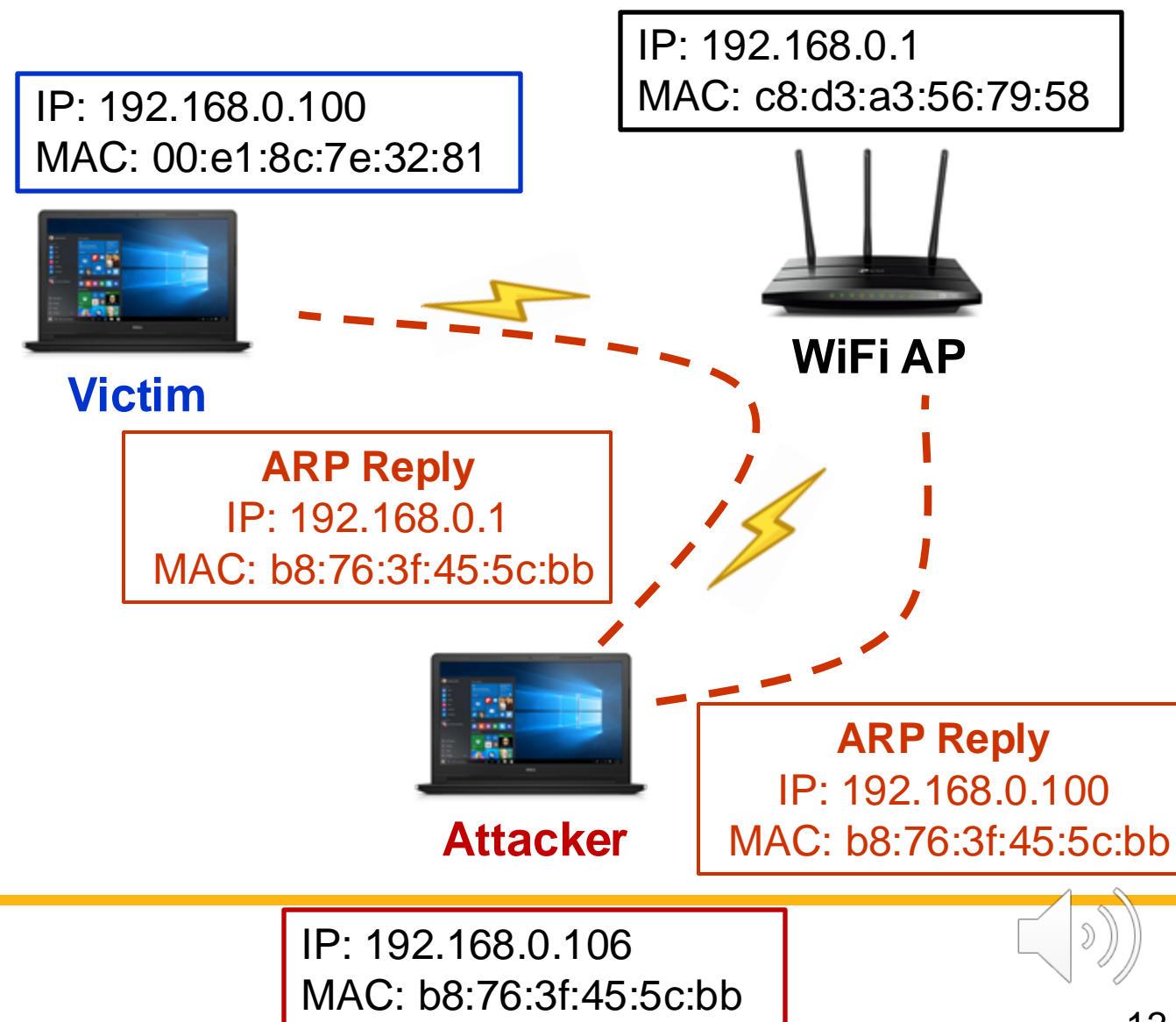
What is ARP (Address Resolution Protocol)?

