MikroTik RouterOS

腳本編寫和 WLAN 無線教程

Mikrotik 公司 1996 年成立於拉脫維亞，RouterOS 是 1997 年研發的專業路由系統，經歷了多年的發展，已經發展到 6.0 版本。最初 MikroTik 開發 RouterOS 目的是解決無線局域網傳輸問題（WLAN），後來通過不斷擴展功能實現了多種功能集于一身路由作業系統。在國內早期最大的使用人群是網吧、社區寬頻和企業網路管理，這個和國外的情況有點差別，在國外 RouterOS 不僅用於解決路由管理，大多應用在 WLAN 的覆蓋和傳輸，RouterOS 在基於 802.11abgn/ac 協議上的高頻寬傳輸有自己的明顯優勢，特別是他獨有的 Nstrem 和 Nv2 協議，近段時間國內的 WLAN 市場發展非常迅猛，對基於 RouterOS 開發的 RouterBOARD 產品需求也在不斷增長。其實當前的 RouterOS 已經具備 1Gbps 網路環境的完整解決方案。通過腳本可以讓 RouterOS 完成二次功能開發和應用，在設計上具備一定的開放性。

RouterOS 不管從功能，還是性能方面已經超過了許多中端路由器，隨著 RouterOS 在國內越來越多的人接受，從最開始的網吧多線路與流控和社區寬頻，到後來的 VPN 應用和企業網路管理，還是當前 RouterOS 的 WLAN 無線應用，都在不斷衝擊整個網路行業！在 2005 年後出現了幾款類似的軟體路由系統，雖然從個別方面比起 RouterOS 優越，但整體上仍然難以超越。

從 2003 年開始系統接觸和使用 RouterOS，多年的學習和工作中積累了大量的實際經驗，在這期間也得到廣大 RouterOS 愛好者的支持和幫助，也是大家相互學習的過程。本套課程通過各方面的資料整理後編輯而成，從最基礎的安裝、登陸配置、多線路策略和流量控制、到諸如 WLAN 配置、VPN 實現、腳本編寫等高級應用進行深入的講解，適合於所有使用或學習 RouterOS 的朋友，在此基礎上還有另外一套教程：《RouterOS 腳本編寫和 WLAN 無線教程》，以及 MikroTik 配套的監控軟體《The Dude 中文實用教程》。

從 RouterOS 6.0 版本開始，MikroTik 對 RouterOS 做大的改動，從內核優化、Queue 大改動、新的 Tilte 硬體構架等，核心改動較之前變化很大，性能提升明顯。但對於 x86 平臺的用戶感覺沒有 RouterBOARD 那麼明顯，因為後期的 RouterBOARD 加入了 FastPath 功能，能通過硬體快速轉發資料包，這個是 x86 平臺無法做到的，但從總體上內核的優化和 Queue 性能的改進也能帶來不小的提升。

《RouterOS 腳本編寫和 WLAN 無線教程》對 RouterOS 的腳本編寫和 WLAN 的基礎知識和常見問題做通俗的講解，通過 RouterOS 來介紹 WLAN 網路的常見的應用，如覆蓋、點對點、點對多點、中繼、WDS 和 Mesh 網路，能讓讀者對 MikroTik 產品 WLAN 應用和配置等得到充分瞭解，當然這些基本的 WLAN 無線知識也可以應用到其他產品中。

該手冊是提供對MikroTik RouterOS 嵌入式腳本介紹，主機腳本提供了自動維護路由器任務的功能，通過借助用戶自訂發生事件腳本。對於 RouterOS 的腳本操作，需要讀者有一定的程式設計知識，而且對 RouterOS 各個功能熟悉和掌握。

學習 RouterOS 還是希望大家具備一定的路由交換的基礎，這樣看更容易理解，網路的路由交換是任何網路工程師入門的基礎，因此我在教程裡開始加入了一些基礎知識，希望初步涉及網路的朋友能瞭解下，但這些基礎知識還不夠，需要大家自己去學習相關的路由交換內容，初級的網路工程師如 CCNA 或 HCNA 等，這些資料可以去看看和學習。

版本:	V6.6 e
適用於:	RouterOS v5.x v6.x
編寫:	余 松
版權申明:	該教程由本人經多年整理和編寫，請勿非法篡改或其他商業用途
網站	www.irouteros.com
E-mail:	athlon_sds@163.com
	內容如有更新，恕不通知!

目 錄

第一章 Scheduler（計畫任務）

Scheduler 計畫任務，通過設置定時或週期安排執行相應的腳本操作，計畫任務能有效完成預期的任務，說的高級點就是基於 Scheduler 和 Script 兩個功能，實現自己編寫自己的腳本，實現智慧路由器。

規格

功能包需求：*system*
等級需求：*Level1*
操作路徑：*/system scheduler*

1.1 計畫任務介紹

計畫任務列表通過調用腳本，並觸發腳本執行，在指定的時間或者是在指定的時間週期執行任務，在計畫任務調用腳本，可以設置腳本名稱或執行寫入腳本。

屬性描述

interval（*時間*; 默認：**0s**）- 腳本執行的間隔週期時間，腳本將在指定的時間週期內反復執行。
name（*名稱*） - 任務名稱
on-event（*名稱*） - 腳本執行名。通過調用**/system script** 裡的腳本規則名稱，也可以直接寫入腳本。
run-count（*唯讀: 整型*） - 監視腳本使用數，這個計數器記錄當每個腳本執行一次，計數器便增加 1
start-date（*日期*） - 開始腳本執行的日期
start-time（*時間*） - 開始腳本執行的時間
startup - 預設在系統啟動 3 秒後執行腳本. 計畫任務選項裡對 start-time 設置了 startup，則在系統啟動完成後 3 秒運行。

Run-count 記錄了計畫任務的執行次數，當路由器重啟後，計數器會重置，如果有複雜的腳本執行模式，通常可能會涉及到計畫多個腳本，在多個計畫任務中切換，執行一個時禁用另外一個。

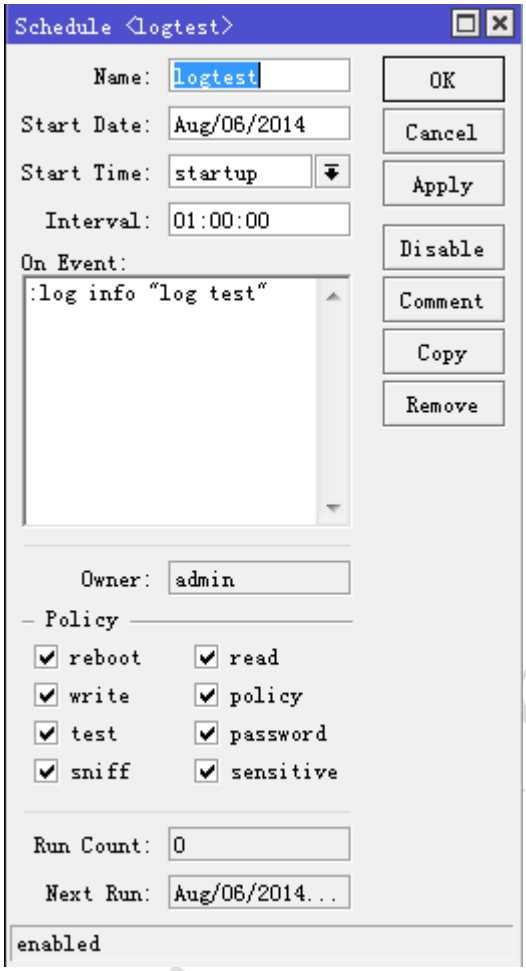
1.2 計畫任務事例

通過下面的簡單事例，介紹下計畫任務在 RouterOS 是如何操作執行的。

事例 1：我們添加一個任務執行系統日誌記錄測試，並間隔 1 小時執行一次，在 log 日誌中顯示“log test”，on-event 我可以直接寫入 RouterOS 腳本

```
[admin@MikroTik] system script> add name=logtest on-event=:log info “test” start-time=startup
interval=1h
[admin@MikroTik] /system scheduler> print detail
Flags: X - disabled
 0   name="logtest" start-time=startup interval=1h on-event=:log info "log test" owner="admin"
policy=ftp,reboot,read,write,policy,test,winbox,password,sniff,sensitive,
       api run-count=5 next-run=09:05:19
```

在 winbox 中，將腳本直接配置到 Schedule 的 On-event 下：



事例 2：添加 2 個腳本改變頻寬設置佇列規則”ABC”，每天上午 9 點限制為 10Mb/s，下午 5 點限制為 20Mb/s。這個佇列的規則。

先在 queue simple 添加一條對內網 ip 段 192.168.0.0/24 的流控規則(6.0 版本前配置流控目標 IP 為 target-addresses，6.0 開始為 target)：

```
[admin@MikroTik] /queue simple> add name=ABC target=192.168.0.0/24 max-limit=10M/10M
[admin@MikroTik] /queue simple> print value-list
             name: queue1
             target: 192.168.0.0/24
             parent: none
      packet-marks:
             priority: 8/8
              queue: default-small/default-small
             limit-at: 0/0
          max-limit: 10M/10M
         burst-limit: 0/0
    burst-threshold: 0/0
         burst-time: 0s/0s
        bucket-size: 0.1/0.1
```

然後配置腳本，並添加計畫任務(**注：**在 2.9 之前定義腳本查找字串是不需要加雙引號的，但在 3.0 後中需要注明字串，需加上雙引號，如查找流控規則名”ABC”)，進入 script 腳本編輯器，添加兩台腳本規則 start_limit 和 stop_limit

```
[admin@MikroTik] queue simple> /system script
[admin@MikroTik] system script> add name=start_limit source={/queue simple set [find
name=”ABC”] max-limit=10M/10M }
[admin@MikroTik] system script> add name=stop_limit source={/queue simple set [find name=”ABC”]
max-limit=20M/20M }
[admin@MikroTik] system script> print
  0 name="start_limit" source="/queue simple set [find name=”ABC”] max-limit=10M/10M "
    owner=admin run-count=0

  1 name="stop_limit" source="/queue simple set [find name=”ABC”] max-limit=20M/20M "
    owner=admin run-count=0
```

進入計畫任務下使用 on-event 調用腳本編輯器中的兩條腳本，可以直接在 on-event 中填寫腳本名稱

```
[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add interval=24h name="set-10M" \
\... start-time=9:00:00 on-event=start_limit
[admin@MikroTik] system scheduler> add interval=24h name="set-20M" \
\... start-time=17:00:00 on-event=stop_limit
[admin@MikroTik] system scheduler> print
Flags: X - disabled
  #   NAME      ON-EVENT  START-DATE  START-TIME INTERVAL            RUN-COUNT
  0   set-10M   start... oct/30/2008 09:00:00   1d                  0
  1   set-20M  stop_... oct/30/2008 17:00:00   1d                  0
[admin@MikroTik] system scheduler>
```

事例 3：下面是通過電子郵件發送每週備份路由器配置的腳本：

```
[admin@MikroTik] system script> add name=e-backup source={/system backup
save name=email; /tool e-mail send to="root@host.com" subject=([/system
{... identity get name] . " Backup") file=email.backup}
[admin@MikroTik] system script> print
  0 name="e-backup" source="/system backup save name=ema... owner=admin
    run-count=0

[admin@MikroTik] system script> .. scheduler
[admin@MikroTik] system scheduler> add interval=7d name="email-backup" \
\... on-event=e-backup
[admin@MikroTik] system scheduler> print
Flags: X - disabled
  #   NAME      ON-EVENT START-DATE  START-TIME INTERVAL            RUN-COUNT
  0   email-... e-backup oct/30/2008 15:19:28   7d                  1
[admin@MikroTik] system scheduler>
```

用 RouterOS 電子郵件發送，需要對你接收郵箱設置，即開啟接收郵箱的 SMTP 服務，操作路徑/tool e-mail 例如（**注：**建議是自己的 SMTP 伺服器，一些正規網站的郵件伺服器可能會將發送資訊遮罩）：

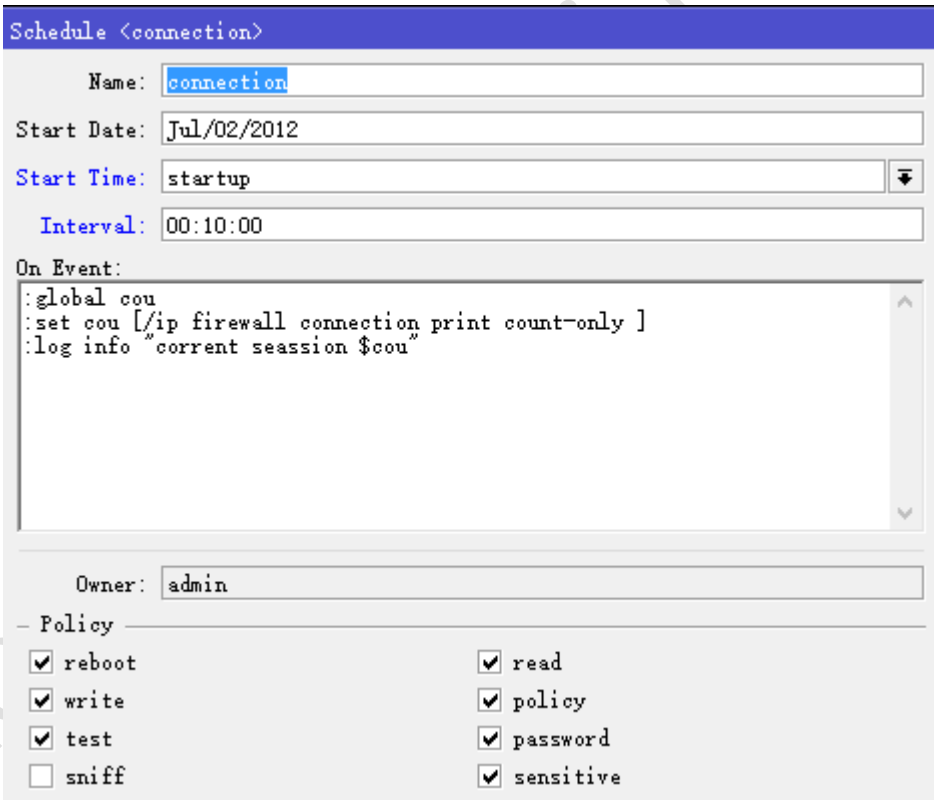
```
[admin@MikroTik] tool e-mail> set server=159.148.147.198 from=SysAdmin@host.com
[admin@MikroTik] tool e-mail> print
    server: 159.148.147.198
      from: SysAdmin@host.com
[admin@MikroTik] tool e-mail>
```

事例 4：下面的例子每 10 分鐘統計一次 firewall connection 的會話數，並記錄在 log info 裡:

首先我們編寫腳本，獲取會話數量，並通過:log info 命令記錄當前會話數

```
:global cou
:set cou [/ip firewall connection print count-only ]
:log info "corrent seassion $cou"
```

