

# Computer Security Capstone

## Project II: MITM and Pharming Attacks in Wi-Fi Networks

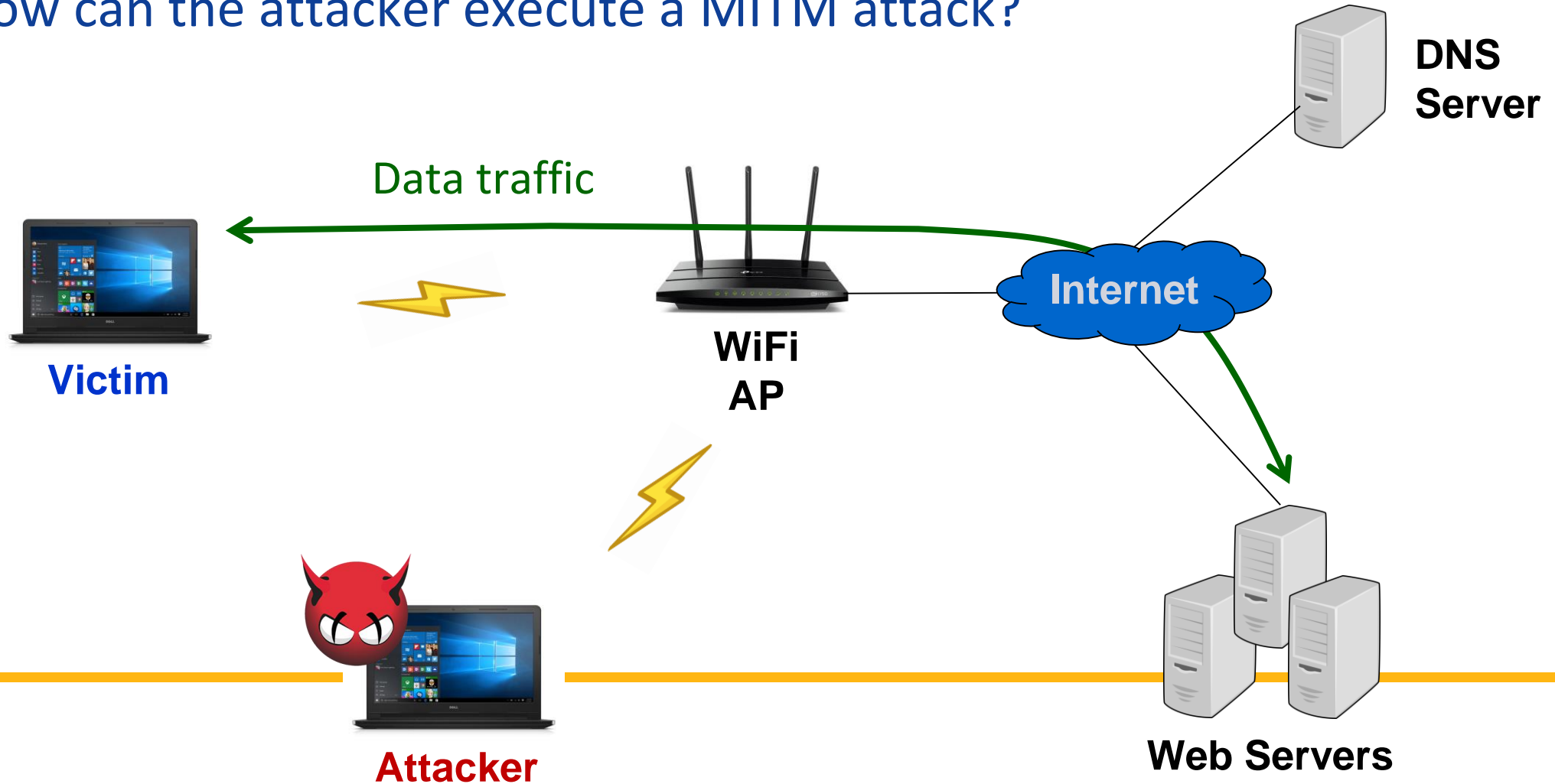
Chi-Yu Li (2025 Spring)  
Computer Science Department  
National Yang Ming Chiao Tung University

# Goal

- Understand how user can be phish by a man-in-the-middle (MITM) attack over Wi-Fi networks
- You will learn how to
  - ❑ scan IP/MAC addresses of the devices in a Wi-Fi network
  - ❑ launch a man-in-the-middle (MITM) attack by using ICMP redirect
  - ❑ launch a pharming attack