● A communication protocol: discovering the link layer (or MAC) address associated with a given IP

● A request-response protocol: messages are encapsulated by a link-layer protocol

☐ ARP request: broadcast

☐ ARP response: unicast

● Never routed across internetworking nodes

What is ARP Spoofing?

- Generate spoofed ARP replies for all other client devices
  - ☐ Hint: ARP format and thread

- Both uplink and downlink should be considered
  - ☐ Other client devices' network services can work normally

IP: 192.168.0.1
MAC: c8:d3:a3:56:79:58

IP: 192.168.0.100
MAC: 00:e1:8c:7e:32:81

Victim

WiFi AP

ARP Reply
IP: 192.168.0.1
MAC: b8:76:3f:45:5c:bb

ARP Reply
IP: 192.168.0.100
MAC: b8:76:3f:45:5c:bb

Attacker

IP: 192.168.0.106
MAC: b8:76:3f:45:5c:bb

Experimental Setting

- The attacker VM executes the command below to redirect specific TLS packets to the MITM agent:

  ☐ `sudo ./setup.sh`

- The victim VM should start the browser using the following command to establish a TLS connection with a forged certificate:

  ☐ `google-chrome --ignore-certificate-errors --user-data-dir=/tmp/chrome_dev`
  - In real-life situations, such as IoT environments, where certificates are often not verified or when a certificate is injected into the browser, this type of attack can be launched

  ☐ Recommend to open the browser in Incognito mode.

Experimental Setting: ARP Spoofing

● Attacker VM executes the command below in the MITM agent

  ❑ `sudo arpspoof -i INTERFACE -t GATEWAY_IP CLIENT_IP`

  ❑ `sudo arpspoof -i INTERFACE -t CLIENT_IP GATEWAY_IP`

```
csc2025@csc2025-vbox:~$ sudo arpspoof -i enp0s3 -t 10.0.2.6 10.0.2.1        csc2025@csc2025-vbox:~$ sudo arpspoof -i enp0s3 -t 10.0.2.1 10.0.2.6
[sudo] password for csc2025:                                                [sudo] password for csc2025:
8:0:27:b5:13:37 8:0:27:26:3a:90 0806 42: arp reply 10.0.2.1 is-at 8:0:27:b5:13:37    8:0:27:b5:13:37 52:54:0:12:35:0 0806 42: arp reply 10.0.2.6 is-at 8:0:27:b5:13:37
8:0:27:b5:13:37 8:0:27:26:3a:90 0806 42: arp reply 10.0.2.1 is-at 8:0:27:b5:13:37    8:0:27:b5:13:37 52:54:0:12:35:0 0806 42: arp reply 10.0.2.6 is-at 8:0:27:b5:13:37
8:0:27:b5:13:37 8:0:27:26:3a:90 0806 42: arp reply 10.0.2.1 is-at 8:0:27:b5:13:37    8:0:27:b5:13:37 52:54:0:12:35:0 0806 42: arp reply 10.0.2.6 is-at 8:0:27:b5:13:37
8:0:27:b5:13:37 8:0:27:26:3a:90 0806 42: arp reply 10.0.2.1 is-at 8:0:27:b5:13:37    8:0:27:b5:13:37 52:54:0:12:35:0 0806 42: arp reply 10.0.2.6 is-at 8:0:27:b5:13:37
```

● Victim VM executes `arp -a` to check ARP table

  ❑ If the gateway's mac address is the same with that of the attacker, ARP spoofing is successful

```
csc2025@csc2025-vbox:~/Project1$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::5c6:c4ed:b631:71f9  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:b5:13:37  txqueuelen 1000  (Ethernet)
        RX packets 140  bytes 47456 (47.4 KB)
```

```
csc2025@csc2025-vbox:~$ arp -a
? (10.0.2.15) at 08:00:27:b5:13:37 [ether] on enp0s3
? (10.0.2.3) at 08:00:27:58:a7:12 [ether] on enp0s3
_gateway (10.0.2.1) at 08:00:27:b5:13:37 [ether] on enp0s3
```

MITM Agent   TLS Client

Task I: Hijacking a TLS Connection

● **TLS Client to MITM Agent:**

❏ The MITM agent can use a forged certificate to establish a TLS connection.

