Network Administration/System Administration Homework #1

B10202012 劉仲楷

Acknowledgement 這份作業有參考 ChatGPT，但都是用自己的話寫出，其餘資源都有標注在各題。

1 問答題

1. DAI 利用檢查 IP 和 MAC 的對應關係來防範 ARP spoofing 的攻擊手法。利用 DHCP 動態分配 IP 時，建立一個 IP-MAC-VLAN-port 的 binding table 作為對照（也可以直接手動設置），如果檢查發現 IP-MAC pair 和 snooping table 上的資料不相符，則丟棄。

2. (a)
   - Access mode: 連接孔只傳輸單一 VLAN 的 frame，送到 client 端的 frame 不帶 802.1Q tag，client 也不需要理解 VLAN tag。
   - Trunk mode：連接孔會傳輸多個 VLAN 流量，送到的 frame 會有 802.1Q tag，用來在 switch/router 之間傳輸。

   (b) 在 Trunk port 上，屬於 native VLAN 的封包不會加上 VLAN tag（其他都會），用來和不支援 VLAN tag 的設備相容，也可以留做當作一個不標記的通道。來自 Reddit 原生 VLAN 的目的是什麼？的回覆使用情境：為了設置 ZTP（zero touch provisioning，零接觸配置），我們可以將 native VLAN 設定 VLAN 10，用 DHCP 發放 bootp server 的 IP。然後接上一個新的 switch，但他沒設定過，所以會跑到 native VLAN。接他成果用 DHCP 拿到 IP 連上 bootp server 並下載、安裝設定檔。這時候他就可以傳送一般的封包，並走一般設定的 VLAN 了。

   (c) 一個 VTP domain 裡，switch 會透過 Trunk port 交換 VTP broadcast 資訊。Server 能建立、刪除、修改 VLAN，並更新給同 domain 的其他 switch。Client 只能收到資訊並同步。還有一種 Transparent 模式只轉送資訊不同步。優點是集中管理 VLAN 到 server 上，解決每台都要重複配置的麻煩。缺點是疏失/攻擊等導致的錯誤，就會影響整個 domain。

3. (a) 只是兩個不同來源的封包可以同時過，但個別來源傳輸速度還是一樣。傳輸量增加不等於傳輸速度提升。

   (b) Active 模式會主動發送 LACP 封包去協商 link aggregate。Passive 模式不主動發送 LACP 封包，只會在收到對端的 LACP 封包時才回應。

   (c) 沒人發起協商就不會成功 aggregate。

4. (a) 所有 switch 互相交換 BPDU (Bridge Protocol Data Unit)，以 Bridge ID（含 Priority 與 MAC）決定哪台 switch 當 root。每台 switch 根據到 root 的路徑成本決定最佳路徑。最後選出到 root 的最佳 port，以及會需要用來轉發的 port，剩餘的 port 都塞起來。

(b)
- Disabled：該 port 被手動關閉或停用，不參與 STP。
- Blocking：只接收 BPDU 但不轉發資料，不學習 MAC，用來避免 loop。
- Listening：開始參與 STP，接收/發送 BPDU，尚未轉發資料，也不學習 MAC。
- Learning：開始學習 MAC 位址表，但仍不轉發資料。
- Forwarding：正常運作，轉發資料也學習 MAC。

2 真好，又有新的 switch 可以玩了：）

1. 下方指令如果沒寫到 Switch2，則同樣指令要設定在 Switch2。（重複的都省略）

```
Switch(config)# hostname Switch1
```

2.
```
Switch1(config)# enable secret enable
```

3.
```
Switch1(config)# ip domain-name nasa.com
Switch1(config)# ip ssh version 2
```

4.
```
Switch1(config)# line vty 0 4
Switch1(config-line)# transport input ssh
Switch1(config-line)# login local
Switch1(config)# line vty 5 15
Switch1(config-line)# transport input none
```

5.
```
Switch1(config)#vlan 10
Switch1(config-vlan)#name VLAN10
Switch1(config)#vlan 20
Switch1(config-vlan)#name VLAN20
Switch1(config)#vlan 99
Switch1(config-vlan)#name VLAN99
```

6.
```
Switch1(config)#interface Fa0/1
Switch1(config-if)#switchport mode access
Switch1(config-if)#switchport access vlan 10
Switch1(config)#interface Fa0/2
Switch1(config-if)#switchport mode access
Switch1(config-if)#switchport access vlan 20
Switch1(config)#interface Fa0/3
Switch1(config-if)#switchport mode access
Switch1(config-if)#switchport access vlan 99
```

```
Switch2(config)#interface Fa0/4
Switch2(config-if)#switchport mode access
Switch2(config-if)#switchport access vlan 10
Switch2(config)#interface Fa0/5
Switch2(config-if)#switchport mode access
Switch2(config-if)#switchport access vlan 20
```