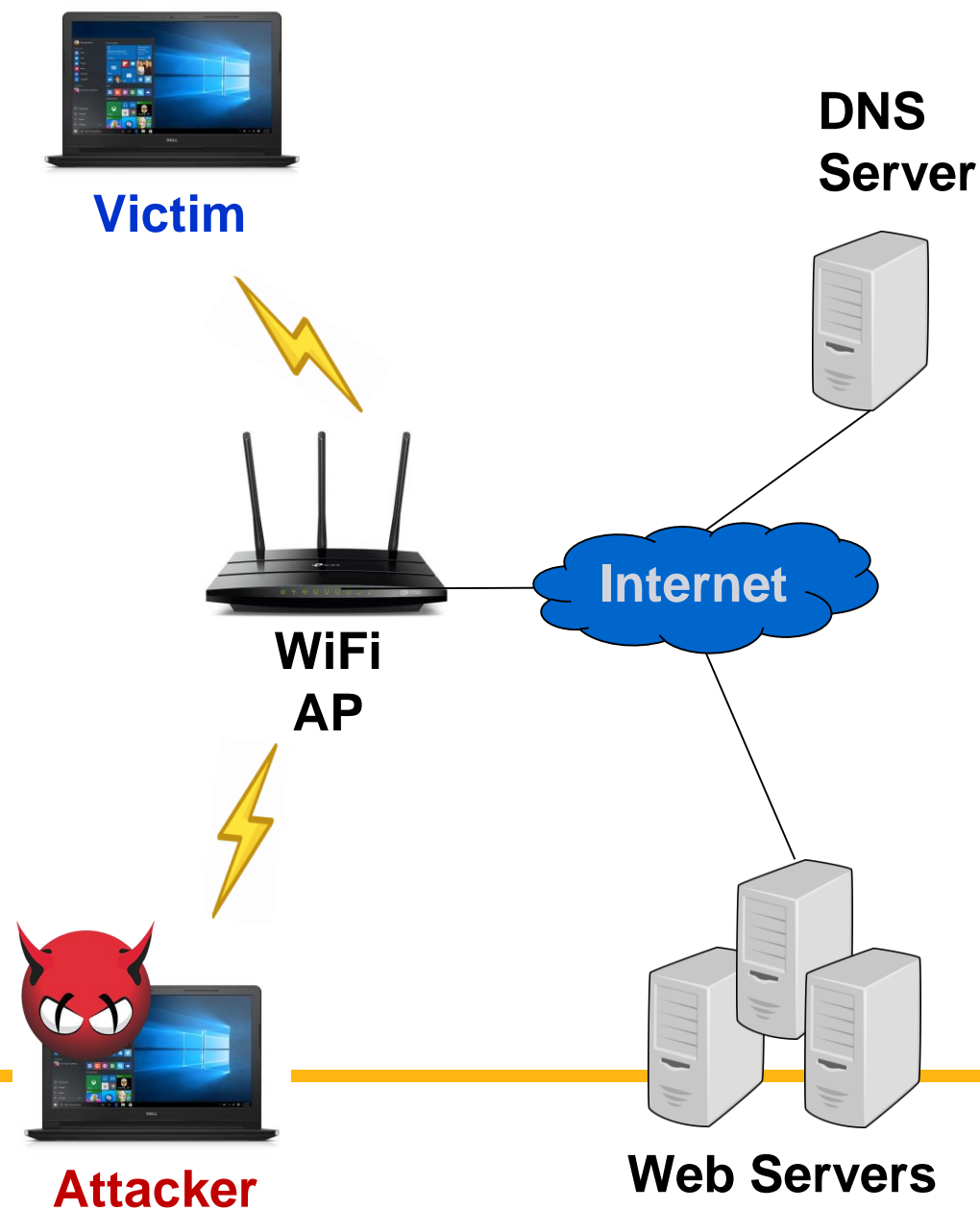
# Attack Scenario

- How can the attacker execute a MITM attack?