進入 schedule 新建一個計畫任務取名 connection，設置 interval 為 10 分鐘，On-event 加入腳本



以上的計畫任務都涉及了 RouterOS 的 Script 腳本編輯，所以 RouterOS 要完成指定的計畫任務，會編寫腳本是必須的。

第二章 RouterOS Script（腳本）

在 RouterOS 中一個腳本配置構成由控制命令和運算式(ICE - internal console expression 內部控制台運算式)。

RouterOS 操作命令，例如：/ip firewall filter add chain=forward protocol=gre action=drop 這個是描述在防火牆中過濾 gre 協議的操作，即通過“/”來達到命令執行的目的。在腳本 ICE 運算式首碼需要用 “：”並能在任何目錄路徑下操作。

一個事件（event）用來觸發腳本執行，在 RouterOS 下包括：System Scheduler, Traffic Monitoring Tool, Netwatch Tool 。

功能包需求: *system*
操作路徑: */system script*

注： RouterOS2.9 本版與 RouterOS 3.0 的腳本有一定的區別：

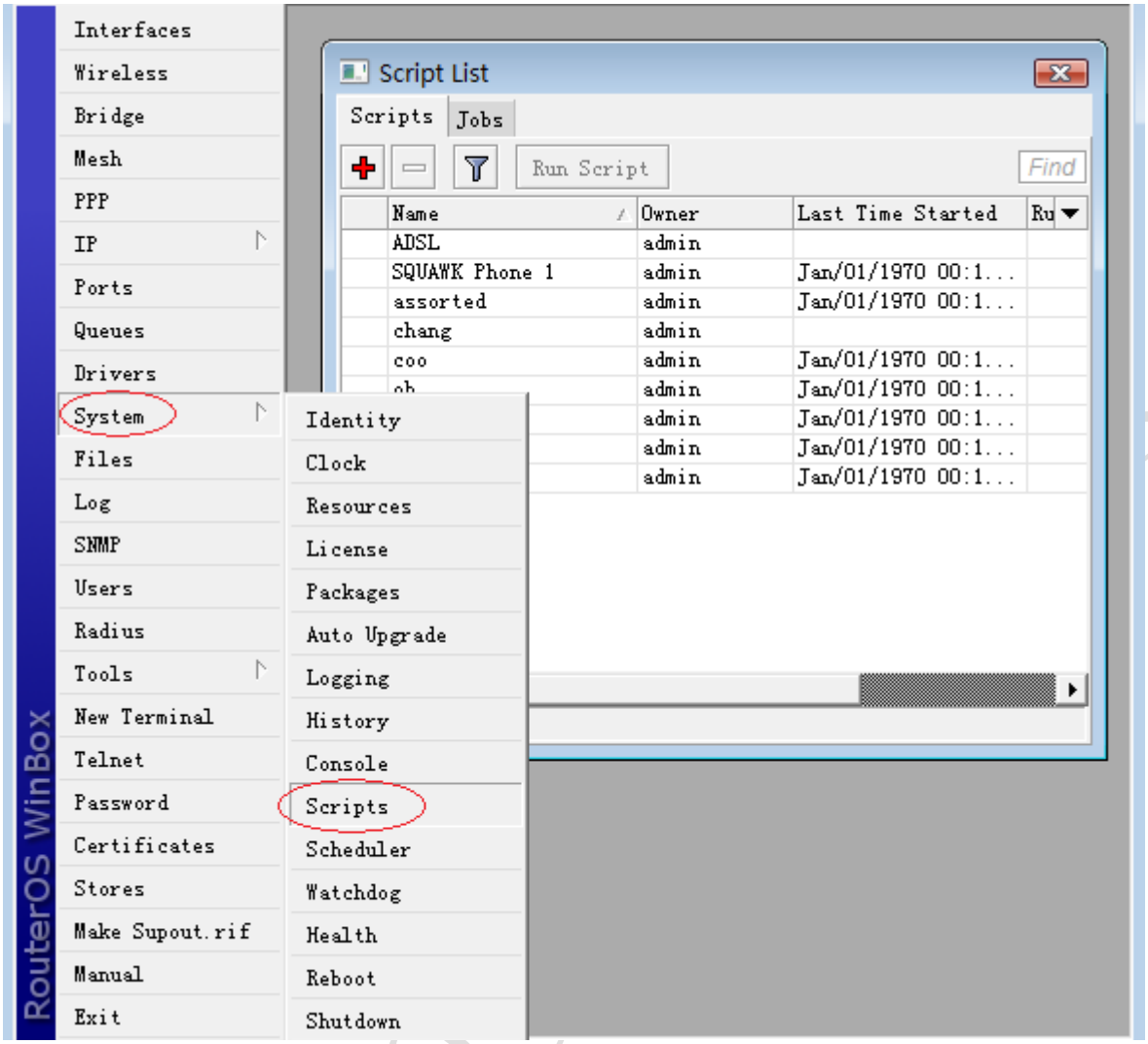
1、RouterOS3.0 字元參數後需要使用引號注明
如 comment=”test”；name=”pppoe-out1”；:set i $”tx-current”

2、RouterOS3.0 的變數定義不支援“中橫杠”的定義
如 :global test-address 這樣定義在 3.0 和 4.0 中是非法的

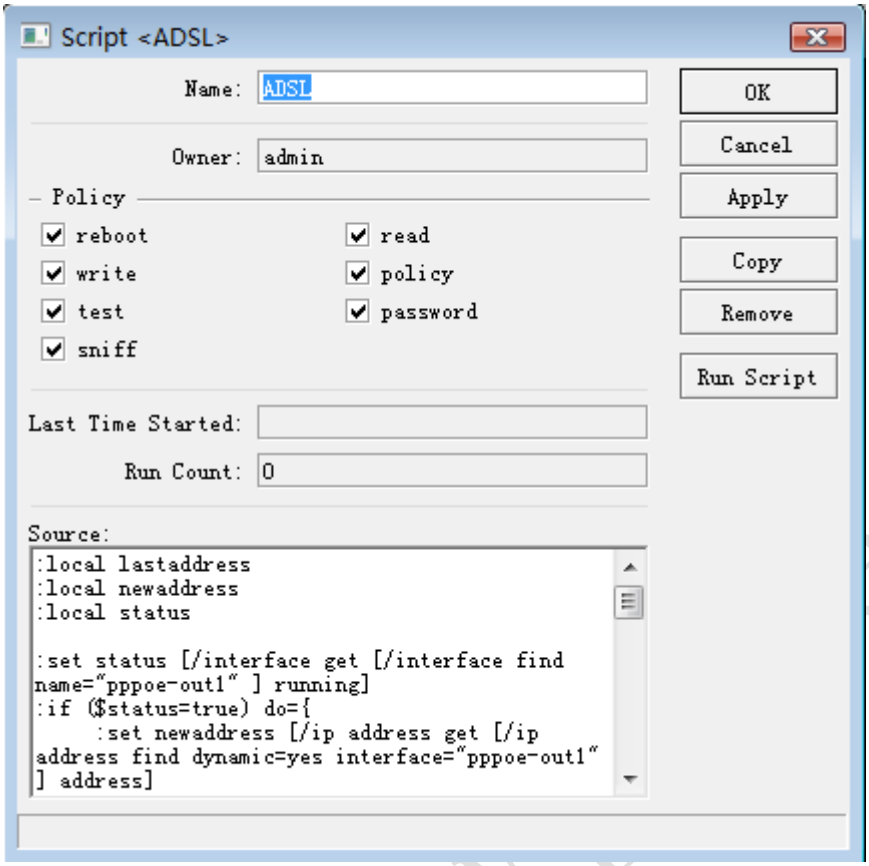
2.1 RouterOS 腳本基本操作

操作路徑：/system script

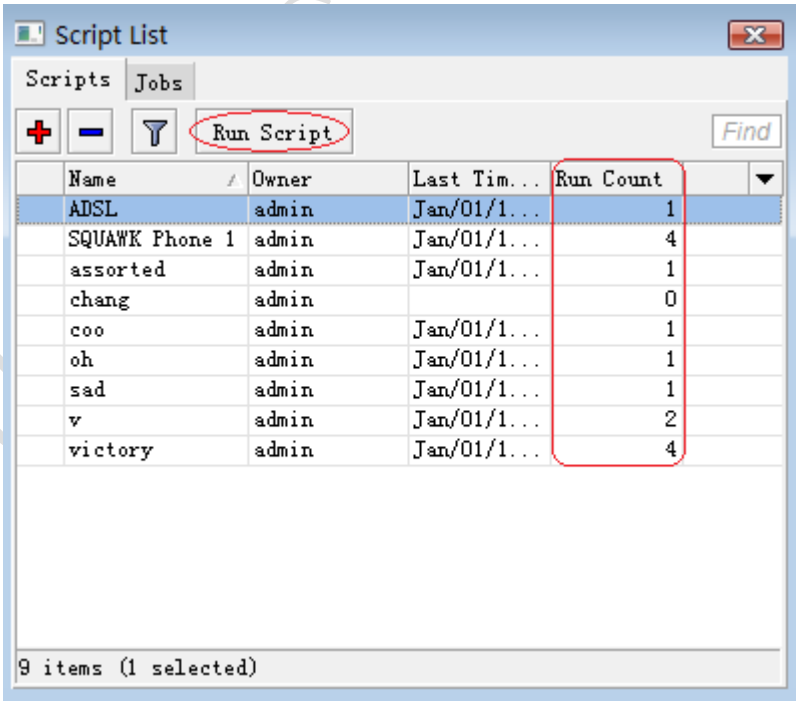
腳本編寫進入 script 目錄下，在 script 中我們可以定義多條腳本規則,如下圖：



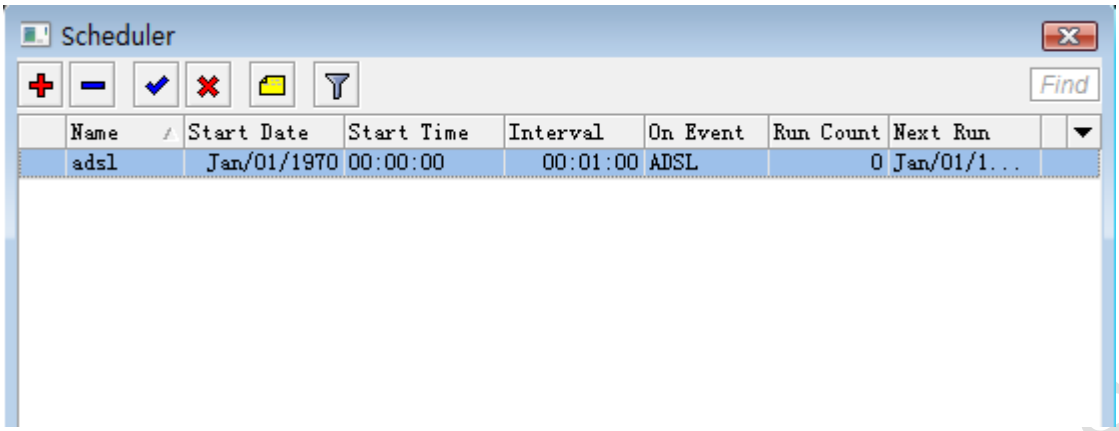
我通過 script 編輯器編輯腳本：



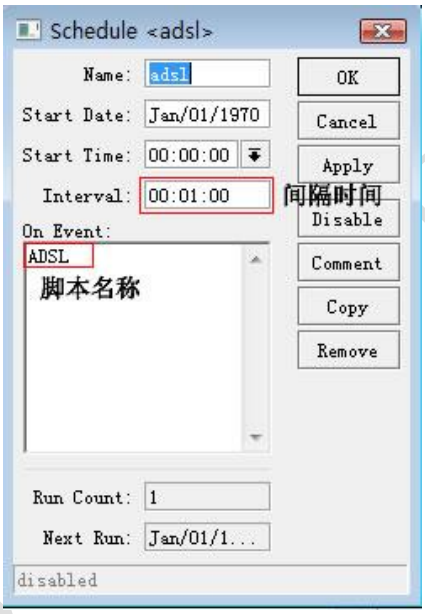
我們可以通過 Run Script 命令運行當前的腳本，在 Run Count 中會紀錄腳本運行的次數：



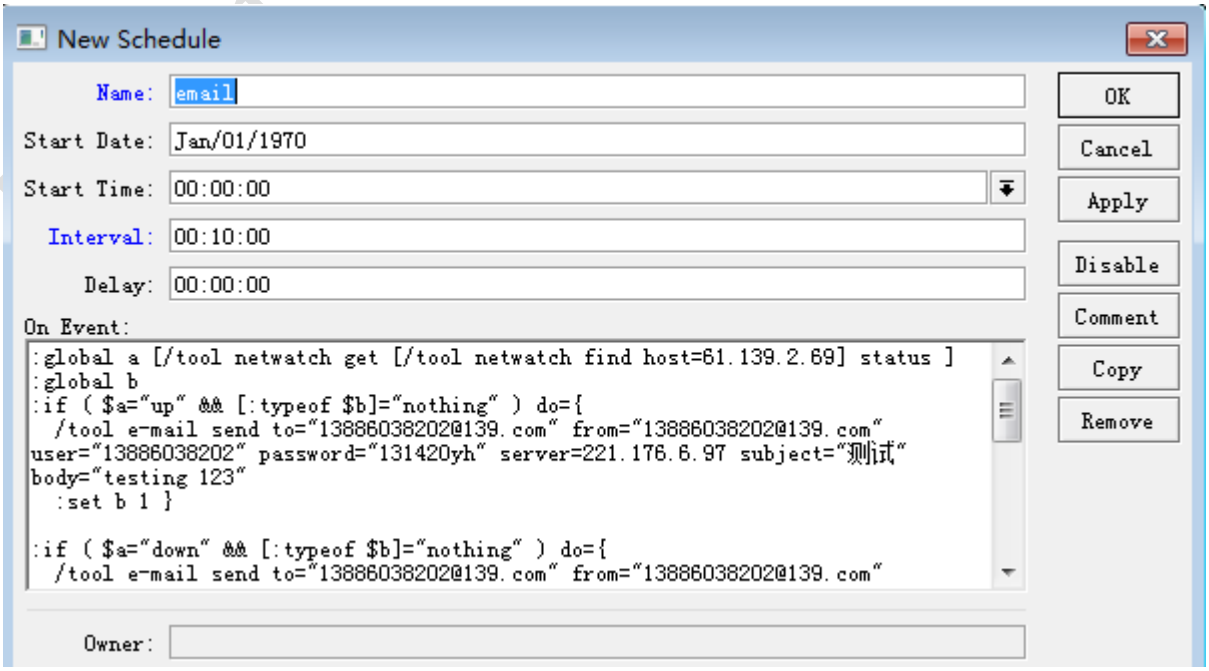
在 Script 中我們編輯好腳本後，我們可以通過 RouterOS 相應的功能調用腳本，並執行這些腳本，如/system scheduler（計畫任務）、/tool 中的 netwatch、traffic-monitor 等。通常執行腳本在 scheduler 計畫任務中最常用，如這裡我們使用了 ADSL 腳本，需要每間隔 1 分鐘執行一次，我們可以通過 scheduler 來完成，通過/system scheduler 進入計畫任務目錄：



