

**行動通訊網路**

**Mobile Communication Network**

**課程實驗LAB**

**教授：張宏慶 [jang@cs.nccu.edu.tw](mailto:jang@cs.nccu.edu.tw)**

**助教：劉育佑 [111753145@nccu.edu.tw](mailto:111753145@nccu.edu.tw)**

# 目錄

- 實驗規定
- 使用工具
- Lab-0 基本環境安裝
- Lab-1 測試NS-3
- Lab-2 執行NS-3 程式碼
- Lab-3 視覺化工具NetAnmin
- Lab-4 以NS-3模擬5G use case scenario

# 實驗規定

- 實驗繳交一份PDF檔
- 檔案名稱: 學號\_Lab.pdf
- PDF內需包含學號及姓名
- 一人一份，頁數不拘。
- 繳交期限：2024/1/5

# 使用工具

- Network Simulator NS-3
- Docker（選用，可直接用terminal執行）
- XServer（Lab 3）
- WireShark (Lab 2)
- TCPDUMP (Lab 2)

# 實驗概述

實驗一：執行NS-3中的test.py

實驗二：以waf執行NS-3中的third.cc檔，並以WireShark分析產生的pcap檔

實驗三：以NetAnim分析同樣在third.cc執行後產生的xml檔

實驗四：以waf 執行Lab1-4.cc，並從其執行結果分析並思考問題

# Lab-0 基本環境安裝

## NS-3

- NS-3官方網站：<https://www.nsnam.org/>
- NS-3官方教學網站：<https://www.nsnam.org/docs/tutorial/html/>
- NS-3官方資料庫：<https://www.nsnam.org/doxygen/index.html>
- 本次實驗使用**ns3.35**版本，其中使用的 waf 程式在ns3.36版本中被 Cmake取代無法使用。
- 目前最新版本為2023年9月更新的ns3.40，網路上以ns3.23及ns3.26此兩版本有較多的參考資源。
- 雖然在Windows和MacOS的環境下都可下載NS-3的檔案，但以安裝成功性及執行穩定度來說，較**建議在Linux的作業環境下安裝**。

# NS3安裝

- 第一種方法是利用Docker 直接安裝並執行container，並在執行中的container裡面進行操作
- 若Docker container無法開啟，可直接在Linux 作業系統中安裝，剩餘的實驗也可直接在Linux的Terminal中執行  
參考影片：[https://youtu.be/vU4GE2oKao0?si=rXsku\\_rn9KIpfuJi](https://youtu.be/vU4GE2oKao0?si=rXsku_rn9KIpfuJi)  
(影片底下說明欄有相關安裝指令)

# Docker

Docker File  $\longrightarrow$  Image  $\longrightarrow$  Container

下載 Docker File 指令：docker pull ...  $\rightarrow$  docker pull nightring/ns3

## XServer / XQuartz

可使運行中的container 調用 GUI ，跑出圖形介面應用程式

Windows 可使用 WSLG 內建可調用GUI程式，[參考連結](#)

Mac 可使用 XQuartz 。實驗三NetAnim若打不開有可能是沒安裝這個

**WireShark/TCPDUMP** 讀取封包軟體



# Docker container 環境建置

- Pull Docker Image
- Run Docker Image
- `docker ps` : 檢查正在執行的Container及其 ID
- `docker exec -it <Docker ID> /bin/bash` : 進入執行中的container

# Lab-1 測試NS-3

## Docker安裝NS-3

- 執行測試腳本
  - `cd /ns-3/ns-3.35`
  - `./test.py`
- 將 test.py 測試結果截圖

# Lab-2 執行NS-3程式碼

## Docker安裝NS-3

- 執行測試腳本
  - `cd /ns-3/ns-3.35`
  - `./test.py`
- 將 test.py 測試結果截圖

# Lab-2 執行NS-3程式碼

## 執行NS-3教學程式碼 - third.cc

- `cp -rf ./examples/tutorial/third.cc ./scratch`
- `vim ./scratch/third.cc`
- 更改 `tracing = true` 才可跑出pcap檔
- `./waf --run scratch/third.cc` （用waf去跑third.cc）
- 將產生的pcap檔導出
- `docker cp container_ID:/ns-3/ns-3.35/third-X-X.pcap .<路徑>` (4個檔案)
- 使用wireshark或tcpdump查看、截圖及解讀內容

- `cd examples/tutorial/`
- `cp third.cc ../../scratch/`
- `cd ../../scratch/`
- `vim third.cc`
- `i`（進入insert模式），加入標題、視覺化敘述、tracing改成true
- 按esc退出insert模式
- `:wq`（儲存並離開）

# Lab-2 執行NS-3程式碼

## Bonus

1. 更改third.cc中的變數，觀察更改前後執行的差異，並截圖及解讀內容
2. 讓third.cc執行結果可以顯示傳送資料的節點的位置

# Lab-3 視覺化工具NetAnim

## NS-3視覺化

- NS-3中能使用的視覺化工具有PyViz 和NetAnim兩種
- 本實驗使用NetAnim
  - #include “ns3/netanim-module.h”
  - AnimationInterface anim (“<file name>.xml”);
  - cd /ns-3/netanim-3.108
  - ./NetAnim

# Lab-3 視覺化工具NetAnim

## 執行NetAnim

- 請讓LAB-2的third.cc可以執行NetAnim，截圖並利用NetAnim解讀所產生的xml檔Stats和Packets的內容

## Bonus

- 更改third.cc程式碼中的變數觀察並截圖敘述執行結果的差異



# Lab-4 以NS-3模擬5G use case scenario

## NS-3之5G場景模擬

- 執行附檔之LAB1-4.cc，下載連結：[https://drive.google.com/file/d/1gwX\\_kql1Ki62PlnyKlfmGCLsae73PEE9/view?usp=sharing](https://drive.google.com/file/d/1gwX_kql1Ki62PlnyKlfmGCLsae73PEE9/view?usp=sharing)
- `docker cp <file_location>/lab-4.cc container_ID:/ns-3/ns-3.35/scratch` 將執行成功之畫面截圖，並回答以下問題：
  1. 試根據模擬及NetAnim的結果敘述此模擬過程做了哪些事
  2. 請問此模擬適合對應5G三大服務場景(URLLC/mMTC/eMBB)的何者
  3. 請問此模擬之程式碼中如何增加使用者數量
  4. 試觀察此模擬產生之txt檔，敘述模擬可能有的問題，及相對應可能的解決方法