7.
```
Switch1(config)#interface range Gig0/1-2
Switch1(config-if-range)#switchport mode trunk
Switch1(config-if-range)#switchport trunk allowed vlan 10,20,99
Switch1(config-if-range)#channel-group 1 mode active

Switch1(config)#interface port-channel 1
Switch1(config-if)#switchport mode trunk
Switch1(config-if)#switchport trunk allowed vlan 10,20,99
```

8.
```
Switch1(config)# username admin privilege 15 secret nasa2025
```

3 你在 switch 上玩什麼！

1. SNMP（Simple Network Management Protocol）是一種用來監控、管理網路設備（如 routers、switches）的 application-layer protocol。他能夠取得設備的狀態資訊，例如 CPU、記憶體使用率、介面流量等。也能設定設備參數或接收設備異常通知。SNMPv2 和 SNMPv3 的主要差異是安全性。相較於 v2 直接用明文傳輸，v3 利用 SHA、AES 等提供認證及加密。

2. MIB (Management Information Base) 是一種 資料庫/結構化目錄，用來記錄網路設備可管理的資訊物件。在 SNMP 框架，SNMP manager 如果想查詢或設定某個設備資訊，必須知道該物件的 OID，這些都可以去 MIB 裡查。此外不同廠商設備只要都遵守標準 MIB，就能讓 SNMP manager 一致地讀取或管理。

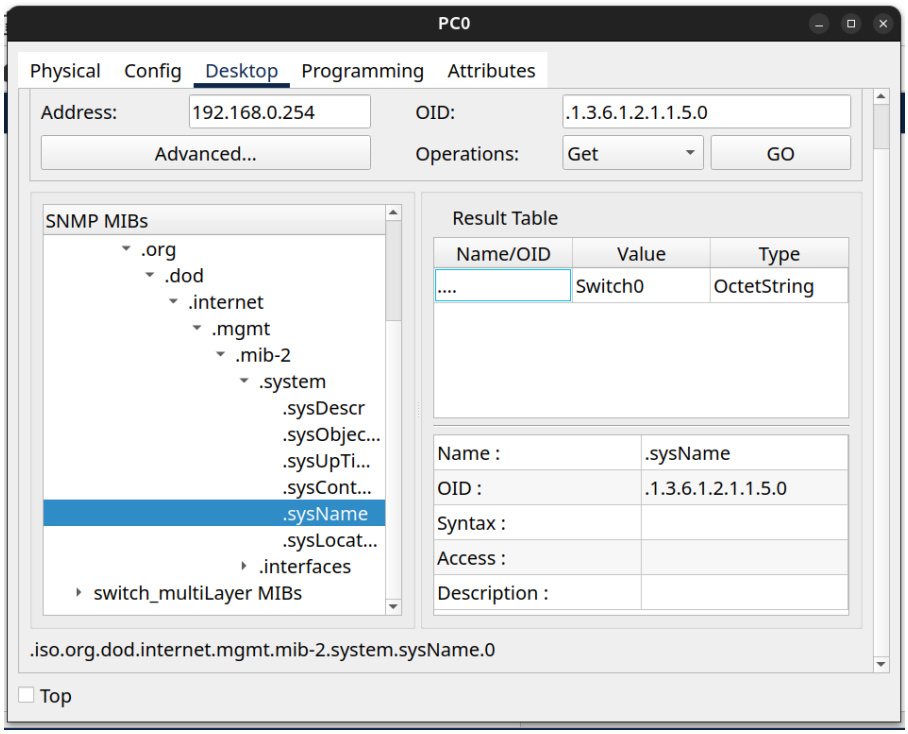
3. 因為異常流量會直接反映在介面流量上，能抓到 broadcast storm 或 DoS 攻擊的跡象。

   - `ifInOctets` / `ifOutOctets` 可以監控介面流量大小
   - `ifInErrors` / `ifOutErrors` 可以監控封包錯誤數
   - `ifInDiscards` / `ifOutDiscards` 可以監控被丟棄的封包
   - `ifInBroadcastPkts` 可以監控廣播封包量

   此外，也可以觀察 CPU 使用率，判斷是否有某個 process 消耗過高。

   - `cpmCPUTotal5min` 可以監控 CPU 平均使用率
   - `cpmCPUTotal1min` / `cpmCPUTotal5sec` 可以監控即時 CPU 使用狀況
   - `cpmProcessEntry` 可以監控各個 process 的 CPU 使用比例

4. 結果截圖如下：



用到的指令如下

```
Switch0(config)# snmp-server community public RO
Switch0# write memory
```