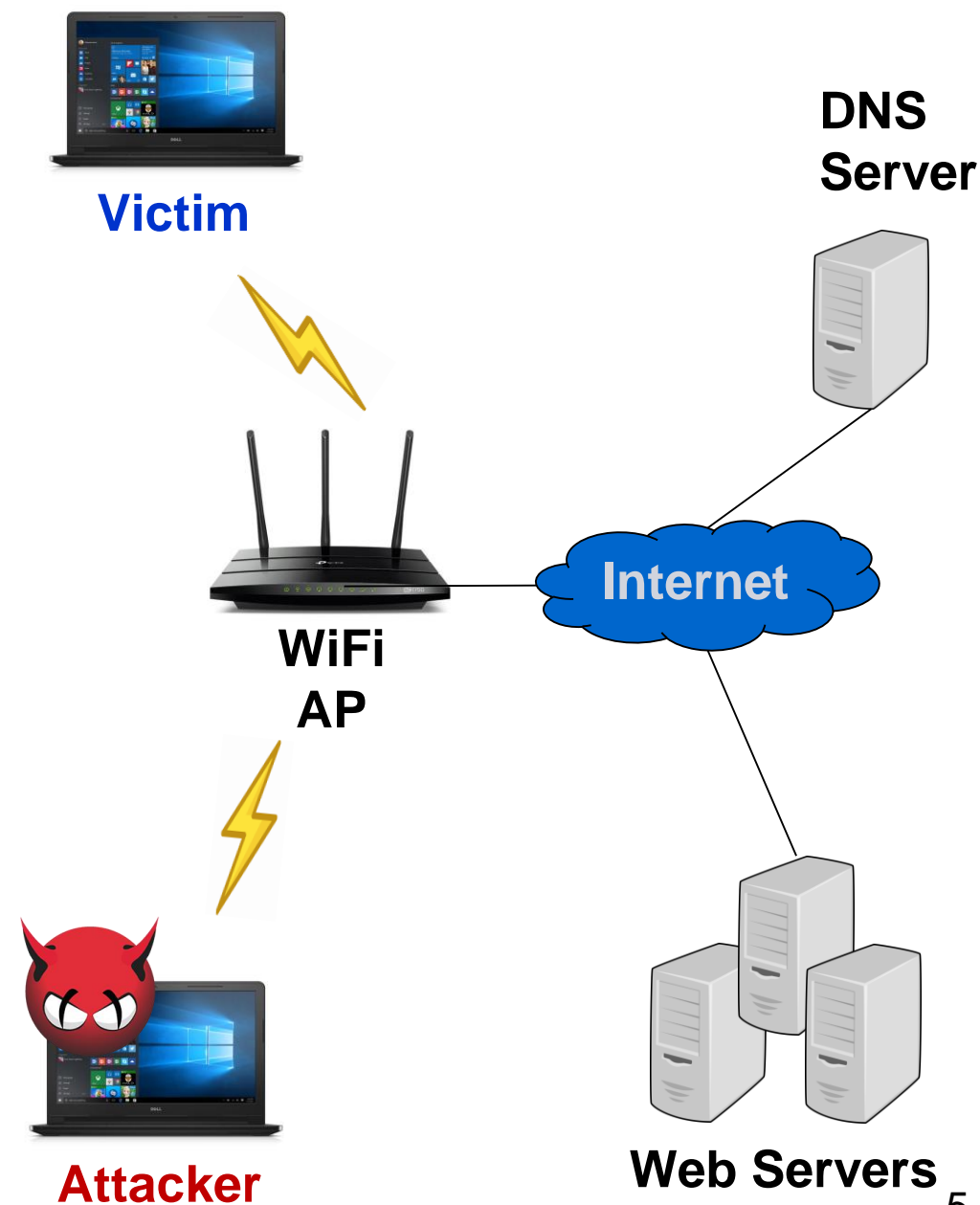
# Major Ideas

- Identify the Victim
  - Device Address Information Collection
- Redirect Victim Traffic
  - ICMP redirect attack
- Launch Pharming Attack
  - DNS Spoofing



# Tasks: MITM and Pharming

- Device Address Information Collection (Task I: 20%)
  - ❑ Obtain all other client devices' IP/MAC addresses in a connected Wi-Fi network
- ICMP redirect attack (Task II: 30%)
  - ❑ ICMP redirect for specific client device in the Wi-Fi network
- DNS Spoofing (Task III: 30%)
  - ❑ DNS spoofing attack for web services
- Some implementation question during the demo (20%)



# Task I: Device Address Information Collection

- Scan all the devices' IP/MAC addresses in the Wi-Fi network
  - Hint: ARP format and raw socket.
- Fetch the IP/MAC addresses of all the other client devices

```
● csc2025-attacker@csc2025-attacker:~/Desktop/csc2025-project2$ sudo ./icmp_redirect 163.182.194.25
Available devices
-----
Index | IP      | MAC
-----
0     | 10.0.2.1 | 52:54:00:12:35:00
1     | 10.0.2.2 | 52:54:00:12:35:00
2     | 10.0.2.3 | 80:00:27:36:DA:66
3     | 10.0.2.16 | 80:00:27:6F:58:50
-----
```

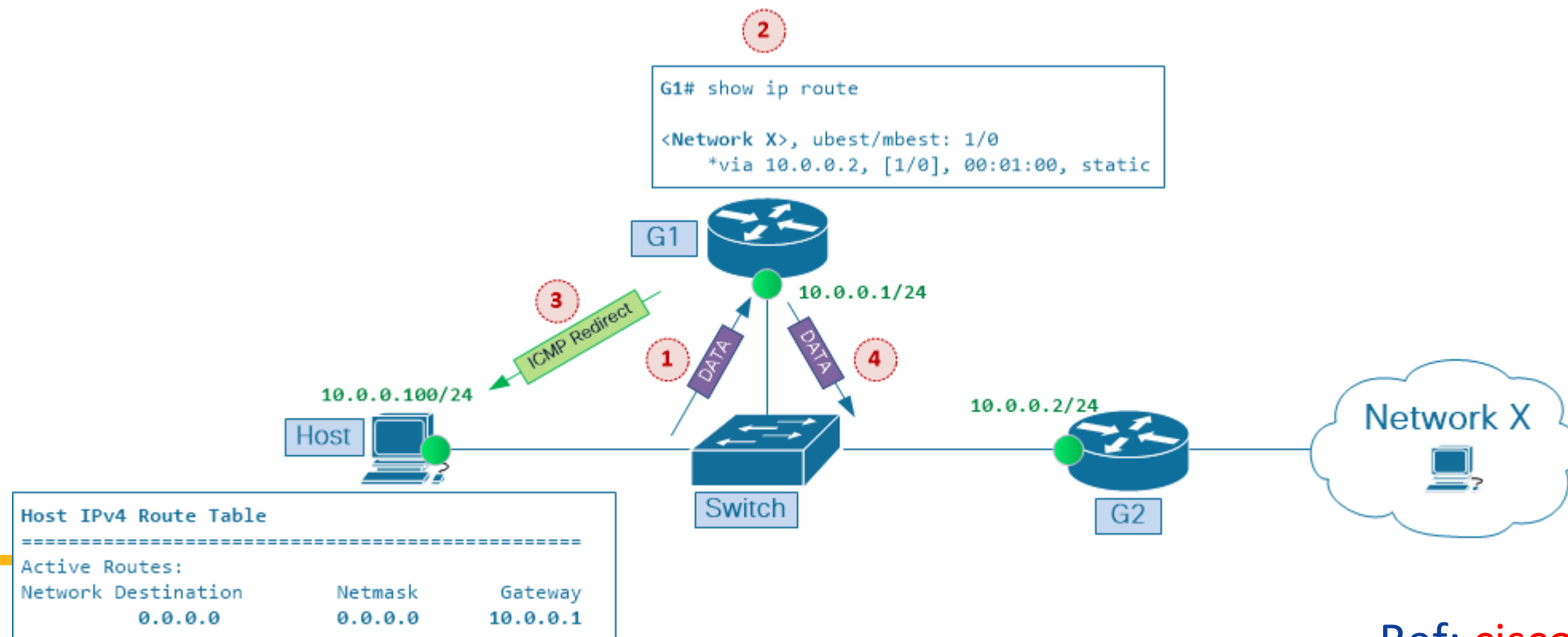
# Task II: ICMP redirect attack

## ● What is ICMP redirect?

- ❑ It is a type of ICMP (Internet Control Message Protocol) message used to inform a host to update its routing information.
  - Purpose: Informs the host of a better route for its traffic.
  - Trigger Condition: Occurs when a router detects unnecessary routing (e.g., packets looping back to the same network segment).
  - Message Content: Contains the IP address of the new next-hop router.
  - Common Use: Optimizes routing by reducing unnecessary hops.