添加一個名稱為 adsl 的計畫任務規則，設置 Interval 時間 1 分鐘，On Event 中添加腳本名稱：



注：我們也可以在 on event 中直接輸入腳本的命令：



2.2 RouterOS 腳本語法

RouterOS 腳本被劃分為多個命令列，命令列是一個接一個單元執行直到腳本結束或者直到 錯誤發生。

命令列結構

RouterOS 控制台是使用下面的命令語法：

[首碼] [路徑] 命令 [未具名引數] [參數=[值]] .. [參數=[值]]

- **[首碼]** - 如果命令是 ICE 或者路徑通過":" 或者 "/" 字元表示
- **[路徑]** - 得到操作功能表的路徑
- **命令** - 一個命令獲取在指定的功能表路徑下
- **[未具名引數]** - 即事先定義參數，如果命令需要必須指定該參數
- **[參數]** - 按先後順序各自訂值

命令列結束以“;”標識為代表或者換行，在結束命令列有時不需要“;”或者換行

獨立的命令包含 (), [] 或者 {}不需要任何的結尾命令字符，命令結尾取決於腳本的內容

```
:if ( true ) do={  :put "lala" }
```

每條命令列包含其他命令列，起始通過方括號定義"[ ]"

```
:put [/ip route get [find gateway=1.1.1.1]];
```

注意，上面這條代碼包含 3 條命令列：

- :put
- /ip route get
- find gateway=1.1.1.1

命令列能通過多余一個行的方式建立，可以查看後面的行連接規則. 幾種常見的命令實例

- **Prefix(首碼)** - 指示那一個命令到一個 ICE, 如: 腳本:put 或者命令部分從根目錄下執行, 如 “ / ”

```
[admin@MikroTik] ip firewall mangle> /ping 10.0.0.1
```

- **Path（路徑）** - 希望到達目錄的一個相關路徑，如：.. filter

```
[admin@MikroTik] ip firewall mangle> .. filter print
```

- **Action（執行）** - 在指定的目錄下一個可操作的執行命令，如：add

```
[admin@MikroTik] ip firewall mangle> /ip firewall filter add chain=forward action=drop
```

- **unnamed parameter（無名參數）**- 需要通過一些執行和輸入固定格式在命令後的執行名稱，如 10.0.0.1

```
[admin@MikroTik] ip firewall mangle> /ping 10.0.0.1
```

- **name[=value]（參數值）**- 一個跟在參數名後的各自的值，如：**ssid=myssid**

```
/interface wireless set wlan1 ssid=myssid
```

事例

在下面的例子中解釋了控制台內的部分命令

Ping 命令操作

```
/ping 10.0.0.1 count=5
```

命令分解

首碼	/
執行	ping
未具名引數	10.0.0.1
name[=值]	count=5

For 迴圈

```
:for i from=1 to=10 do={:put $i}
```

命令分解：

首碼	:
執行	for
未具名引數	i
name[=值]	from=1 to=10 do={:put $i}

多網卡流量查看

```
/interface monitor-traffic ether1,ether2,ipip1
```

命令分解：

首碼	/
路徑	interface
執行	monitor-traffic
未具名引數	ether1,ether2,ipip1

變數

RouterOS 指令碼語言支援兩種類型的變數，global（系統變數）和 local（僅當前腳本運行的變數）取變數值使用’$’標記符號，但除了 set 和 unset 後面不需要’ $’標記外，其他的都需要使用該標記。一個變數必須在腳本中首先被聲明，下面有四種類型的變數：

- **全域變數** – 使用 global 關鍵字定義，全域變數可用被所有腳本和通過控制台登陸到的同一台路由器調用。注意，重啟後全域變數無法保存。
- **本地變數** – 使用 local 關鍵字定義，本地變數不能和其他任何腳本或其他控制台登陸的共用。本地變數值會隨腳本執行完成而丟失。
- **迴圈變數** – 在 for 和 foreach 內部定義，這裡的變數僅能使用在 do 命令塊中，在命令執行完成後將被刪除掉
- **監聽變數** – 一些 monitor 命令在 do 中能插入變數或控制命令額。

你分配一個新的變數值使用:set 命令，並定義兩個為命名的參數：變數名稱和新的變數值。如果一個變數不需要長時間被調用，可以通過:unset 命令釋放變數。如果釋放一個本地變數，該值會清空。如果你釋放一個全域變數，該值仍然會保存在路由器中，但是在當前腳本無法調用。

迴圈變數會影響到前面已經聲明過的同樣名稱的變數。

事例

```
[admin@MikroTik] ip route> /
[admin@MikroTik] > :global g1 "this is global variable"
[admin@MikroTik] > :put $g1
this is global variable
[admin@MikroTik] >
```

注釋

一個注釋從“#”號字元開始執行，並結束在一行的結尾，空格或者任何其他標示不允許在#標示之前。如果“#”字元出現在一個字串中將不會考慮為一個注釋內容。

```
# this is a comment （正確注釋）
 # bad comment    （錯誤注釋）
:global a; # bad comment   （錯誤注釋）

:global myStr "lala # this is not a comment"    （不是注釋）
```

行連接

通過“\”字元可以將兩行或者多行連接到一個邏輯行。一行以反斜線結束不能進行注釋，即一個反斜線不能連接一個注釋。一個字串不會繼續一個指令。

```
:if ($a = true \
      and $b=false) do={  :put  “$a $b”; }
```

```
:if ($a = true \      # bad comment  （錯誤注釋）
      and $b=false) do={  :put “$a $b”; }

# comment \
    continued  -  invalid  (語法錯誤)
```

指令之間的空格

空格可以用於分隔指令。空格必須在兩個指令之間僅在他們一系列互相關聯的事物解釋為一個不通過的指令，例如：

```
{
    :local a true;  :local b false;
# 空格不需要
    :put (a&&b);
# 空格需要
    :put (a and b);
}
```

空格不允許

- 在 ’<參數>=’ 之間不允許
- 在’from=’ ’to=’ ’step=’ ’in=’ ’do=’ ’else=’ 之間不允許

```
#錯誤:
:for i from = 1 to = 2 do = {  :put $i }

#正確:
:for i from=1 to=2 do={  :put $i }
:for i from= 1 to= 2 do={  :put $i }

#錯誤
/ip route add gateway = 3.3.3.3

#正確
/ip route add gateway=3.3.3.3
```

關鍵字

下面的字元是關鍵字，不能用作變數和功能名稱：

```
and       or        not       in
```

分隔符號

下面記號作為分隔符號的語法：

```
()  []  {}   :    ;   $   /
```

資料類型

RouterOS 指令碼語言有以下資料類型：

類型	屬性
number	- 64bit 整型，支援十六進位輸入；
boolean	- 布林型，真與假；
string	- 字元型；
IP	- IP 地址；
内部 ID	- 十六進位，首碼通過‘*’標記。每個功能表目錄下都被分配唯一的數字——内部 ID；
time	- 日期與時間值；
array	- 一個陣列有序的值；
nil	- 如果沒有值被分配，變數的預設值

命令替換和返回值

一些終端命令是非常有用的，如他們可以輸出一個變數值給其他命令。在 RouterOS 終端控制中通過命令得到返回值。返回值不會被顯示出來。從一個命令中得到返回值，應包含在“[ ]”括弧中。之前執行返回值的命令所得到的值包含在括弧中，這個稱為命令替換。

命令產生的返回值，但不限制 **find**，返回一個參考特殊專案 **ping**，返回 ping 成功的數目， **time**，返回測量時間長度值，**incr** 和 **decr**，返回新的變數值，**add**，返回內部最新建立專案編號

find 命令的使用方法：

```
[admin@MikroTik] > /interface
[admin@MikroTik] interface> find type=ether
[admin@MikroTik] interface>
[admin@MikroTik] interface> :put [find type=ether]
*1,*2
[admin@MikroTik] interface>
```

這個方式你能看到內部控制台的項目編號。自然的，你能使用他們到其他的命令的操作中：

```
[admin@MikroTik] interface> enable [find type=ether]
[admin@MikroTik] interface>
```

運算子

RouterOS 控制台能對數值、時間值、IP 位址、字串和表等做簡單的運算。從一個運算式中得到結果。

命令描述

- - 一元減法。對一個數值做反運算。
- - 二進位減，扣除兩個數值、兩個時間值、兩個 IP 地址或 IP 地址和其數值。
! - 邏輯非（NOT）。
/ - 除法運算子。
. - 連接子，連接兩個字串或拼接一個表到其他表上或拼接一個元素給一個表。
^ - 位運算移（XOR）。
~ - 按位反， which inverts bits in IP address
* - 乘法運算子。
& - 位運算與（AND）
&& - 邏輯與（AND）
+ - 加法運算子。對兩個數值、兩個時間值或 IP 地址做加法運算。
< - 小於符。返回值為布林型。
<< - 左移運算子。
<= - 小於等於符，返回值為布林型。
> - 大於符。返回值為布林型。
>= - 大於等於符，返回值為布林型。
>> - 右移運算子。
| - 位運算或（OR）
|| - 邏輯或（OR），返回值為布林型。
-> - 通過關鍵字獲取陣列成員

當比較兩個陣列的時注意：如果他們各自元素是相等的，那麼兩個陣列即相等。

運算子的優先順序和求值命令

```
[admin@MikroTik] ip firewall rule forward> :put (10+1-6*2=11-12=2+(-3)=-1)
false
[admin@MikroTik] ip firewall rule forward> :put (10+1-6*2=11-12=(2+(-3)=-1))
true
[admin@MikroTik] ip firewall rule forward
```

邏輯非（NOT）

```
[admin@MikroTik] interface> :put (!true)
false
[admin@MikroTik] interface> :put (!(2>3))
true
[admin@MikroTik] interface>
```

邏輯運算

```
[admin@MikroTik] interface> :put (-1<0)
true
[admin@MikroTik] >
1
```

按位反

```
[admin@MikroTik] interface> :put (~255.255.0.0)
0.0.255.255
[admin@MikroTik] interface>
```

加法運算

```
[admin@MikroTik] interface> :put (3ms + 5s)
00:00:05.003
[admin@MikroTik] interface> :put (10.0.0.15 + 0.0.10.0)
cannot add ip address to ip address
[admin@MikroTik] interface> :put (10.0.0.15 + 10)
10.0.0.25
[admin@MikroTik] interface>
```

減法運算

```
[admin@MikroTik] interface> :put (15 - 10)
5
[admin@MikroTik] interface> :put (10.0.0.15 - 10.0.0.3)
12
[admin@MikroTik] interface> :put (10.0.0.15 - 12)
10.0.0.3
[admin@MikroTik] interface> :put (15h - 2s)
14:59:58
[admin@MikroTik] interface>
```

乘法運算

```
[admin@MikroTik] interface> :put (12s * 4)
00:00:48
[admin@MikroTik] interface> :put (-5 * -2)
10
[admin@MikroTik] interface>
```

除法運算

```
[admin@MikroTik] interface> :put (10s / 3)
00:00:03.333
[admin@MikroTik] interface> :put (5 / 2)
2
[admin@MikroTik] interface>
[admin@MikroTik] > :put (0:0.10 / 3)
00:00:02
[admin@MikroTik] >
```

各種邏輯比較運算

```
[admin@MikroTik] interface> :put (10.0.2.3<=2.0.3.10)
```

```
false
[admin@MikroTik] interface> :put (100000s>27h)
true
[admin@MikroTik] interface> :put (60s,1d!=1m,3600s)
true
[admin@MikroTik] interface> :put (bridge=routing)
false
[admin@MikroTik] interface> :put (yes=false)
false
[admin@MikroTik] interface> :put (true=aye)
false
[admin@MikroTik] interface>
```

邏輯與和或運算

```
[admin@MikroTik] interface> :put ((yes && yes) || (yes && no))
true
[admin@MikroTik] interface> :put ((no || no) && (no || yes))
false
[admin@MikroTik] interface>
```

按位與、或、異或運算

```
[admin@MikroTik] interface> :put (10.16.0.134 & ~255.255.255.0)
0.0.0.134
[admin@MikroTik] interface>
```

移位運算

```
[admin@MikroTik] interface> :put (~((0.0.0.1 << 7) - 1))
255.255.255.128
[admin@MikroTik] interface>
```

連接運算子

```
[admin@MikroTik] interface> :put (1 . 3)
13
[admin@MikroTik] interface> :put (1,2 . 3)
1,2,3
[admin@MikroTik] interface> :put (1 . 3,4)
13,4
[admin@MikroTik] interface> :put (1,2 . 3,4)
1,2,3,4
[admin@MikroTik] interface> :put ((1 . 3) + 1)
14
[admin@MikroTik] interface>
```

資料類型

RouterOS 區分幾種資料類型，字元型、布林型、數值型、時間型、IP 地址、內碼和列表。 RouterOS 首先會試著將任何值轉換為制定的類型。

內部指令碼語言彌補了特殊函數之間類型轉換的不足。通過內部的 **toarray**, **tobool**, **toid**, **toip**, **tonum**, **tostr** 和 **totime** 函數每個值轉換到相應的列表（**list**）中，對應為：**boolean**, **internal number**, **IP address**, **number**, **string** 或 **time**.

數位類型在內部表示為 64 位元帶符號的整型， 因此一個數位類型值變數可用長度從-9223372036854775808 到 9223372036854775807. 同樣可用輸入十六進位的數值，在前面加入 **0x** ，例如：

```
[admin@MikroTik] > :global MyVar 0x10
[admin@MikroTik] > :put $MyVar
16
[admin@MikroTik] >
```

列表通過逗號來區分值的次序，在空白出使用逗號間隔方式部推薦使用，因為這會讓控制終端無法識別字元的邊境。

Boolean 型的值為 **true** 或 **false**. 控制終端判斷 **true** 為“**yes**”，**false** 為“**no**”。

時間間隔可以輸入 HH:MM:SS 例如：

```
[admin@MikroTik] > :put 01:12:1.01
01:12:01.010
[admin@MikroTik] >
```

或者通過累計數位，具體指明單位時間的標記(**d** 對應 days, **h** 對應 hours, **m** 對應 minutes, **s** 對應 seconds, 以及 **ms** 對應 milliseconds)例如：

```
[admin@MikroTik] > :put 2d11h12
2d11:00:12
[admin@MikroTik] >
```

時間單位：

- **d, day, days** - 一天，或者 24 小時
- **h, hour, hours** - 一小時
- **m, min** - 一分鐘
- **s** - 一秒
- **ms** - 一毫秒，等同 0.001 秒

控制終端同樣接受時間為小數點的形式：

```
[admin@MikroTik] > :put 0.1day1.2s
02:24:01.200
[admin@MikroTik] >
```

運算元組

RouterOS v6 支援對陣列成員定義關鍵字 key，通過”->”方式獲取，操作如下：

```
[admin@x86] >:global aaa {a=1;b=2}
[admin@x86] > :put ($aaa->"a")
1
[admin@x86] > :put ($aaa->"b")
```

注意：陣列元素定義關鍵 key，與小寫字母的字元有區別，陣列關鍵字 key 必須放在引號中，例如：

```
[admin@ce0] > {:local a { "aX"=1  ; ay=2 };  :put ($a->"aX")}

1
```

陣列元素值輸出

Foreach 命令能通過依次查詢陣列元素和關鍵 key，並輸出：

```
[admin@ce0] >  :foreach k,v in={2; "aX"=1  ; y=2; 5} do={:put ("$k=$v")}

0=2
1=5
aX=1
y=2
```

當 foreach 命令使用一個值，元素會逐一返回 ：

```
[admin@ce0] >  :foreach k in={2; y=2; "aX"=1  ; 5} do={:put ("$k")}

2
5
1
2
```

上面的輸出順序大家都看到了，感覺有的奇怪，因為陣列會首先查詢普通元素，然後再是關鍵 key，關鍵 key 輸出時是按照字母循序執行，因此我們看到普通元素首先被輸出，然後是關鍵 key，關鍵 key 又按照字母順序輸出結果，

修改單個陣列元素值

```
[admin@MikroTik] >  :global a {x=1; y=2}
[admin@MikroTik] >  :set ($a->"x") 5
[admin@MikroTik] >  :environment print
a={x=5; y=2}
```

命令參考文檔

RouterOS 有多個嵌入式的控制終端命令和運算式 ICE 不依賴於當前操作目錄。這些命令不能直接改變配置，但他們可以做日常的維護工作。所有 ICE 可以通過在操作符輸入“ ： ”後敲擊“ ？ ” 顯示出。例如：

```
[admin@MikroTik] > :
environment  do       for       led     nothing  resolve  tobool   tonum   while
terminal     error    foreach   len     parse    set      toid     tostr
beep          execute global    local   pick     time     toip     totime
delay         find     if        log     put       toarray   toip6   typeof
```

常用命令屬性

beep - 通過 PC 內置的蜂鳴器或者揚聲器發出一個指定 length （時間長度）的 frequency Hz（頻率 Hz）.