▪ Configure the server settings (TLS version, check mode, etc.) so that the victim accepts the TLS connection.

● **MITM Agent to TLS server:**

❏ The MITM agent can retrieve the destination address from the victim's packet

❏ The MITM agent uses this address to connect to the TLS server.

▪ A fixed address for the TLS server connection is not allowed.

• Should be able to connect to different websites.

Task II: Hijacking multiple TLS conn. concurrently

- **The program should still work normally when connecting to another website**
  - ❑ Handling concurrency
    - ■ Ensure the program can manage multiple TLS connections concurrently
    - ■ Consider using threading, fork(), or asynchronous I/O (select(), epoll()) to avoid blocking connections

  - ❑ Session management
    - ■ Each connection should maintain its own independent TLS session context
    - ■ Avoid session interference between multiple websites being accessed simultaneously

Verification Steps

- **1. MITM agent can correctly hijack a TLS connection (60%)**
  - ☐ A sub-connection between TLS client and MITM agent
  - ☐ A sub-connection between MITM agent and TLS server

- **2. Fetch the username/password and show on the terminal (20%)**
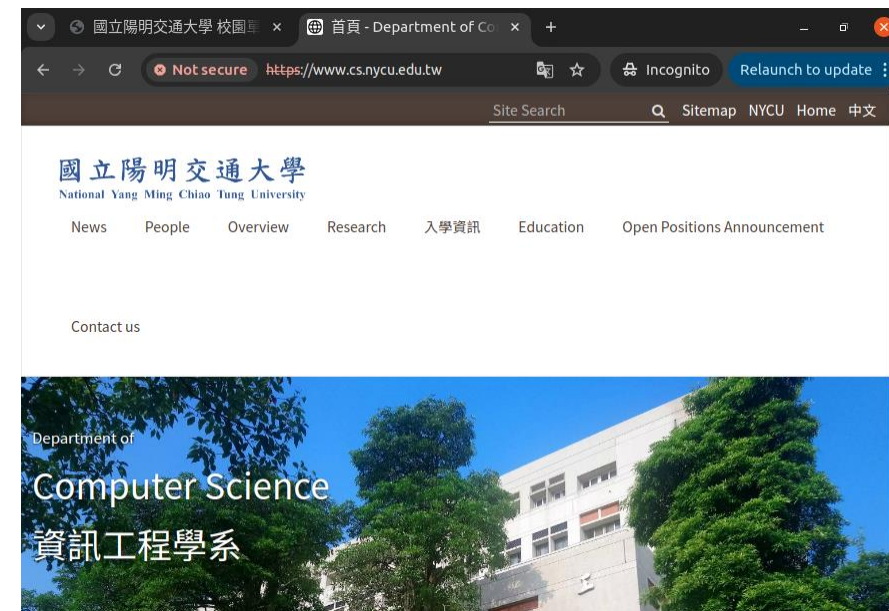 - ☐ MITM agent prints out the username/password inputted to nycu portal

- **3. MITM agent can concurrently hijack multiple TLS connections (20%)**

Verification Steps

● 1. MITM agent can correctly hijack a TLS connection (60%)

☐ When executing the attack program, the client can successfully connect to the school's portal webpage.

☐ The program should also print out the destination IP and port.