## Task II: ICMP redirect attack (Cont.)

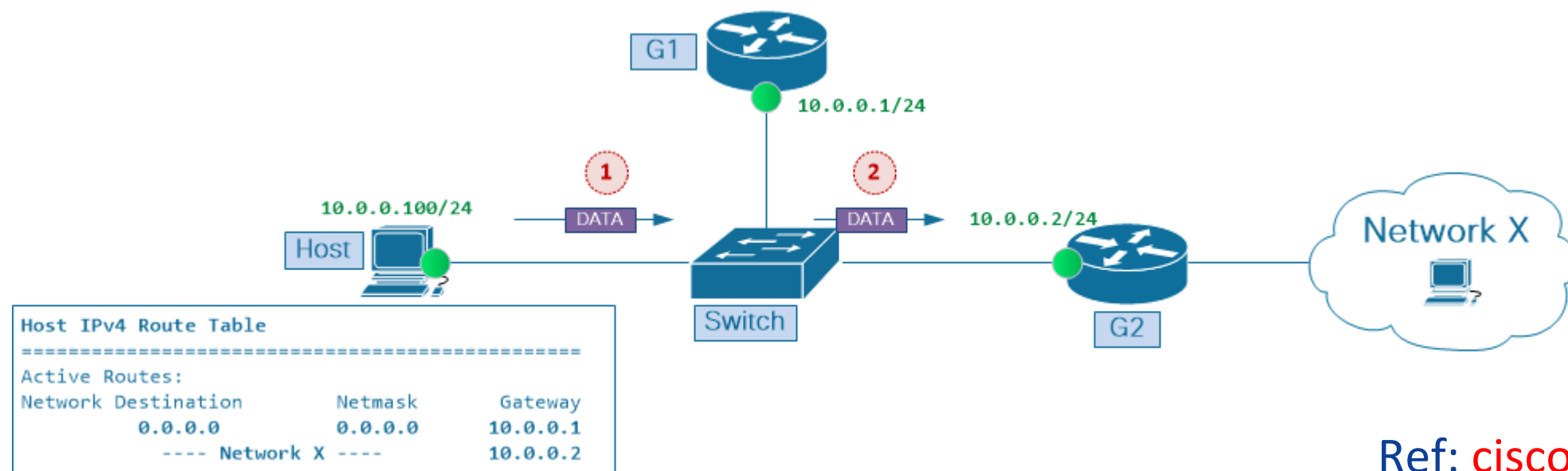
- Router G1 identifies a better route through the G2 and sends an ICMP redirect to the host to update its routing table.





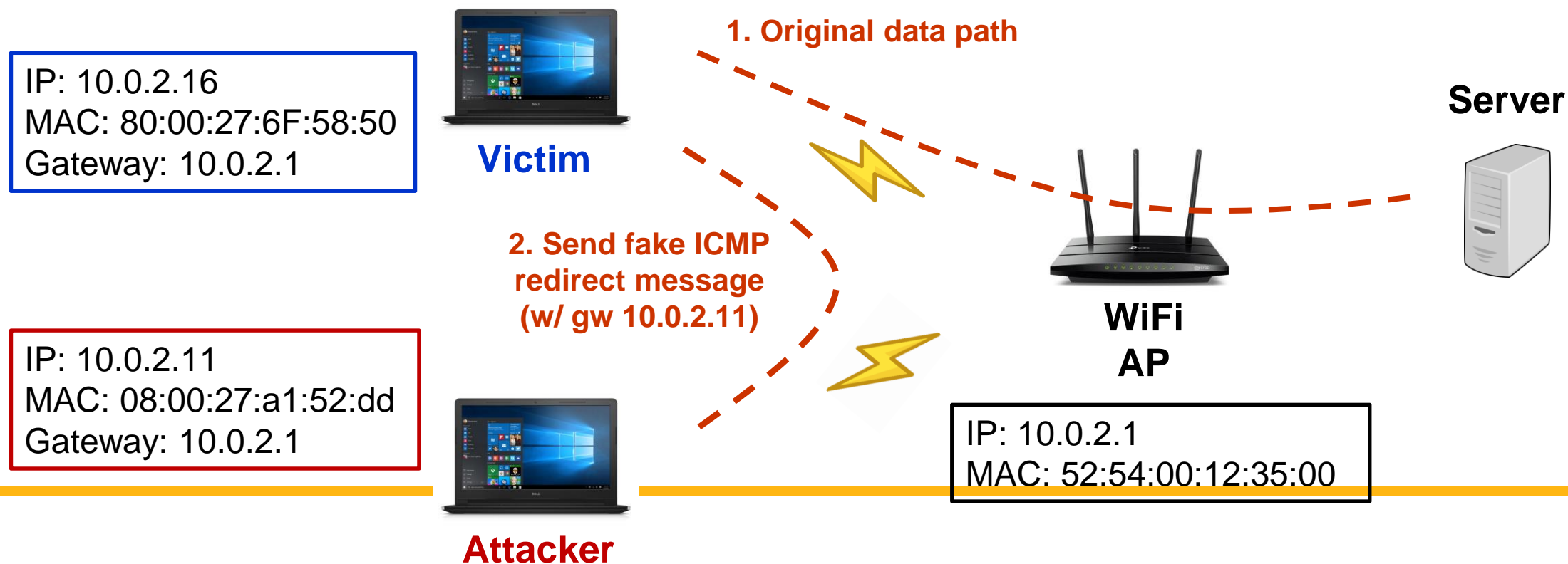
## Task II: ICMP redirect attack (Cont.)

- The host updates its routing table and sets the gateway to G2.