輸入參數

- frequency (整型; 默認: 1000) - 信號頻率大小用單位 Hz
- length (時間; 默認: 100ms) - 信號長度

```
[admin@MikroTik] > :beep length=2s frequency=10000

[admin@MikroTik] >
```

delay - 在一個給定的時間長度不做任何操作

輸入參數

- delay-time (時間) - 等待的時間長度
- omitted - 無限制延遲

do - 根據條件執行命令直到獲取一個適當的值。如果沒有參數獲取，do 只執行有效操作一次，其中不會有什麼作用。 如果一邏輯條件被指定到 while 參數種,將會在命令執行後作判斷，在該條件判斷中為 true， do 語句會被一次一次的執行直到滿足 false 條件， if 參數，在做後面語句的任何操作時判斷一次，如果 false 不會執行任何的操作

輸入參數

- unnamed (文本) - 執行的操作

```
[admin@MikroTik] > {:global i 10; :do {:put $i; :set i ($i - 1);} \
\... while (($i < 11) && ($i > 0)); :unset i;}
10
9
8
7
6
5
4
3
2
1
[admin@MikroTik] >
```

environment print – 顯示關於當前變數的初始化情況。所有在系統中的全域變數 global variables 被以標題為 Global Variables 下列出。所有變數插入當前腳本（通過:local、通過:for 或者:foreach）被以標題為 Local Variables 下列出。

創建變數並顯示出他們的清單：

```
[admin@MikroTik] > :local A "This is a local variable"
[admin@MikroTik] > :global B "This is a global one"
[admin@MikroTik] > :environment print
Global Variables
B=This is a global one
Local Variables
A=This is a local variable
[admin@MikroTik] >
```

find – 查找字串內的一個字元或者一個元素在專案內的值，根據變數類型並返回一個變數的位置。

輸入參數

- unnamed(文本 | 清單) – 搜索字元或者字元清單值，並執行相關的操作
- unnamed(文本) – 字元的搜索
- unnamed（整型） – 搜索的起始位置，或搜索到目標返回的位置

```
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333"  ]
0
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333" 3  ]
1
[admin@MikroTik] interface pppoe-server> :put [:find "13sdf1sdfss1sfsdf324333" 3  3]
17
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3" 3 ]
4
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3" 3 3]
4
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3" 3 4]
5
[admin@MikroTik] interface pppoe-server> :put [:find "1,1,1,2,3,3,4,5,6,7,8,9,0,1,2,3" 3 5]
15
[admin@MikroTik]
```

for – 執行所給定的數次反復迴圈的命令，通過 from 和 to 設置起始和結算參數。

輸入參數

- unnamed（名稱） – 定義迴圈計數器變數名稱。
- from（整型） – 迴圈起始的變數值
- to（整型） – 迴圈結束的變數值
- step（整型; 默認: 1） – 遞增變數. 在迴圈起始到結束中間每迴圈一次的間距變數
- do（文本） – 執行包含在內的命令

```
[admin@MikroTik] > :for i from=1 to=100 step=37  do={:put ($i . " - " . 1000/$i)}
1 - 1000
38 - 26
75 - 13
[admin@MikroTik] >
```

foreach – 執行所提供在清單中的每一個元素

輸入參數

- unnamed (名稱) – 定義迴圈計數器變數名稱。
- in (陣列清單) – 陣列清單範圍或路徑
- do (text) – 執行包含在內的命令

顯示出一個 interface 中獲得列表各自的 IP 位址

```
:foreach i in=[/interface find type=ether ] \
\... do={:put ("+--" . [/interface get $i name]); \
\... :foreach j in=[/ip address find interface=$i]
\... do={:put ("|  `--" . [/ip address get $j address])}}
+--ether1
|  `--1.1.1.3/24
|  `--192.168.50.1/24
|  `--10.0.0.2/24
+--ether2
|  `--10.10.0.2/24
[admin@MikroTik] >
```

global – 聲明全域變數

輸入參數

- unnamed(名稱) – 變數名稱
- unnamed(文本) – 值, 分配給變數的內容

```
[admin@MikroTik] > :global MyString "This is a string"
[admin@MikroTik] > :global IPAddr 10.0.0.1
[admin@MikroTik] > :global time   0:10
[admin@MikroTik] > :environment print
Global Variables
IPAddr=10.0.0.1
time=00:10:00
MyString=This is a string
Local Variables
[admin@MikroTik] >
```

if – 條件陳述式. 如果一個邏輯判斷為真，這時執行 do 分區塊中的命令，否則選擇 else 分區塊中的執行。

輸入參數

- unnamed(yes | no) – 邏輯條件陳述式，在執行之後聲明內容前判斷一次
- do(文本) – 如果 if 語句判斷為真，在這個分區塊的命令會被執行。
- else(文本) -如果 if 語句判斷為假，在這個分區塊的命令會被執行。

通過 if 語句檢查 firewall 中是否有任何規則被添加

```
[admin@MikroTik] > :if ([:len [/ip firewall filter find]] > 0) do={:put true} else={:put false}
true
[admin@MikroTik] >
```

檢查閘道是否能到達。在這個事例中閘道位址為 10.0.0.254

```
[admin@MikroTik] > :if ([/ping 10.0.0.254 count=1] = 0) do {:put "gateway unreachable"}
10.0.0.254 ping timeout
1 packets transmitted, 0 packets received, 100% packet loss
gateway unreachable
[admin@MikroTik] >
```

led – 允許控制系統內嵌的 LED（發光二極體）。這個命令僅能在 RouterBOARD 平臺與安裝 routerboard 或 rb500 功能包。LED 數量根據 RouterBOARD 型號不同而定。

輸入參數

- led1(yes | no) – 控制第一個 LED
- led2(yes | no) -控制第二個 LED
- led3(yes | no) -控制第三個 LED
- led4(yes | no) -控制第四個 LED
- length(time) – 具體指定操作的長度
- omitted – LED 長亮

打開 LED2 和 LED3 時間為 5 秒

```
[admin@MikroTik] > :led led2=yes led3=yes length=5s
```

len – 返回在字串的字元數或清單中的元素數目

輸入參數

- unnamed(name) – 返回字串或清單的長度

```
[admin@MikroTik] > :put [:len gvejimezyfopmekun]
17
[admin@MikroTik] > :put [:len gve,jim,ezy,fop,mek,un]
6
[admin@MikroTik] >
```

local - 聲明本地變數

輸入參數

- unnamed(名稱) - 變數名稱
- unnamed(文本) - 值，分配給變數的內容

```
[admin@MikroTik] > :local MyString "This is a string"
[admin@MikroTik] > :local IPAddr 10.0.0.1
[admin@MikroTik] > :local time   0:10
[admin@MikroTik] > :environment print
Global Variables
Local Variables
IPAddr=10.0.0.1
time=00:10:00
MyString=This is a string
[admin@MikroTik] >
```

log - 通過參數添加一個指定的資訊到系統 logs 中。

輸入參數

- unnamed(名稱) - 記錄日誌的功能名稱
- unnamed(文本) - 被記錄的文本資訊

發送資訊到 info 日誌中

```
[admin@MikroTik] > :log info "Very Good thing happened. We have received our first packet!"
[admin@MikroTik] > /log print follow
...
19:57:46 script,info Very Good thing happened. We have received our first packet!
...
```

nothing - 沒有任何操作，並返回數值型別為“nothing”。 在條件陳述式中 nothing 等同“false”

從一個字串中挑選一個不存在的符號

```
[admin@MikroTik] > :local string qwerty
[admin@MikroTik] > :if ([:pick $string 10]=[:nothing]) do={
{... :put "pick and nothing commands return the same value"}
pick and nothing commands return the same value
[admin@MikroTik] >
```

pick - 根據輸入的值返回一個元素長度或一個子串值

輸入參數

- unnamed(文本 | 清單) - 字串或值清單的來源

- unnamed(整型) - 字串中子串的起始位置
- unnamed(整型) -字串中子串的結束位置

```
[admin@MikroTik] > :set a 1,2,3,4,5,6,7,8
[admin@MikroTik] > :put [:len $a]
8
[admin@MikroTik] > :put [:pick $a]
1
[admin@MikroTik] > :put [:pick $a 0 4]
1,2,3,4
[admin@MikroTik] > :put [:pick $a 2 4]
3,4
[admin@MikroTik] > :put [:pick $a 2]
3
[admin@MikroTik] > :put [:pick $a 5 1000000]
6,7,8
[admin@MikroTik] > :set a abcdefghij
[admin@MikroTik] > :put [:len $a]
10
[admin@MikroTik] > :put [:pick $a]
a
[admin@MikroTik] > :put [:pick $a 0 4]
abcd
[admin@MikroTik] > :put [:pick $a 2 4]
cd
[admin@MikroTik] > :put [:pick $a 2]
c
[admin@MikroTik] > :put [:pick $a 5 1000000]
fghij
```

put - 回復所提供的變數值到控制台

輸入參數

- unnamed(文本) - 需要回復的文本資訊

顯示 ether1 介面的 MTU 值

```
[admin@MikroTik] > :put [/interface get ether1 mtu]
1500
[admin@MikroTik] >
```

resolve - 解析 DNS 功能變數名稱並返回主機的 IP 位址，首先需要配置好路由器的 DNS 參數(/ip dns 目錄下)

輸入參數

- unnamed(文本) - 需要解析 IP 的主機功能變數名稱

DNS 配置和 resolve 命令事例

```
[admin@MikroTik] ip route> /ip dns set primary-dns=159.148.60.2
[admin@MikroTik] ip route> :put [:resolve “www.example.com”]
192.0.34.166
```

set - 分配一個新值給變數

輸入參數

- unnamed(name) - 變數名稱
- unnamed(text) - 新的變數值

通過 **/ip route find dst 0.0.0.0** 的命令，查找路由表中 **dst-address** 返回值為 **0.0.0.0** 的值，這個值通常是路由器的預設閘道器，當查到後通過**/ip route set** 命令修改閘道地址為 10.0.0.217

```
[admin@MikroTik] > /ip route set [/ip route find dst “0.0.0.0”] gateway 10.0.0.1
[admin@MikroTik] >
```

time - 計算出所給命令的執行時間總長度

輸入參數

- unnamed(text) - 控制台命令測量執行時間

計算出解析 www.example.com 需要的時間

```
[admin@MikroTik] > :put [:time [:resolve “www.example.com” ]]
00:00:00.006
[admin@MikroTik] >
```

while - 反復執行給定的控制命令，直到邏輯條件為 true

輸入參數

- unnamed(yes | no) - 條件，在每一次執行前判斷聲明範圍
- **do**(文本) - 配合 while 執行的控制命令

```
[admin@MikroTik] > :set i 0; :while ($i < 10) do={:put $i; :set i ($i + 1)};
0
1
2
3
4
5
6
7
8
```

```
9
[admin@MikroTik] >
```

Typeof - 判斷變數類型，返回值 num、str、nothing、time、ip、bool

輸入參數:

unnamed(name) - 變數名稱

判斷一個變數的類型

```
[admin@MikroTik] >:global a 192.168.1.1
[admin@MikroTik] > :put [:typeof $a]
ip
[admin@MikroTik] >:global b
[admin@MikroTik] >:put [:typeof $b]
nothing
```

2.3 Scripte 事例

啟動延遲

如果你的腳本依賴介面相關的配置，可能有 RouterOS 剛啟動時無法執行的情況，建議延遲執行或者檢查所有需要的介面可以獲取的時候執行，下面的腳本定義有 10 個介面，設置等待時間 30 秒，在 30 秒內沒有 10 個介面，就在日誌中顯示介面沒有載入完

```
#初始化變數 i
:local i 0
#介面總數量
:local x 10
#最大等待時間 30
:local t 30
#while 迴圈如果 i 小於 30，且介面數量小於 10 執行
while ($i < $t && [:len [/interface find]] < $x) do={
:put $i
:set $i ($i + 1)
:delay 1
}
#如果 i 等於 t，就說明介面沒有載入完，反之載入完成可以執行後續腳本
if ($i = $t) do={
:log warning message="Could not load all physical interfaces"
} else={
#執行你的後續腳本
}
```

自動創建多條策略

在 firewall **input** 規則中通過腳本添加接受從 1.1.1.1 開始到 1.1.1.100 位址的資料包，即 100 條規則 ：

```
:for e from 1 to 100 do={
  /ip firewall filte add \
    chain=input src-address=(“1.1.1.” . $e)
   }
```

我們使用類似的腳本編寫，添加 200 個 IP 位址的流量規則：

```
:for e from 1 to 200 do={
  /queue simple add target-addresses=("192.168.1." . $e) max-limit=256000/512000
}
```

獲取 bandwidth 測試參數

這個事例描述的是如果獲取 **bandwidth-test** 命令的結果。 在事例中使用 global（全域變數）另外一個腳本在同一時間運行，並獲取當前的 TX 參數。

```
  :global i
  /tool bandwidth-test 1.1.1.1 direction=transmit duration=14s
    do={
      :if ($status="running") do={
        :set i $” tx-current”
        }
      }
  }
```

創建一個檔

在 v3.x 不能直接創建檔，然而有一種變通方法，我們創建一個 myFile 檔，默認是 txt 格式，通過 set 命名可寫入相應的內容

```
/file print file=myFile
/file set myFile.txt contents="123"
```