Verification Steps

● 2. Fetch the username/password and show on the terminal (20%)

□ MITM agent prints out the username/password inputted to nycu portal
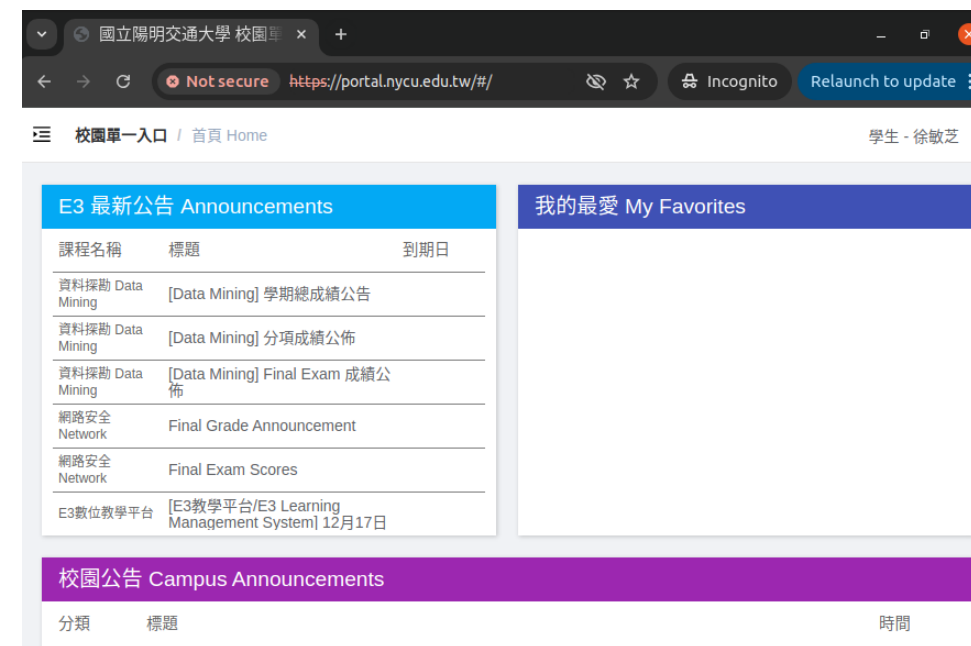
Verification Steps

● 3. MITM agent can concurrently hijack multiple TLS connections (20%)

  ❑ The program still works normally when connecting to other HTTPS websites

Important: How to Prepare Your Attack Programs?

- **You need to develop and run your program in the provided VM**
  - ☐ **VM Image**: Please download it from the provided <u>link</u>
    - Username/password: csc2025/csc2025
  - ☐ Network setting: **NAT Network**

- **Do not hardcode the network interface. You are allowed to assign it during execution.**

  - ☐ During the demo, the program may be run on either VMware or VirtualBox, so ensure that no fixed values are used.

- **Only Python is allowed for the development.**

Important: How to Prepare Your Attack Programs?

- Must provide an attack program named **attack.py** (Missing: -20%)

- Test requirements for the program

  ☐ Due to the environment settings, this project focuses on hijacking websites within the school's IP domain (140.113.*.*)

  ■ You can use the nslookup command to verify if a specific host is within the school IP domain

  ☐ During the demo, all certificates will be provided by the TA and will be located in the ../certificates/directory

- The program must work with the following test commands:

  ☐ `sudo ./attack.py <victim ip>` or `sudo ./attack.py <victim ip> <interface>`

- You are allowed to team up. Each team has at most 2 students.

  ☐ Teams: discussions are allowed, but no collaboration

Project Submission

- Due date: 3/19

- Makeup submission (75 points at most): TBA (After the final)

- Submission rules

  ☐ Put your source code files into a directory and name it using your student ID(s)

  ▪ If your team has two members, please concatenate your IDs separated by "-"

  ☐ Zip the directory and upload the zip file to E3 (only upload python files)

  ☐ A sample of the zip file: 01212112-02121221.zip

  ```
  01212112-02121221.zip
  └── 01212112-02121221(dir)
       ├── attack.py
       └── bbb.py
  ```

☐ If files are not in a directory after unzip, 10 points will be deducted.

Online Project Demo

- Demo date: 3/21

- TA will prepare your zip file and run your programs for the demo on behalf of you

  □ TA will run your program in the same given virtual environment

- You will

  □ be asked to launch a TLS hijacking attack

  □ be not allowed to modify your codes or scripts in the demo

  □ be asked some questions

  □ be responsible to show and explain the outcome to TA

Questions?