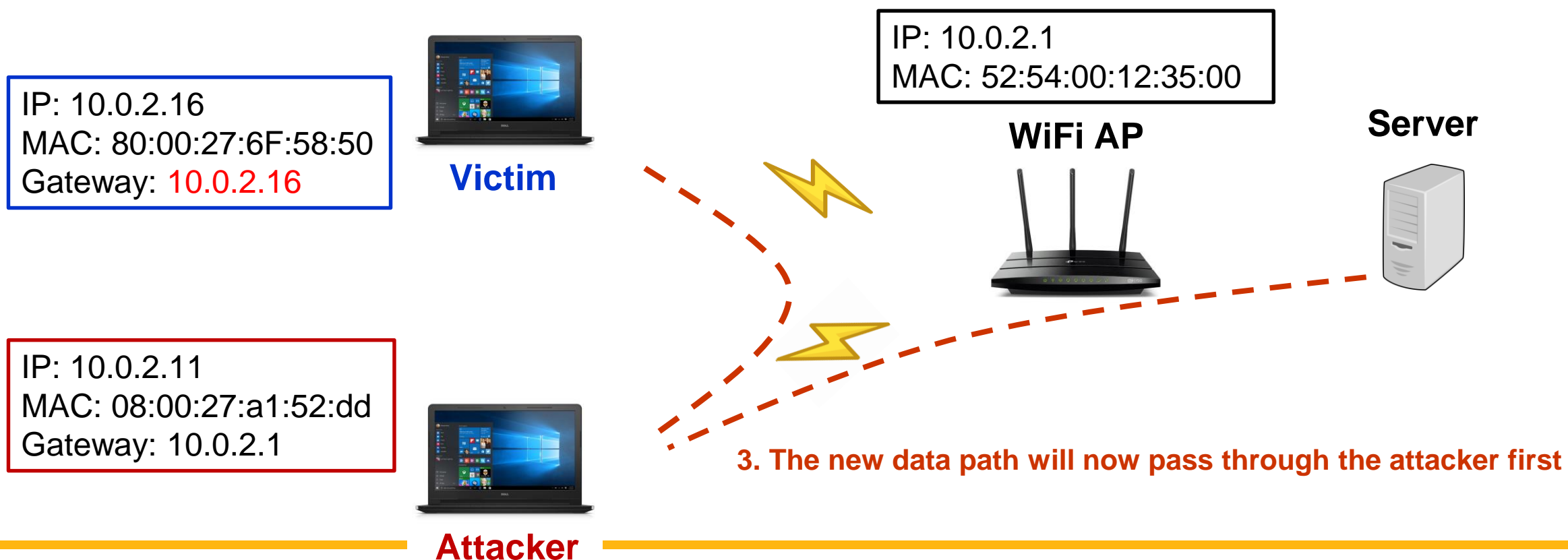
## Task II: ICMP redirect attack (Cont.)

- The attacker generates a fake ICMP redirect packet, claiming there is a better route for the victim's traffic.



## Task II: ICMP redirect attack (Cont.)

- The victim updates its routing table and sets the gateway as the attacker.



## Task II: ICMP redirect attack (Cont.)

- Attacker scans devices and sends fake ICMP redirect message to the victim.

```
csc2025-attacker@csc2025-attacker:~/Desktop/csc2025-project2$ sudo ./icmp_redirect 163.182.194.25
Available devices
-----
Index | IP | MAC
-----
0 | 10.0.2.1 | 52:54:00:12:35:00
1 | 10.0.2.2 | 52:54:00:12:35:00
2 | 10.0.2.3 | 80:00:27:36:DA:66
3 | 10.0.2.16 | 80:00:27:6F:58:50
-----
Select Victim IP index: 3
Select Gateway IP index: 0
Victim IP: 10.0.2.16, Gateway IP: 10.0.2.1, Attacker IP: 10.0.2.11
ICMP Redirect packet sent successfully!
```

- The victim updates its routing table and sets the gateway as the attacker.