檢查 IP 位址在一個介面上是否改變

有時運營商提供動態的 IP 位址，這個腳本會比較動態 IP 位址是否改變。

```
:global currentIP;

:local newIP [/ip address get [find interface="ether1"] address];

:if ($newIP  != $currentIP) do={
```

```
    :put "ip address $currentIP changed to $newIP";
    :set currentIP $newIP;
}
```

分離子網路遮罩

下面的腳本可用於分離子網路遮罩的操作：

```
方法 1
:global ipaddress 10.1.101.1/24

:for i from=( [:len $ipaddress] - 1) to=0 do={
       :if ( [:pick $ipaddress $i] = "/") do={
                :put [:pick $ipaddress 0 $i]
       }
}

方法 2
:set ipaddress [:pick $ddns-ip 0 ([:len $ddns-ip] - 3)]

方法 3
:set ipaddress [:pick $ipaddress 0 [:find $ddns-ip "/"]]
```

功能變數名稱解析

許多用戶通過解析 DNS 功能變數名稱來替換 Radius 伺服器、VPN 伺服器和防火牆等 IP 位址，例如下面是如何解析一個 Radius 伺服器功能變數名稱的 IP 位址。下面是配置一個 Radius 伺服器連接，配置內容是如下：

```
/radius
add address=3.4.5.6 comment=myRad
```

這裡的腳本將 server.example.com 的功能變數名稱解析為 IP 位址，比較之前解析的 IP 位址，如果不相同則進行替換：

```
/system script add name="resolver" source= {

:local resolvedIP [:resolve "server.example.com"];
:local radiusID [/radius find comment="myRad"];
:local currentIP [/radius get $radiusID address];

:if ($resolvedIP  != $currentIP) do={
   /radius set $radiusID address=$resolvedIP;
   /log info "radius ip updated";
}

}
```

這個腳本將在 scheduler 每間隔 5 分鐘運行一次：

```
/system scheduler add name=resolveRadiusIP on-event="resolver" interval=5m
```

產生備份檔案並通過 e-mail 發送

這個腳本產生備份檔案，並發送到規定的 e-mail 位址，Mail 題目包含路由器的名字，當前日期和時間
注意：在腳本執行前，SMTP 伺服器必須配置。查看在/tool e-mail 的配置選項

```
/system backup save name=email_backup
/tool e-mail send file=email_backup.backup to="me@test.com" body="See attached file" \
   subject="$[/system identity get name] $[/system clock get time] $[/system clock get date]
Backup")
```

解析功能變數名稱 IP 地址，並添加入 address-list

這個實例對一些功能變數名稱有多 IP 位址的網站進行解析，我們可以通過設置計畫任務每間隔一個週期執行腳本，比如 netbar.qq.com 進行解析，每次解析對比 ip firewall address-list 是否存在相同 IP，如果沒有相同 IP 位址，則添加入 address-list。

```
:global a [:resolve netbar.qq.com]
:global b
:foreach i in=[/ip firewall address-list find list=qqgame] do={
   :if ($a = [/ip firewall address-list get $i address ]) do={
    :set b 1
  }
   else={
    :set b 0
  }

}
:if ($b = 0) do={ /ip firewall address-list add list=qqgame address=$a }
```

使用注釋

如果有時候在一個規則中與其他規則許多屬性相同，指定的參數就無法從規則中獲取（例如：firewall 或者 routing 規則等）。這裡我們可以通過編輯注釋來解決。I

假設，我們需要從 ip firewall nat 中修改端口映射的目標位址，目標 IP 位址是 218.16.18.12，而我們做了多條端口映射規則，為方便查找和修改，我們可以通過注釋標記。

```
[admin@MikroTik] ip firewall nat> set 0 comment=dst1
```

現在我們看看一個具體的應用實例，ADSL 的 IP 位址變動後，我們修改端口映射的 IP 位址：

```
:global adsl "pppoe-out1"   # 定義 ADSL 撥號介面名稱變數: pppoe-out1
:global adsllast        # 之前的 ADSL IP 位址變數
```

```
:global adslip [ /ip address get [/ip address find interface=$adsl] address ] //獲取當前 ADSL ip 地址

:if ([ :typeof $adsllast ] = nil ) do={ :set adsllast 0.0.0.0/0 } # 判斷之前的 ADSL IP 位址是否為空,如果為空設置一個 0.0.0.0/0 的 IP 位址

:if ([ :typeof $adsl ] = nil ) do={

  :log info ("No ip address present on " . $adsl . ", please check.") # 判斷當前 ADSL IP 是否為空,否則執行 else 的操作

} else={

  :if ($adslip != $adsllast) do={
    :log info [/ip firewall nat set [/ip firewall nat find comment = “dst1”] dst-address= $adslip ]    #  判斷當前 IP 和之前 IP 是否相同，如果不同便修改 dst-address 的 IP 位址
    :log info "ADSL IP: UPDATE!"

    :set adsllast $adslip   # 交換 IP 地址

  } else={

    :log info "ADSL IP: No change"
 }

}
```

在這個事例中，我們可以看到通過添加注釋，為編寫腳本判斷正確的規則。

DDNS 動態功能變數名稱配置

這裡我們來看看，當我們使用 ADSL 時由於重新撥號或者租約到期，IP 位址是動態變化的，而又需要讓外網的客戶通過動態功能變數名稱來訪問我們的內部伺服器，這裡我們需要使用到 DDNS 的動態功能變數名稱腳本，注意 RouterOS 支援的動態功能變數名稱伺服器只有 www.changeip.com

```
:log info “DDNS: Begin“    #在 log 日誌中顯示 DDNS 開始運行

:global ddnsuser “ddns.test.com“   # 定義 ddnsuser 用戶名變數
:global ddnspass “cdnat“                # 定義 ddnspass 密碼變數
:global ddnshost “ddns.test.com“   # 定義 ddnshost 主機
:global ddnsinterface “pppoe-out1“  # 定義 ADSL 撥號介面，使用者獲取 IP
:global ddnsipt  #定義一個零時存儲的 IP 位址變數
:global ddnsip [ /ip address get [/ip address find interface=$ddnsinterface] address ]  # 根據 ADSL 撥號的介面獲取 IP 位址，並分配給 ddnsip
```

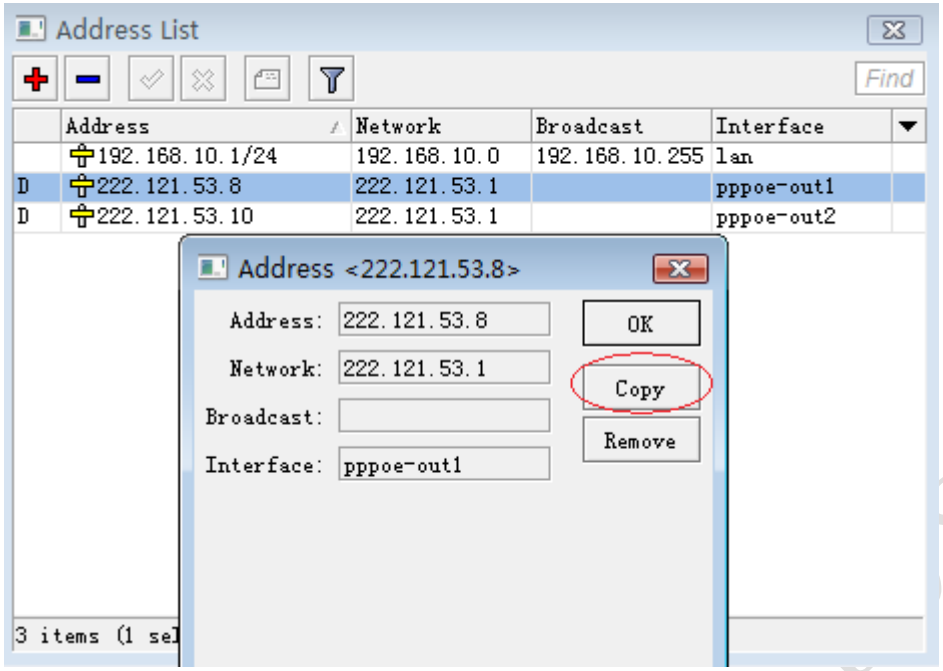
```
:if ([ :typeof $ddnslastip ] = nil ) do={ :global ddnslastip 0.0.0.0/0 }  # 查看 ddnslastip 變數是否為空，如果為空則分配 0.0.0.0/0 地址
:if ([ :typeof $ddnsip ] = nil ) do={

  :log info (“DDNS: No ip address present on ” . $ddnsinterface . “, please check.”)  # 查看 ddns-ip 變數是否為空，如果為空則在 log 中提示未獲取 IP

} else={

  :if ($ddnsip != $ddnslastip) do={                                    # 否則執行新 IP 與以前的 IP 位址對比
    :set ddnsipt [:pick $a 0 ([:len $a] - 3)]                          # 去掉 IP 地址後面的子網路遮罩，並將修改後的 IP 分配給 ddns-ipt 變數
    :log info [/ip fi nat set [/ip fi nat find comment = s1] to-address $ddnsipt ]    # 根據注釋名稱修改相應的 nat 規則
    :log info [/ip fi nat set [/ip fi nat find comment = a1] dst-address $ddnsip ]
    :log info [/ip fi nat set [/ip fi nat find comment = a2] dst-address $ddnsip ]
    :log info [/ip fi nat set [/ip fi nat find comment = a3] dst-address $ddnsip ]
    :log info [/ip fi nat set [/ip fi nat find comment = a4] dst-address $ddnsip ]
    :log info [/ip fi nat set [/ip fi nat find comment = a5] dst-address $ddnsip ]
    :log info [/ip fi nat set [/ip fi nat find comment = a6] dst-address $ddnsip ]

    :log info “DDNS: Sending UPDATE!“     # log 中提示 DDNS 更新
    :log info [ /tool dns-update name=$ddnshost address=[:pick $ddnsip 0 [:find $ddnsip "/"] ] key-name=$ddnsuser key=$ddnspass ]
    # 發送最新的 IP 位址，到 DDNS 伺服器
    :global ddnslastip $ddnsip   # 將新老 IP 地址交換

  } else={
    :log info “DDNS: No change”
 }
}
:log info "DDNS: End"
```

相同 ADSL 閘道指令碼修改

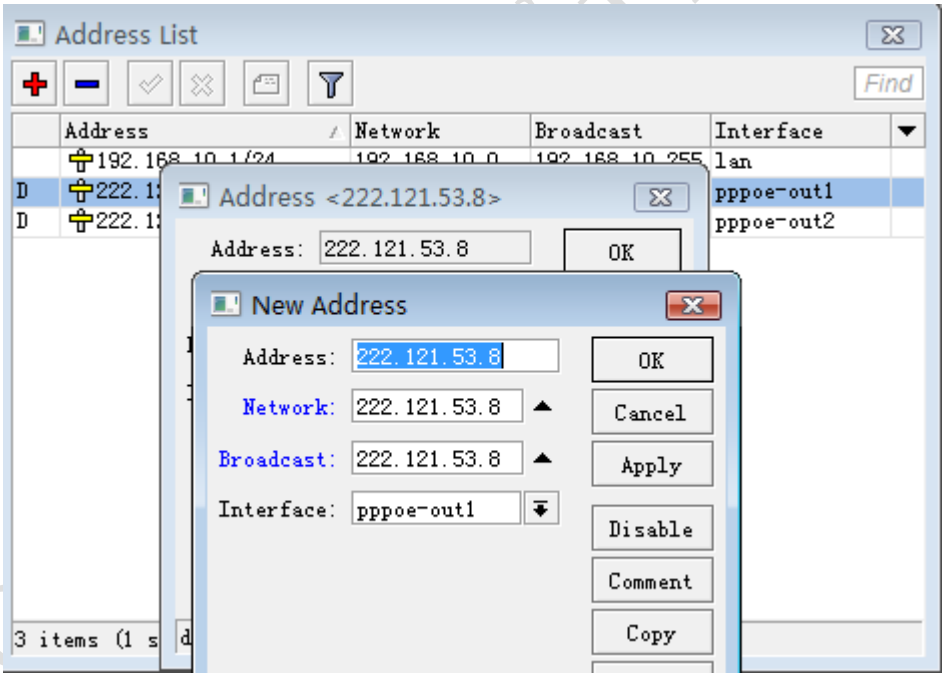
注： 該腳本適用於 v3.0 版本，v4.0 在相同 ADSL 閘道下可以通介面設置，已經不需要腳本。

當我們使用到多線路的 ADSL 時，可能會遇到這些 ADSL 的閘道是相同的，因為 RouterOS 只能識別不同閘道的策略路由，因此在多 ADSL 相同閘道的情況下，我們可以通過隧道協議的特點，用本地 IP 位址做為路由器的閘道。

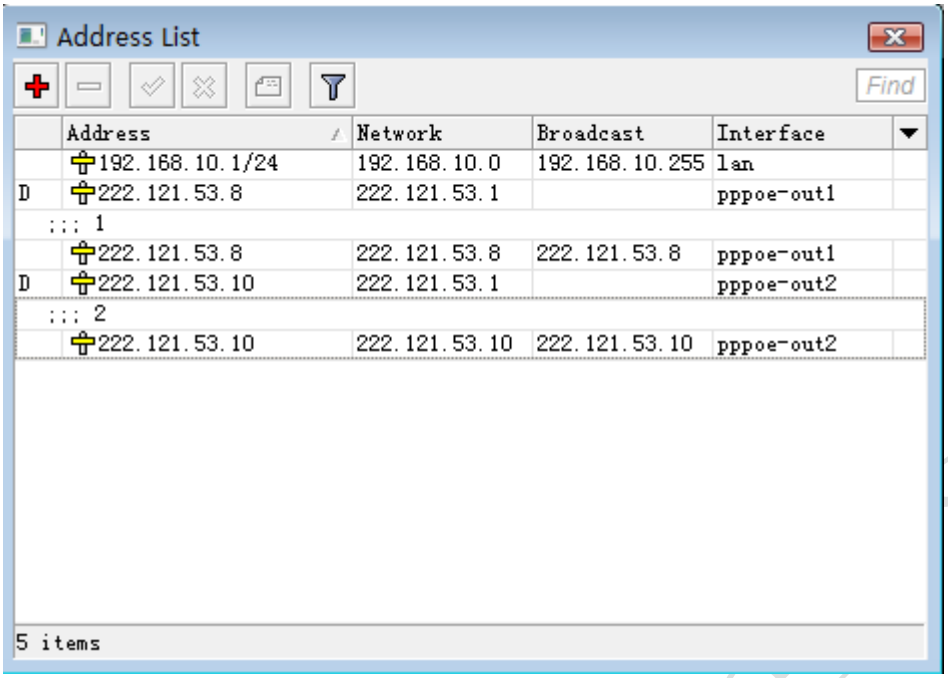
假設我們有 2 條 ADSL，分別為 pppoe-out1 和 pppoe-out2，做 ADSL 操作的時候需要將動態獲取的 ADSL 位址修改為靜態，通過在/ip address 中使用 copy 命令操作，如下圖



