# 網路實驗

Mininet

Mac address table overflow attack

2022 Spring

T-C. Hou

# Outline

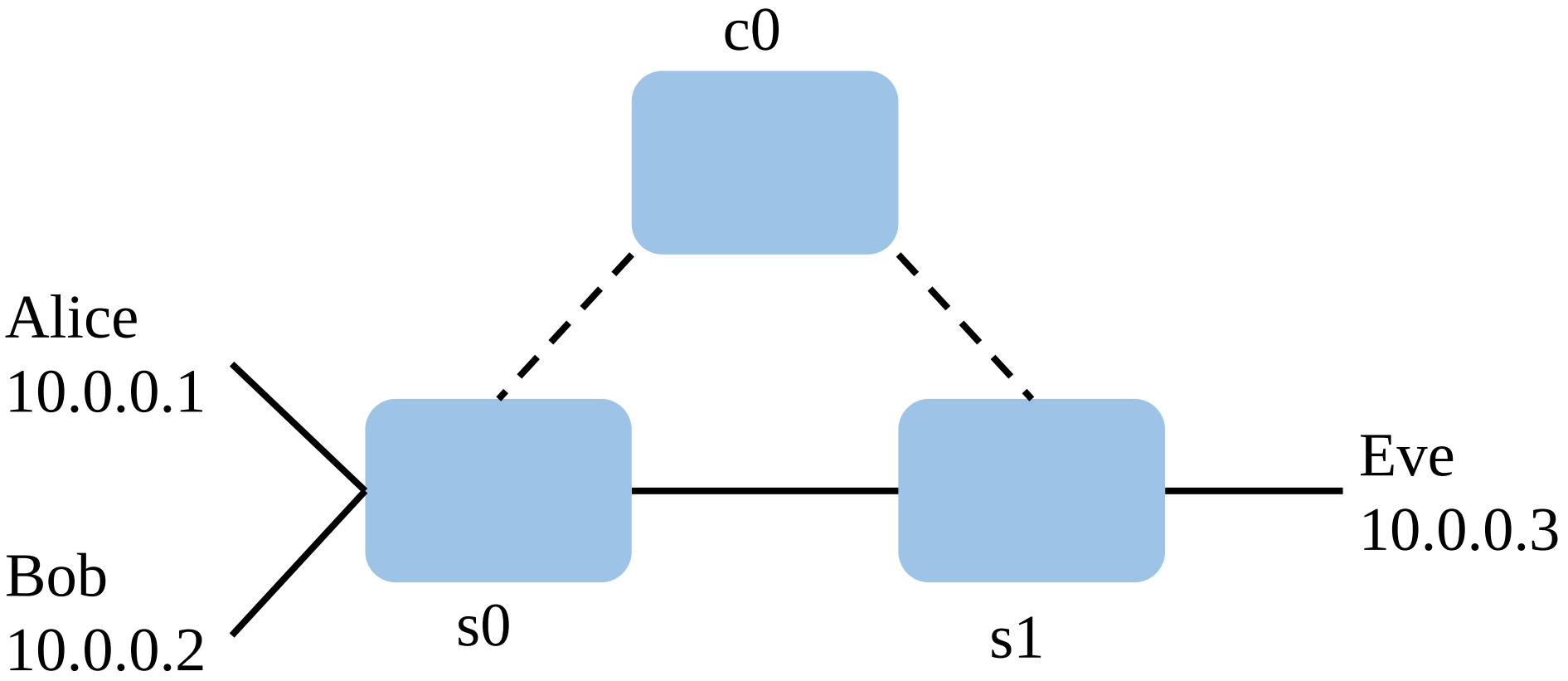
- Introduction
- Installation
- Attack Procedure

# Introduction

- We have learned how a learning switch works in the previous lab.
- In this lab, you are going to let a third party (Eve) eavesdrop the traffic between Alice and Bob by overflowing the mac address table in the switches.
- This attack would work even if Eve is not connected to the same switch used by Alice and Bob.
- Ref: https://github.com/mininet/mininet/wiki/Mac-address-table-overflow-attack

# Introduction

- The topology we use in the lab is :



# Installation

- Install ltprotocol
  - $ pip3 install ltprotocol
- Install impacket
  - $ git clone https://github.com/SecureAuthCorp/impacket.git
  - $ cd impacket/
  - $ sudo pip3 install .
  - $ sudo python3 setup.py install

# Installation

- Check out starter code (Download from Google Drive)
  - https://drive.google.com/drive/folders/1heuHoqCwRaihCKMadGcIRYAT4fJ1lUxd?usp=sharing
  - $ cd ~/Download
  - Untar file to home page
  - $ tar -C ~/ -zxvf cs144_security.tar.gz
  - $ cd ~/cs144_security/
  - $ ./config.sh

# Attack Procedure

- **Start POX network Controller**
  - In one terminal
  - $ cd ~/cs144_security
  - $ ./run_pox.sh
- **Start Mininet Emulation**
  - In another terminal
  - $ cd ~/cs144_security
  - $ ./run.sh

–This will start the Mininet network emulator and there will be terminals pops up for each of the nodes in the network. Close the terminals for switches and controllers, but keep the terminals for Alice, Bob and Eve.

# Attack Procedure

- **In Eve's and Bob's terminals**
  - $ tcpdump -n host 10.0.0.1
- **In Alice terminal**
  - $ ping 10.0.0.2

PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.

64 bytes from 10.0.0.2: icmp_req=1 ttl=64 time=100 ms

64 bytes from 10.0.0.2: icmp_req=2 ttl=64 time=98.2 ms

64 bytes from 10.0.0.2: icmp_req=3 ttl=64 time=96.7 ms

64 bytes from 10.0.0.2: icmp_req=4 ttl=64 time=58.5 ms

64 bytes from 10.0.0.2: icmp_req=5 ttl=64 time=56.3 ms

# Attack Procedure

- **In the mininet terminal, open another xterm for eve**
  - mininet> xterm eve
- **In the second eve terminal**
  - $ python3 attack.py

# Attack Procedure

- What will Eve observe in the first xterm?
- In your report, explain
  - What did Eve do in her attack?
  - Were Eve's attack frames delivered?
  - Why is switch's MAC table overflow?
  - Why can Eve observe Alice's packets?