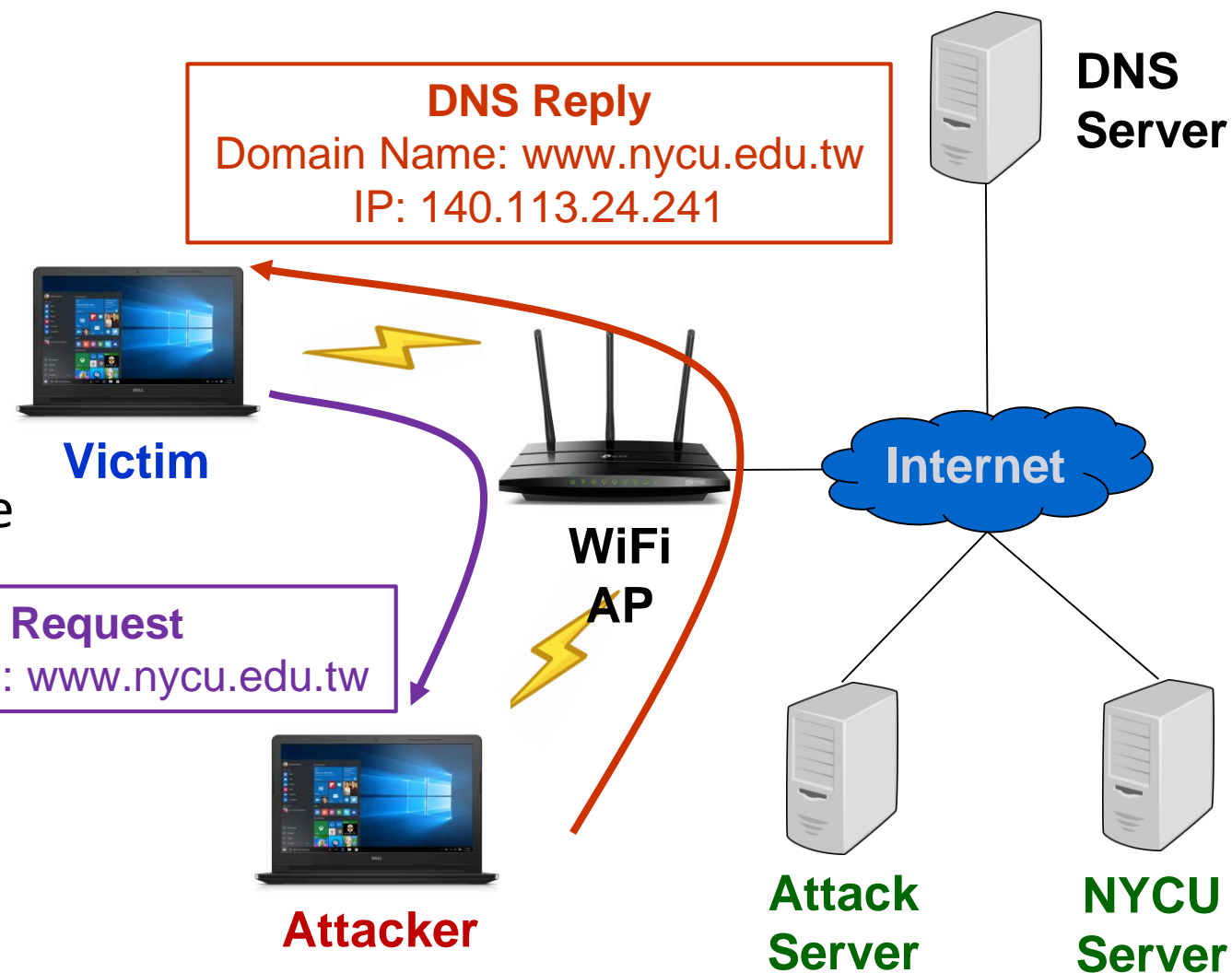
```
csc2025-victim@csc2025-victim:~$ ip route get 163.182.194.25
163.182.194.25 via 10.0.2.11 dev enp0s3 src 10.0.2.16 uid 1000
cache <redirected> expires 296sec
```

# Task III: DNS Spoofing

- Intercept DNS requests for a specific web page and generate spoofed DNS replies with the attack server's IP