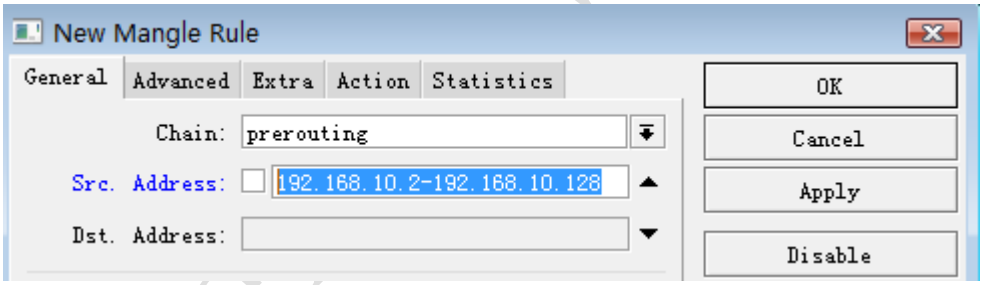
上圖獲取到的 IP 地址是 222.121.53.8，通過 copy 命令修改 Network 和 Boradcast 地址也為 222.121.53.8，如下：

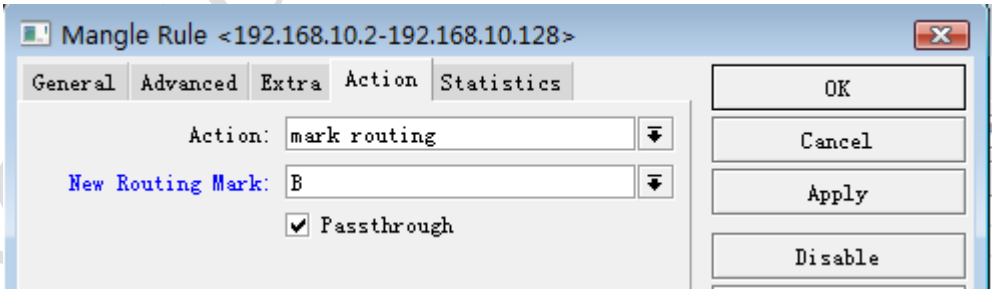


將下面 2 條 ADSL 的 IP 位址都修改為靜態的，並給靜態的 IP 位址標記上注釋 1 和 2：



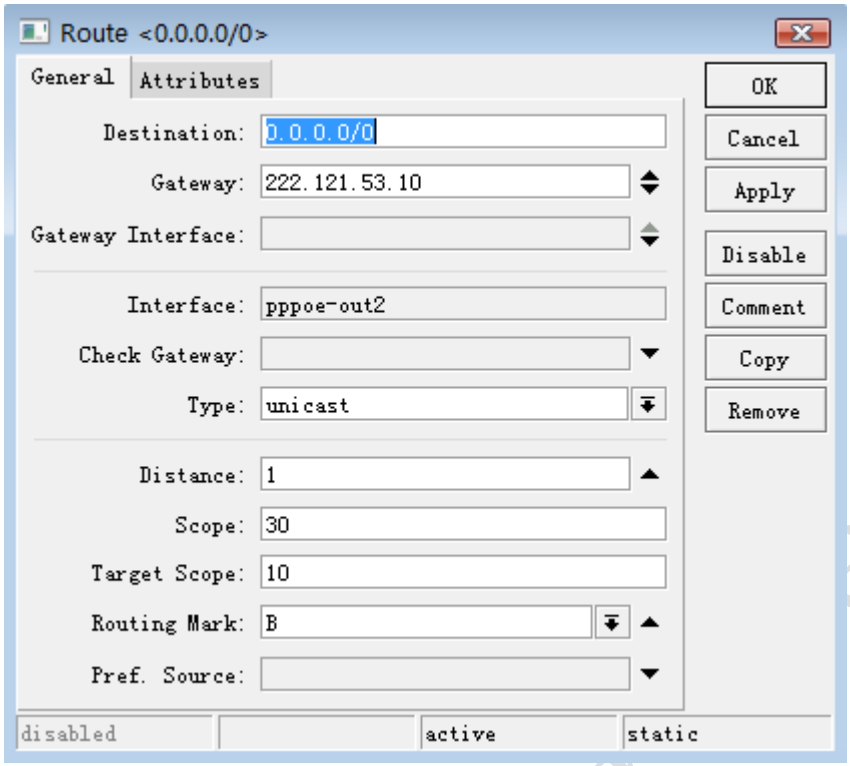
我們將網路的內的使用者平均分組為 A 和 B，A 組走 1 號 ADSL 線路，而 B 組走 2 號 ADSL 線路。1 號 ADSL 設置為默認路由，即 RouterOS 的預設閘道器出口，這裡我們將 2 號線作為用戶 B 組的策略路由，我們在 ip firewall mangle 中配置，定義 chain=prerouting src-address=192.168.10.2-192.168.10.128 action=mark-routing new-routing-mark=B



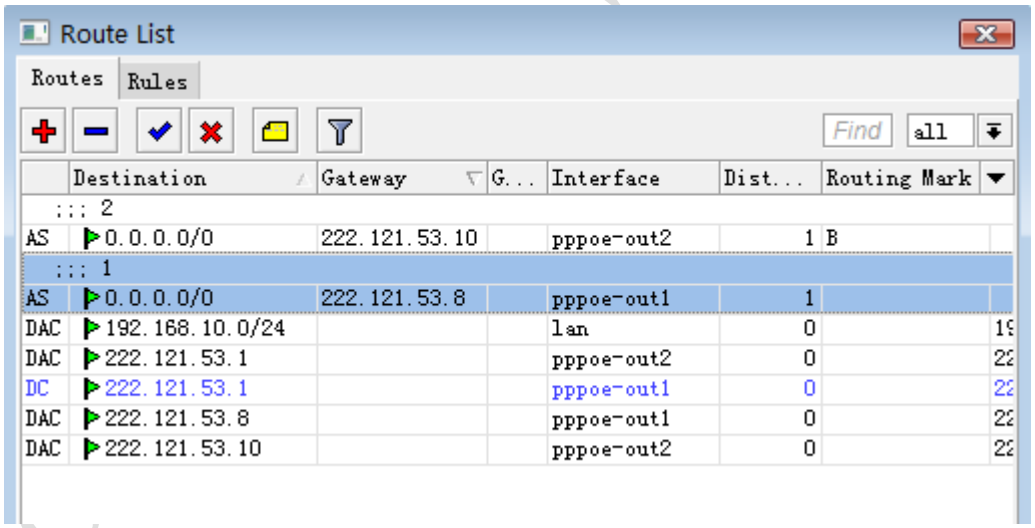


這裡我們只需要定義 B 組用戶，A 組用戶只需要走 1 號 ADSL 的預設閘道器，定義 B 組用戶走 2 號線路的 ADSL

配置路由，這裡我們設置 1 號 ADSL 的 IP222.121.53.8 為預設閘道器，即 ADSL 的預設閘道器。2 號線路的 ADSL 我們用 222.121.53.10，做為 B 組線路的閘道，並在 ip route 中添加 routing-mark 的標記



為 pppoe-out2 做注釋標記 2，配置完成後的路由，B 組用戶通過 22.121.53.10 的閘道出去，剩下的用戶通過預設閘道器如下圖



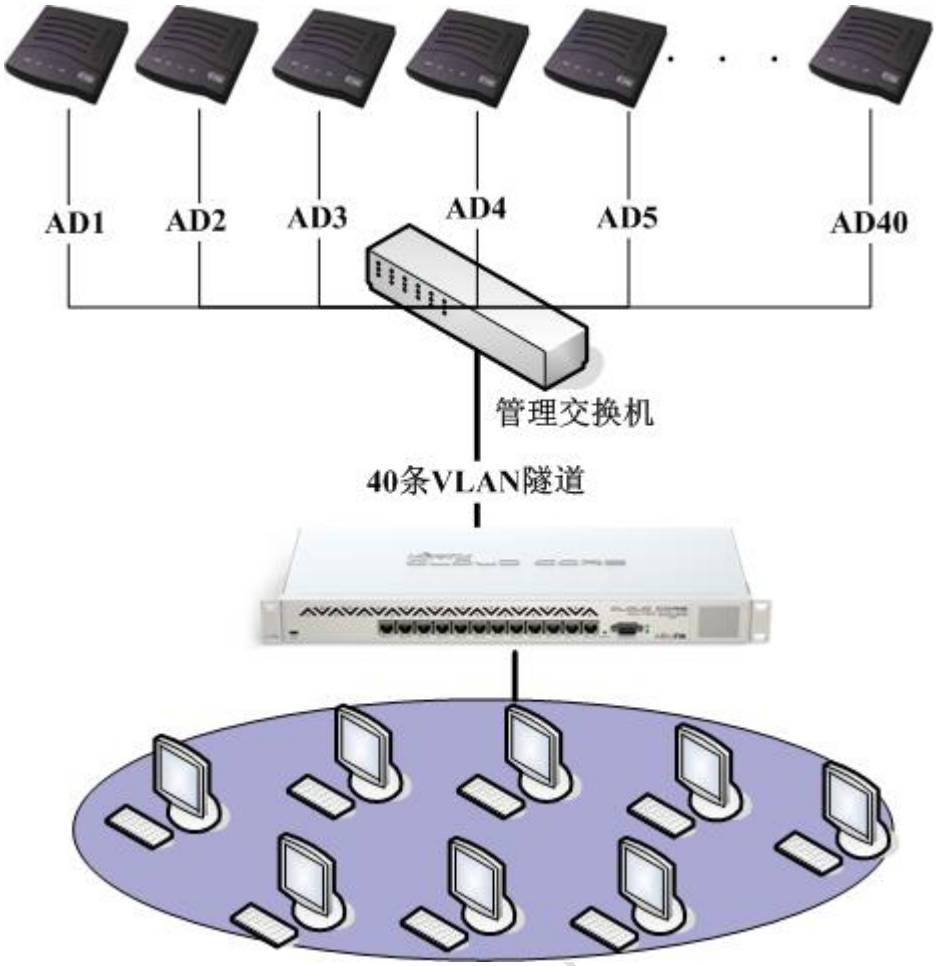
接下來我們需要通過腳本，來判斷 1 和 2 號線 ADSL 的 IP 地址是否變動，如果 IP 變動後用腳本會自動修改變動的 1 和 2 號線 ADSL 參數

```
:local lastaddress
:local newaddress
:local status

:set status [/interface get [/interface find    name="pppoe-out1" ] running]
:if ($status=true) do={
        :set newaddress [/ip address get [/ip address find dynamic=yes interface="pppoe-out1" ]
address]
```

```
        :set newaddress [:pick $newaddress 0 [:find $newaddress "/" ]]
        :set lastaddress [/ip address get [/ip address find dynamic=no interface="pppoe-out1"]
address]
        :set lastaddress [:pick $lastaddress 0 [:find $lastaddress "/" ]]
       :if ($lastaddress != $newaddress) do={
          :log info [/ip address set [/ip address find comment=" 1" ] address=$newaddress
network=$newaddress broadcast=$newaddress]
            :log info [/ip route set [/ip route find comment=" 1" ] gateway=$newaddress]
              }
  }
:set status [/interface get [/interface find    name="pppoe-out2" ] running]
:if ($status=true) do={
        :set newaddress [/ip address get [/ip address find dynamic=yes interface="pppoe-out2" ]
address]
        :set newaddress [:pick $newaddress 0 [:find $newaddress "/" ]]
        :set lastaddress [/ip address get [/ip address find dynamic=no interface="pppoe-out2"]
address]
        :set lastaddress [:pick $lastaddress 0 [:find $lastaddress "/" ]]
       :if ($lastaddress != $newaddress) do={
          :log info [/ip address set [/ip address find comment=" 2" ] address=$newaddress
network=$newaddress broadcast=$newaddress]
            :log info [/ip route set [/ip route find comment=" 2" ] gateway=$newaddress]
              }
  }
```

40 條線路負載均衡配置與腳本

RouterOS 支援多線路的負載均衡，某社區為了節約費用，採用 40 條 2M 頻寬的 AD 通過做彙聚實現高頻寬的社區頻寬，為解決介面問題採用一台 Cisco 的 48 口的交換機做 VLAN 接入 40 條 AD，讓後通過 VLAN 連接到 RouterOS 進行撥號，再做 PCC 負載均衡，網路拓撲圖如下：



外網接入的方法是在交換機和 RouterOS 路由器上劃分 VLAN，然後在 ROS 對應的 VLAN 上做 PPPoE-CLIENT。

1、首先劃分 VLAN（我們這裡是從 2 開始排序的），腳本如下：

```
[admin@MikroTik] > :for i from=2 to=41 do= {interface vlan add name=("vlan".$i) vlan-id=$i
interface=ether2-wan }
```

2、然後添加 PPPOE 撥號（先添加撥號再手動輸入 每個 AD 的帳號和密碼，40 條 AD 設置還是要花點時間了），腳本如下：

```
[admin@MikroTik] > :for i from=2 to=41 do= {interface pppoe-client add name=("pp
poe-out".$i) user=$i password=$i interface=("vlan".$i)}
```

3、我們這裡採用 PCC 的負載均衡，在 ip firewall mangle 裡添加相應的 PCC 規則，通過一些腳本添加 PCC 的規則，注意：如果 PPPoE 用戶端撥號沒有成功，那麼添加的規則則為紅色的，撥號成功後自動正常

```
[admin@MikroTik] > :for i from=2 to=41 do={/ip firewall mangle add chain=input a
ction=mark-connection new-connection-mark=conn1 in-interface=("pppoe-out".$i)}
```

4.然後標記路由讓從哪個介面進來的資料就從哪個介面出去：

```
[admin@MikroTik] >  :for i from=2 to=41 do= {ip firewall mangle add chain=output
 connection-mark=("conn".$i) action=mark-routing new-routing-mark=("rout".$i)}
[admin@MikroTik] >
```

5. 然後將所有內網出來的資料通過 pcc 的 both-addresses 分成 40 分並標記連接和路由：

```
[admin@MikroTik] > :for i from=2 to=41 do= {/ip firewall mangle add chain=prerou
ting src-address-list=lan-add action=mark-connection new-connection-mark=("conn"
.$i) per-connection-classifier=("both-addresses:40/".$i) comment=$i
{... /ip firewall mangle add chain=prerouting src-address-list=lan-add action=ma
rk-routing new-routing-mark=("rout".($i-2)) connection-mark=("conn".$i)}
```

6.為每個路由標記添加路由並添加 pppoe-out41 為默認路由：

```
[admin@MikroTik] > :for i from=2 to=41 do= {ip route add dst-address=0.0.0.0/0 g
ateway=("pppoe-out".$i) routing-mark=("rout".$i)}
[admin@MikroTik] >ip routed add dst-address=0.0.0.0/0 gateway=pppoe-out41
```

7. 最後做 NAT 偽裝，一般最好是對每個出口進行偽裝：

```
[admin@MikroTik] > ip firewall nat add chain=srcnat action=masquerade
```

PCC 負載均衡腳本

為了方便快速設置，我們可以通過腳本進行規則的迴圈添加：

```
:global interface "ether1-lan"
:global pppoe "pppoe-out"
:global address "192.168.0.0/24"
:global n 6
:for i from=1 to=$n do={
        :log info "a1"
        /ip firewall mangle add chain=input in-interface=($pppoe . $i) action=mark-connection
new-connection-mark=("pcc" . $i)
        :log info "a2"
        /ip firewall mangle add chain=prerouting src-address=$address
per-connection-classifier=("both-addresses:" . $n . "/" . $i-1) dst-address-type=!local
action=mark-connection new-connection-mark=("pcc" . $i)
        /ip firewall mangle add chain=prerouting src-address=$address connection-mark=("pcc" .
$i) action=mark-routing new-routing-mark=("route" . $i)
        /ip firewall mangle add chain=output connection-mark=("pcc" . $i) action=mark-routing
new-routing-mark=("route" . $i)

}
```

array 陣列

RouterOS 的陣列使用和其他語言大同小異，簡單介紹下陣列的使用

一個陣列的定義如下，使用全域變數定義陣列 array，包括元素 1,2,3,4

```
:global array {1;2;3;4}
```

輸出陣列的值

```
:put $array
```

輸出第 2 個元素值，記住陣列序列是從 0 開始計數

```
:put [:pick $array 1]
```

我們也可以將一個元素賦值給一個變數

```
:global tmp [:pick $array 2]
```

我們可以通過 len 得到陣列的長度，並輸出

```
:put [:len $array]
```

通過:len 得到該陣列有多少個元素，這樣有利於我們後面一些代碼的處理，比如 for

```
:for i from=0 to=[:len $array]
```

我們也可以使用 foreach 迴圈取值

```
:foreach i in=[/ip arp find] do={
:set arraymac ($arraymac ,[/ip arp get $i mac-address ])
:set arraynum ($arraynum , $i)
}
```

下面是一個通過比較在 arp 列表裡是否存在相同 mac 位址並刪除的實例（腳本寫的很醜，請諒解），注意定義陣列變數時最好不要用全域變數，因為腳本執行完後，全域變數是不會被清空的，下次執行時會重複追加元素，一般使用區域變數 local 定義：

```
# 定義 mac 陣列

:local arraymac

# 定義標號陣列

:local arraynum

# 相同 mac 的標號陣列

:local arrays

# 第一個相同 mac 標號變數
```

```
:local macfirst
:local n
:local n1

# 取 arp 下的 mac 地址和他們在 arp 列表中的標號

:foreach i in=[/ip arp find] do={
:set arraymac ($arraymac ,[/ip arp get $i mac-address ])
:set arraynum ($arraynum , $i)
}

#取得陣列長度

:set n [:len $arraymac]

#雙迴圈比較

:for m from=0 to=($n-1) do={
:for k from=1 to=$n do={
:if ([:pick $arraymac $m] = [:pick $arraymac $k]) do={
:set macfirst [:pick $arraynum $m]
:set arraySN ($arraynum , [:pick $arraynum $k])
}
}
}

#取得陣列長度

:set n1 [:len $arraySN]

#刪除 arp 列表中相同的 mac 位址

:for j from=0 to=$n1 do={
/ip arp remove [:pick $arraySN $j]
}
/ip arp remove $macfirst
```

陣列的使用還有很多方面，而且非常有用，這個就需要各位在實踐中摸索！

通過聲控判斷 WLAN 信號強度

我們可以使用 RouterOS 的腳本 beep 語句，配合迴圈操作，來判斷 WLAN 的信號強度，該腳本主要應用在點對點的 WLAN 搜索信號使用。

下面是 ap-bridge 使用的聲控信號強度腳本，注意 ap-bridge 使用的時候需要填寫對方無線模組的 MAC 位址，才能獲取信號強度

```
:local beep "10ms";
:local s85 "1350ms";
:local s80 "850ms";
:local s75 "650ms";
:local s70 "450ms";
:local s65 "350ms";
:local s60 "250ms";
:local s55 "200ms";
:local s50 "150ms";
:local s45 "100ms";
:local s40 "60ms";
:local s20 "20ms";
:global fr
:for i from=1 to=50 do={
:set fr [/interface wireless registration-table get [/interface wireless registration-table find
radio-name="000C4223D23E"] signal-strength ]
:set fr [:pick $fr 0 [:find $fr "d" ]]
 :if ($fr <= -85 && $fr > -88) do={
  :for i from=1 to=2 do={ :beep length=$beep; :delay $s85; }
  }
 :if ($fr <= -80 && $fr > -85) do={
  :for i from=1 to=3 do={ :beep length=$beep; :delay $s80; }
  }
 :if ($fr <= -75 && $fr > -80) do={
  :for i from=1 to=3 do={ :beep length=$beep; :delay $s75; }
  }
 :if ($fr <= -70 && $fr > -75) do={
  :for i from=1 to=6 do={ :beep length=$beep; :delay $s70; }
  }
 :if ($fr <= -65 && $fr > -70) do={
  :for i from=1 to=8 do={ :beep length=$beep; :delay $s65; }
  }
 :if ($fr <= -60 && $fr > -65) do={
  :for i from=1 to=11 do={ :beep length=$beep; :delay $s60; }
  }
 :if ($fr <= -55 && $fr > -60) do={
  :for i from=1 to=13 do={ :beep length=$beep; :delay $s55; }
  }
 :if ($fr <= -50 && $fr > -55) do={
  :for i from=1 to=18 do={ :beep length=$beep; :delay $s50; }
  }
 :if ($fr <= -45 && $fr > -50) do={
  :for i from=1 to=25 do={ :beep length=$beep; :delay $s45; }
  }
 :if ($fr <= -40 && $fr > -45) do={
  :for i from=1 to=31 do={ :beep length=$beep; :delay $s40; }
  }
```

```
 :if ($fr <= -20 && $fr > -40) do={
  :for i from=1 to=40 do={ :beep length=$beep; :delay $s20; }
  }

  }
 }
```

該腳本在信號強度越強的情況下，發聲頻率越高

下面是 Station-wds 使用的聲控信號強度腳本：

```
:local beep "10ms";
:local s85 "1350ms";
:local s80 "850ms";
:local s75 "650ms";
:local s70 "450ms";
:local s65 "350ms";
:local s60 "250ms";
:local s55 "200ms";
:local s50 "150ms";
:local s45 "100ms";
:local s40 "60ms";
:local s20 "20ms";
:for i from=1 to=100 do={
/interface wireless monitor wlan1 interval=1 do={
 :if ($"signal-strength" <= -85 && $"signal-strength" > -88) do={
  :for i from=1 to=2 do={ :beep length=$beep; :delay $s85; }
  }
 :if ($"signal-strength" <= -80 && $"signal-strength" > -85) do={
  :for i from=1 to=3 do={ :beep length=$beep; :delay $s80; }
  }
 :if ($"signal-strength" <= -75 && $"signal-strength" > -80) do={
  :for i from=1 to=4 do={ :beep length=$beep; :delay $s75; }
  }
 :if ($"signal-strength" <= -70 && $"signal-strength" > -75) do={
  :for i from=1 to=6 do={ :beep length=$beep; :delay $s70; }
  }
 :if ($"signal-strength" <= -65 && $"signal-strength" > -70) do={
  :for i from=1 to=8 do={ :beep length=$beep; :delay $s65; }
  }
 :if ($"signal-strength" <= -60 && $"signal-strength" > -65) do={
  :for i from=1 to=10 do={ :beep length=$beep; :delay $s60; }
  }
 :if ($"signal-strength" <= -55 && $"signal-strength" > -60) do={
  :for i from=1 to=12 do={ :beep length=$beep; :delay $s55; }
  }
 :if ($"signal-strength" <= -50 && $"signal-strength" > -55) do={
```

```
  :for i from=1 to=16 do={ :beep length=$beep; :delay $s50; }
  }
 :if ($"signal-strength" <= -45 && $"signal-strength" > -50) do={
  :for i from=1 to=24 do={ :beep length=$beep; :delay $s45; }
  }
 :if ($"signal-strength" <= -40 && $"signal-strength" > -45) do={
  :for i from=1 to=34 do={ :beep length=$beep; :delay $s40; }
  }
 :if ($"signal-strength" <= -20 && $"signal-strength" > -40) do={
  :for i from=1 to=48 do={ :beep length=$beep; :delay $s20; }
  }
 }
}
```

聲音控制腳本

警報聲

```
:for i from=1 to=3 step=1 do={
   :beep frequency=550 length=494ms;
   :delay 494ms;
   :beep frequency=400 length=494ms;
   :delay 494ms;
 }
```

電話鈴聲

```
:for i from=1 to=10 step=1 do={
   :beep frequency=1195 length=22ms;
   :delay 22ms;
   :beep frequency=2571 length=22ms;
   :delay 22ms;
 }
```

Coo 發聲

```
:for i from=0 to=150 step=10 do={
   :beep frequency=(1295 - i) length=22ms;
   :delay 22ms;
   :beep frequency=(1095 + i) length=22ms;
   :delay 22ms;
 }
```

操作成功發聲

```
:beep frequency=523 length=200ms;
 :delay 1000ms;
```

```
:beep frequency=523 length=200ms;
:delay 1000ms;

:beep frequency=523 length=200ms;
:delay 1000ms;

:beep frequency=659 length=700ms;
:delay 700ms;

:beep frequency=784 length=500ms;
:delay 500ms;

:beep frequency=523 length=200ms;
:delay 1000ms;

:beep frequency=523 length=200ms;
:delay 1000ms;

:beep frequency=523 length=200ms;
:delay 1000ms;

:beep frequency=659 length=700ms;
:delay 700ms;

:beep frequency=784 length=500ms;
:delay 800ms;

:beep frequency=784 length=400ms;
:delay 400ms;

:beep frequency=884 length=200ms;
:delay 200ms;

:beep frequency=784 length=200ms;
:delay 200ms;

:beep frequency=687 length=200ms;
:delay 200ms;

:beep frequency=659 length=200ms;
:delay 200ms;

:beep frequency=579 length=200ms;
:delay 200ms;

:beep frequency=519 length=400ms;
```

```
:delay 400ms;
```

第三章 WLAN 無線基礎知識

3.1 802.11 傳輸協議

學習 WLAN 無線知識前，我們需要先來瞭解下關於 802.11 無線協議的基本應用和技術，這樣才能真正理解和操作包括 RouterOS 在內的所有 802.11 協定的無線設備，同時也能理解為什麼構建 WiFi 或 WLAN 網路前需要做一些勘查和分析，為什麼無線網路比有線網路方便，但會遇到比有線網路更複雜的問題。

不管是 RouterOS 還是其他無線設備配置基本是相差不大，關鍵是需要對無線協定、硬體設施、網路環境以及周邊地理與環境情況進行分析，確定你選擇的是正確的解決方案，雖然 RouterOS 的 802.11 無線技術不是最先進的，但 RouterOS 在整體的網路解決方案中有較大的優勢，因為他的各項網路功能是其他無線網路設備不能比擬的。

802.11 傳輸協定我們知道分為 abgn，他們的頻率使用又分為 2.4GHz 和 5GHz，802.11bg 使用的是 2.4G 頻段，802.11a 使用的是 5G 頻段，2.4GHz 頻率更低，傳輸距離、穿透和繞射能力更強，而 5GHz 頻率較高，傳輸距離、穿透和繞射能力都不如 2.4GHz，但其資料承載能力較 2.4G 更強，且該頻段較為乾淨，干擾小。 所以，通常 2.4GHz 會被用於 WiFi 的覆蓋上網，5GHz 會用於 WLAN 的資料傳輸。

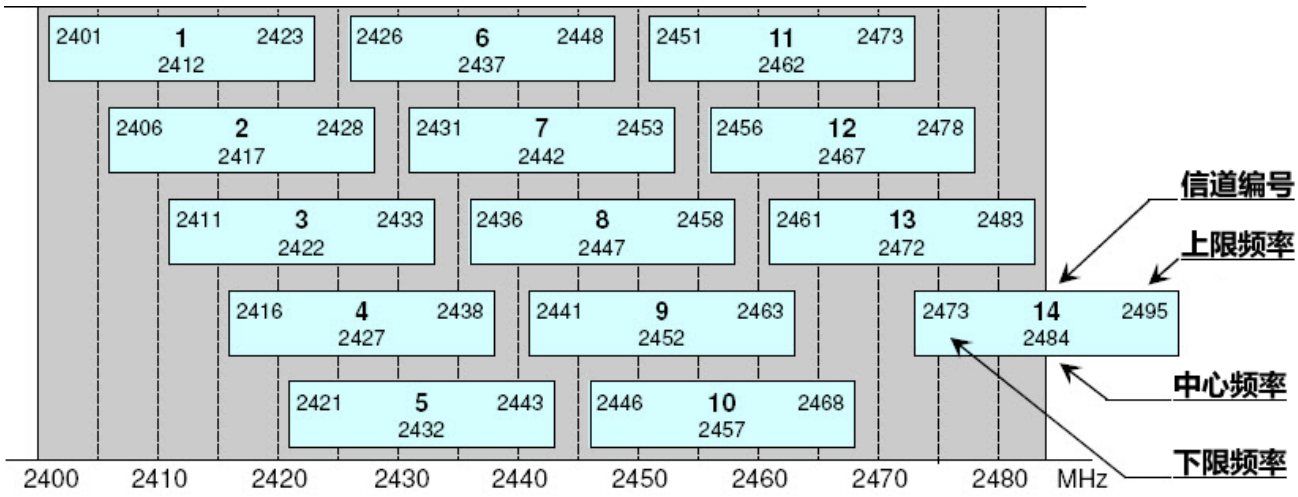
我們在市面上可以看到 11n 的產品說明，例如 802.11bgn 代表的是基於 2.4GHz 的 11n 協議，802.11an 代表的是支持 5GHz 的 11n 協定，如果是 802.11abgn，即同時支持 2.4GHz 和 5GHz 頻率。

下面是各個協議的具體參數：

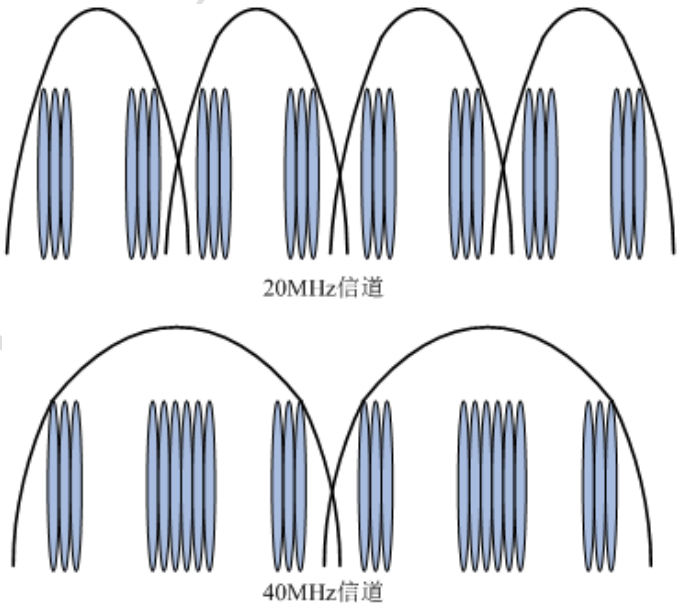
標準	工作頻段	頻道間隔	最大速率
802.11b	2.4GHz： 2312-2599MHz	 5MHz	 11Mbps
802.11g G-Turbo	2.4GHz 2312-2599MHz	 5MHz 44MHz	 54Mbps 108Mbps
802.11a A-Turbo	5GHz： 4920-6100MHz	 5MHz 10MHz 20MHz 40MHz	 13.5Mbps 27Mbps 54Mbps 108Mbps
802.11n	2.4/5Ghz： 2312-2599Mhz 4920-6100Mhz	 5MHz 10MHz 20MHz 40MHz (2×20MHz) 60MHz (3×20MHz)	 37.5Mbps 75Mbps 150Mbps 300Mbps 450Mbps
802.11ac	2.4/5Ghz： 2312-2599Mhz 4920-6100Mhz	20MHz 40MHz 80MHz 160MHz	1.52~2.26Gbps