□ Hint: DNS format, Netfilter queue

□ Note: You should drop the origin victim DNS request

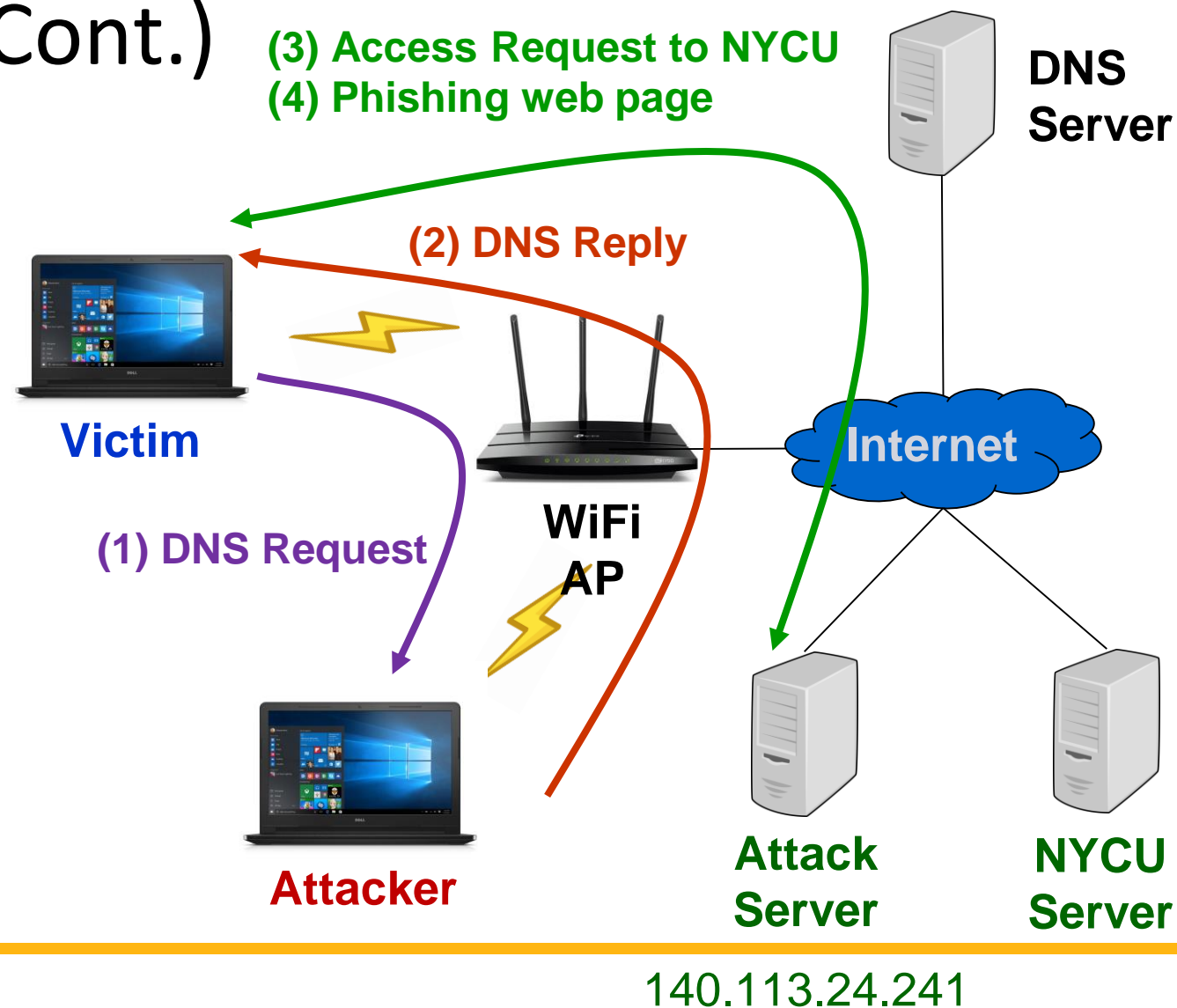


140.113.24.241

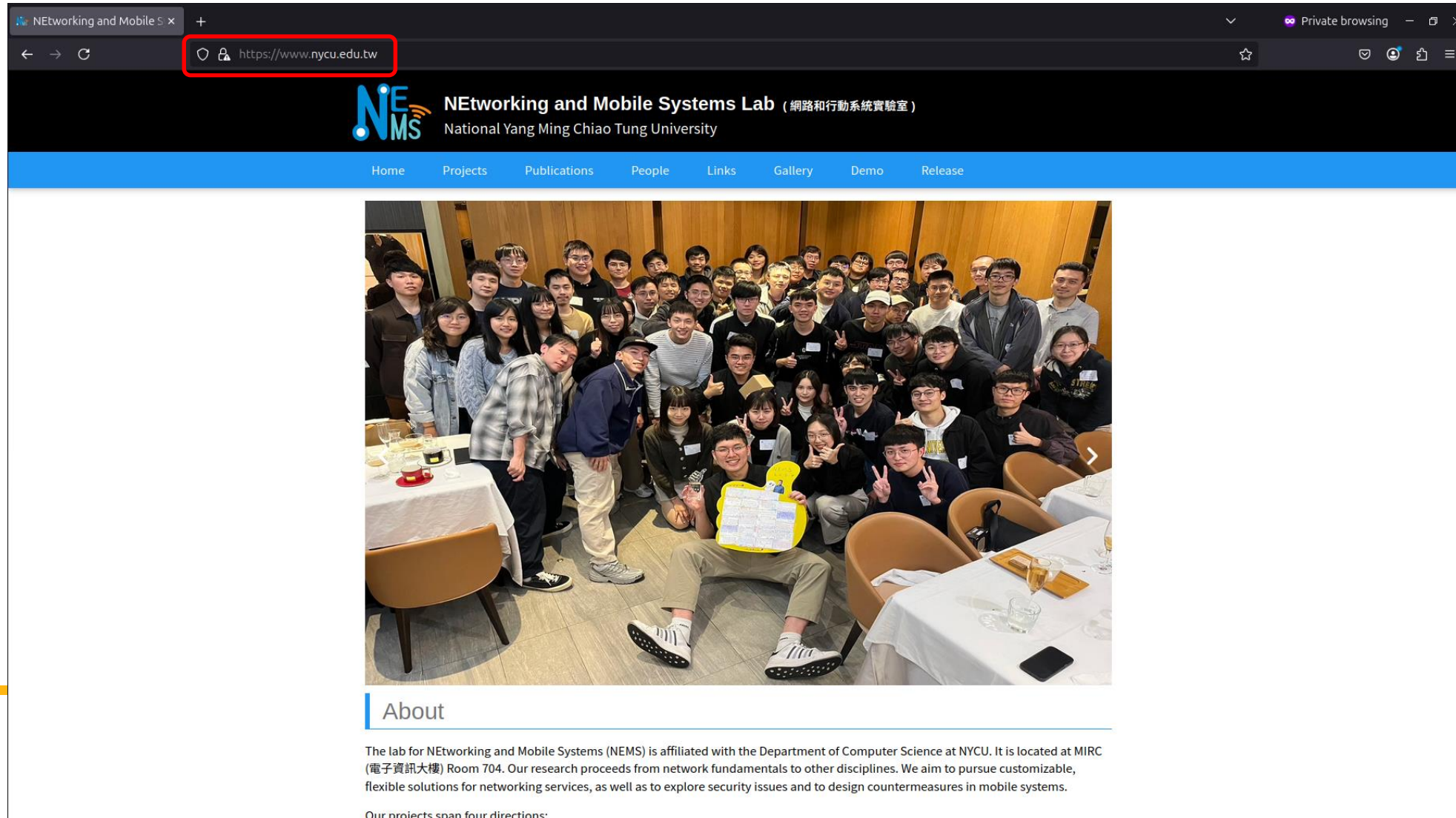
## Task III: DNS Spoofing (Cont.)

### ● Successful attack

- ❑ An access request to NYCU home page will be redirected to the attack server (140.113.24.241)
- ❑ A phishing web page will be shown to Victim




# Task III: DNS Spoofing (Cont.)



NETworking and Mobile Systems Lab (網路和行動系統實驗室)  
National Yang Ming Chiao Tung University

Home Projects Publications People Links Gallery Demo Release



## About

The lab for NETworking and Mobile Systems (NEMS) is affiliated with the Department of Computer Science at NYCU. It is located at MIRC (電子資訊大樓) Room 704. Our research proceeds from network fundamentals to other disciplines. We aim to pursue customizable, flexible solutions for networking services, as well as to explore security issues and to design countermeasures in mobile systems.

Our projects span four directions:



# Requirements

- You need to develop/run your program in a given virtual machine
  - ❑ VM image: Please download it from [Link](#)
    - Username/password: csc2025/csc2025
  - ❑ Network setting: **NAT Network**
- Only C/C++ is allowed for the development
  - ❑ To be better familiar with the protocols (Python is not allowed)



# Requirements

- Do not hardcode the network interface.
  - During the demo, the program may be run on either VMware or VirtualBox, so ensure that no fixed values are used
- You are allowed to team up. Each team has at most 2 students
  - Teams: discussions are allowed, but no collaboration
- Please only submit your source codes to E3

# Important: How to Prepare Your Attack Programs?

- Must provide a **Makefile** which compiles your source codes into two executable files, named **icmp\_redirect** and **pharm\_attack** (Missing: -20%)
- Test requirements for the programs
  - ❑ Must be run in the given VM without any additional tools or libraries
  - ❑ Must use the following parameters
    - DNS spoofing for the NYCU home page: <http://www.nycu.edu.tw> (use private window)
    - Attacker server IP in the DNS spoofing: [140.113.24.241](#)
  - ❑ Must work for test commands

# Important: How to Prepare Your Attack Programs?

## ● Results from the ICMP redirect attack

- ❑ Only allowed to run `./icmp_redirect <address> <interface>`
- ❑ Print out the IP/MAC addresses of **all the Wi-Fi devices/VMs** except for **Attacker**
- ❑ Select the target victim and origin gateway as the struct of fake ICMP redirect packet
- ❑ The victim should update its dynamic routing table (check by “`ip route get <address>`” on victim)

## ● Results from the pharming attack

- ❑ Only allowed to run `./pharm_attack` (**No manual configuration is allowed**)
- ❑ Redirect the NYCU home page ([www.nycu.edu.tw](http://www.nycu.edu.tw)) to the phishing page ([140.113.24.241](http://140.113.24.241))

# Project Submission

- Due date: 4/16 11:55pm
  - ❑ Late submission is not allowed
- Makeup submission (75 points at most): TBA (After the final)
- Submission rules (Wrong file name or format will result in 10 points deduction)
  - ❑ Put all your 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 New E3
  - ❑ A sample of the zip file: 01212112-02121221.zip
    - 01212112-02121221
      - Makefile
      - icmp\_redirect.cpp
      - pharm\_attack.cpp
      - ....

# Online Project Demo

- Date: 4/18
- TA will prepare your zip file and run your programs for the demo on behalf of you
- You will
  - ❑ be asked to reproduce your ICMP redirect and pharming attacks
  - ❑ be only allowed to use “make” to compile all your files, and run your attack binary programs or scripts
  - ❑ be not allowed to modify your codes or scripts
  - ❑ be asked some questions
  - ❑ be responsible to show and explain the outcome to TA

# Hint 1: ICMP Redirect Attack

## ● ICMP redirect format

□ Reference : [ICMP Redirect Message Format](#)

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Gateway Internet Address (32 bits)		
Internet Header + 64 bits of Original Data Datagram (Variable)		

□ You can simply craft Internet Header + 64 bits of Original Data Datagram as an ICMP echo reply packet with a checksum set to 0xffff, and both the Identifier and Sequence Number set to 0.

## Hint 1: ICMP Redirect Attack (Cont.)

- To enable the system to accept ICMP redirects, the victim needs to configure the following network settings to 1:
  - ❑ `net.ipv4.conf.all.accept_redirects`
  - ❑ `net.ipv4.conf.default.accept_redirects`
  - ❑ `net.ipv4.conf.<interface_name>.accept_redirects`

# Hint 2: DNS Spoofing

- DNS format

Identification (16 bits)	Flags (16 bits)	Header 12 bytes
Number of questions (16 bits)	Number of answer RRs (16 bits)	
Number of authority RRs (16 bits)	Number of additional RRs (16 bits)	
Questions (variable bits)		
Answers (variable bits)		
...		



## Hint 2: DNS Spoofing (Cont.)

- DNS packets may employ message compression
  - Reference: <https://datatracker.ietf.org/doc/html/rfc1035#section-4.1.4>

# Hint 3: Netfilter queue

## ● Functions:

- ❑ `nfq_open()`: open a nfqueue handler.
- ❑ `nfq_close()`: close a nfqueue handler.
- ❑ `nfq_unbind_pf()`: unbind nfqueue handler from a protocol family.
- ❑ `nfq_bind_pf()`: bind a nfqueue handler to a given protocol family.
- ❑ `nfq_create_queue()`: create a new queue handle and return it.
- ❑ `nfq_destroy_queue()`: destroy a queue handle.
- ❑ `nfq_set_mode()`: set the amount of packet data that nfqueue copies to userspace.
- ❑ `nfq_fd()`: get the file descriptor associated with the nfqueue handler.
- ❑ `nfq_handle_packet()`: handle a packet received from the nfqueue subsystem.
- ❑ `nfq_get_msg_packet_hdr()`: return the metaheader that wraps the packet.
- ❑ `nfq_get_payload()`: get payload.
- ❑ `nfq_set_verdict()`: issue a verdict on a packet.

- Reference: [https://netfilter.org/projects/libnetfilter\\_queue/doxygen/html/index.html](https://netfilter.org/projects/libnetfilter_queue/doxygen/html/index.html)

## Hint 3: Netfilter queue(Cont.)

- `nfq_set_verdict(struct nfq_q_handle *qh, uint32_t id, uint32_t verdict, uint32_t data_len, const unsigned char *buf):`
  - ❑ **Only read the packet:** `data_len=0, *buf = nullptr`
  - ❑ **Modify packet:** `data_len=new packet length, *buf = Head of buffer that stored new packet data`

# Reference 1: ICMP Redirect Attack

- This paper provides guidance on how to probe the victim in order to construct an accurate Internet Header along with the first 64 bits of the Original Data Datagram.
  - ["Man-in-the-Middle Attacks without Rogue AP: When WPA's Meet ICMP Redirects," 2023 IEEE S&P, San Francisco, CA, USA, 2023](#)
- In this project, we **don't** need to do it.

# Questions?