802.11bg 2.4G 頻率佔用

2.4GHz 頻段中，同一區域覆蓋範圍內最多容納 3 個互不重迭的通道（如下圖），每通道最大可佔用 22 MHz 的頻帶；11b 採用 DSSS 擴頻和 CCK 的調製方式最高提供 11Mbps 的速率，11g 採用 OFDM 的擴頻方式，可提供 54Mbps 的速率，如 2412MHz 頻率,在使用時候會干擾到上至 2423MHz 和下至 2401MHz

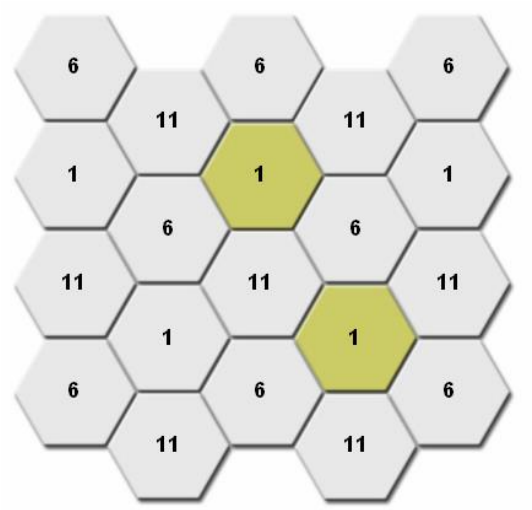


和 11g 類似，在 2.4GHz 頻段可以有三個不重迭的通道，但是對於 11n 來說，使用 20MHz 就意味著只能達到 150Mbps 的速率。11n 與 11g 的 Turbo 模式類似，一旦使用 40MHz 頻寬，在 2.4GHz 頻段只有一個通道可用，基本上無法規模部署，解決方法是使用 5GHz 頻段，頻寬資源豐富（11n 選擇 5G 主要應用於資料傳輸，而非對終端使用者的覆蓋）。



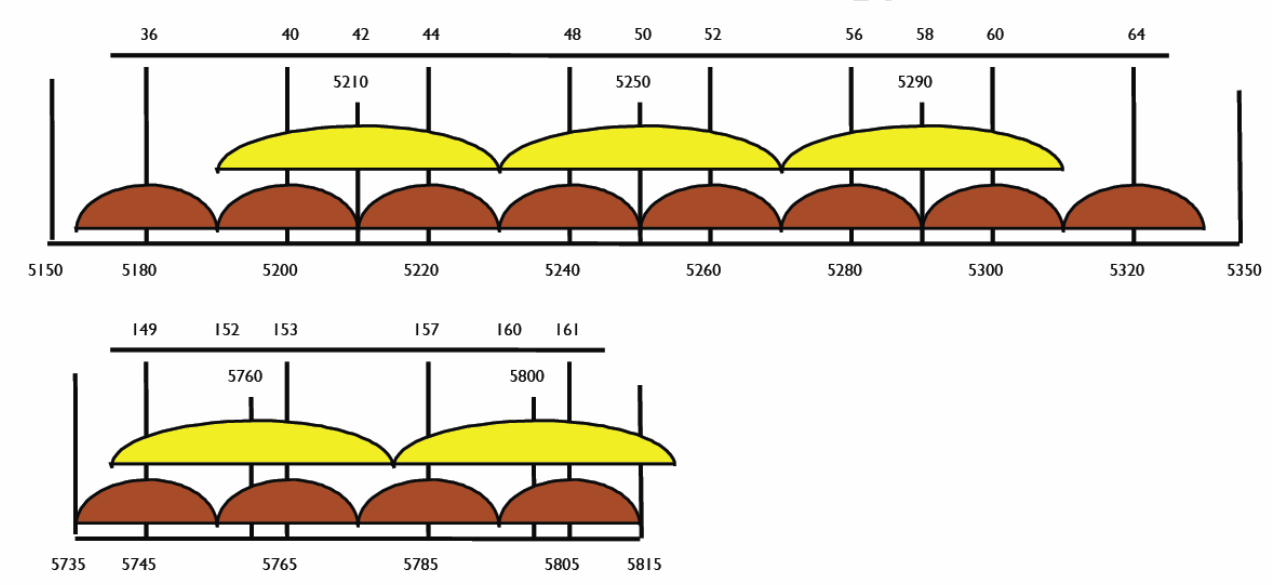
WLAN 2.4G 頻率覆蓋位置規劃

多個鄰居設備間頻率不能相同，頻率應使用不同頻道，如圖多個基站情況相鄰設備間頻道在 1、6、11，可以避免頻道交叉而帶來的干擾問題：



5G 頻道使用情況

802.11a 的 5G 分為 5.2G 和 5.8G 兩個段頻道，5G 有 8+4 個非重迭通道，每個通道使用 16.6MHz 的頻寬，比 11g 的頻寬更小，這就是為什麼我們用 5G 做為骨幹傳輸的原因。



上圖，我們可以看到從 5180-5320 每隔 20MHz 一個頻道，共 8 個頻道，從 5745-5805 的 4 個頻道，而當我們採用 5G-turbo 模式（即多通道傳輸），只能獲得 5 個獨立的頻道，5210、5250、5290、5760 和 5800。

3.2 天線（antenna）

天線是 WLAN 網路的重要組成部分，通過天線將 WLAN 設備發射出的信號放大，並覆蓋或傳向指定的方向。那天線在是如何工作的？下面我們簡單介紹下天線的幾個特性

天線方向性

發射天線的基本功能之一是把從饋線取得的能量向周圍空間輻射出去，基本功能之二是把大部分能量朝所需的方向輻射。垂直放置的半波對稱振子具有平放的 “面包圈” 形的立體方向圖（圖 1）。圖 2 與圖 3 給出了它的兩個主平面方向圖，平面方向圖描述天線在某指定平面上的方向性。從圖 2 側視可以

看出，在振子的軸線方向上輻射為零，最大輻射方向在水平面上；而從圖 3 俯視可以看出，在水平面上各個方向上的輻射一樣大。

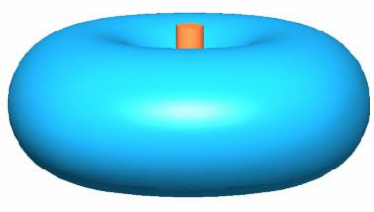


圖 1 面包圈 立體圖

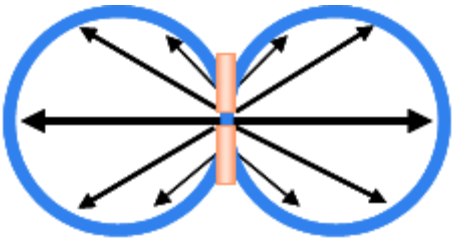


圖 2 側視圖

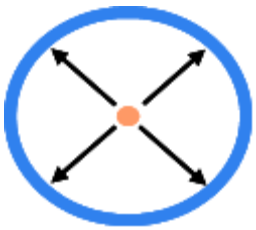
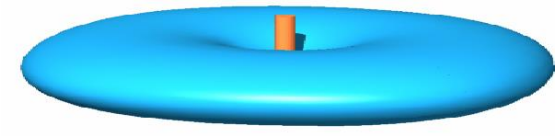
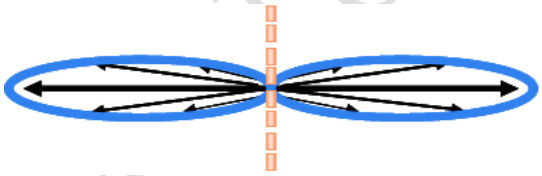


圖 3 俯視

天線方向性增強

若干個對稱振子組陣，能夠控制輻射，產生“扁平的面包圈” ，把信號進一步集中到在水平面方向上。 下圖是 4 個半波對稱振子沿垂線上下排列成一個垂直四元陣時的立體方向圖和垂直面方向圖。

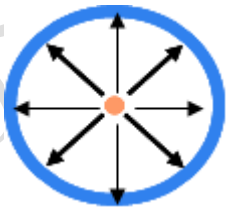




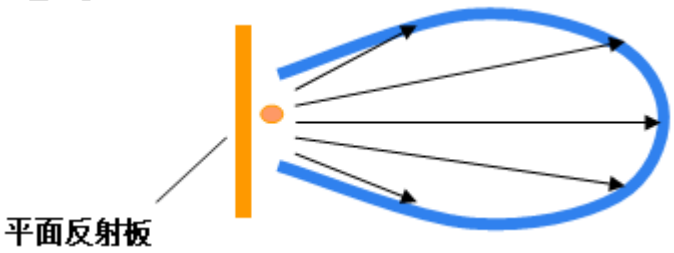
扁平面包圈

利用反射板把輻射能控制到單側方向

平面反射板放在陣列的一邊構成扇形區覆蓋天線。下面的水平面方向圖說明瞭反射面的作用-----反射面把功率反射到單側方向，提高了增益。



全向陣（不帶平面反射板）



扇形區覆蓋 （帶平面反射板）

拋物反射面的使用，更能使天線的輻射，像光學中的探照燈那樣，把能量集中到一個小立體角內，從而獲得很高的增益。不言而喻，拋物面天線的構成包括兩個基本要素：拋物反射面 和 放置在拋物面焦點上的輻射源。

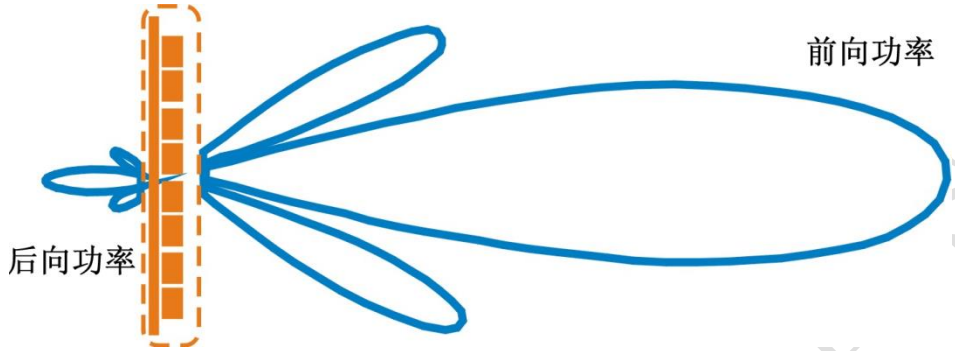
增益

增益是指：在輸入功率相等的條件下，實際天線與理想的輻射單元在空間同一點處所產生的信號的功率密度之比。它定量地描述一個天線把輸入功率集中輻射的程度。增益顯然與天線方向圖有密切的關係，方向圖主瓣越窄，副瓣越小，增益越高。 可以這樣來理解增益的物理含義------為在一定的距離上的某點處產生一定大小的信號，如果用理想的無方向性點源作為發射天線，需要 100W 的輸入功率，而

用增益為 G = 13 dB = 20 的某定向天線作為發射天線時，輸入功率只需 100 / 20 = 5W；換言之，某天線的增益，就其最大輻射方向上的輻射效果來說，與無方向性的理想點源相比，把輸入功率放大的倍數。

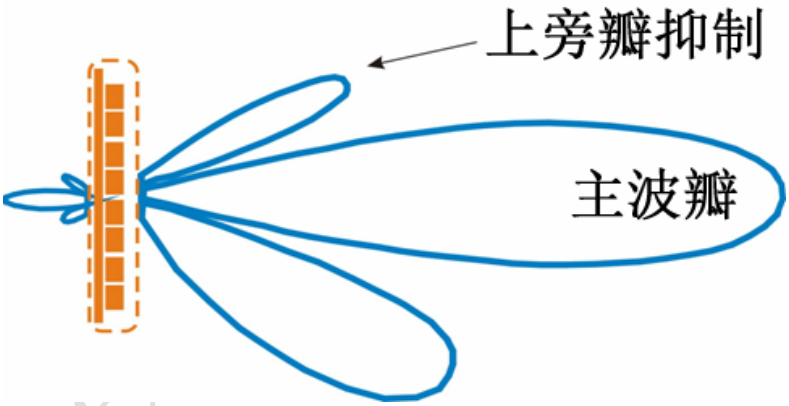
前後比

下圖中，前後瓣最大值之比稱為前後比，記為 F / B 。前後比越大，天線的後向輻射（或接收）越小。其典型值為 （18 ~ 30）dB，特殊情況下則要求達（35 ~ 40）dB .

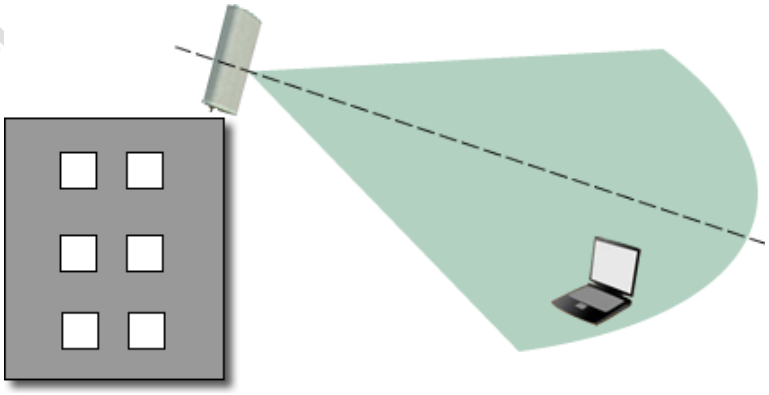


上旁瓣抑制

對於基站天線，人們常常要求它的垂直面（即俯仰面）方向圖中，主瓣上方第一旁瓣盡可能弱一些。這就是所謂的上旁瓣抑制 。基站的服務物件是地面上的行動電話用戶，指向天空的輻射是毫無意義的。

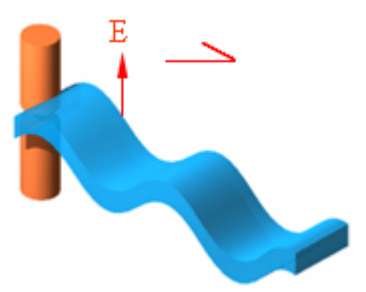


注：天線的下傾——為使主波瓣指向地面，安置時需要將天線適度下傾，這樣的情況用在當 WLAN 設備假設在高處，並對區域覆蓋時。

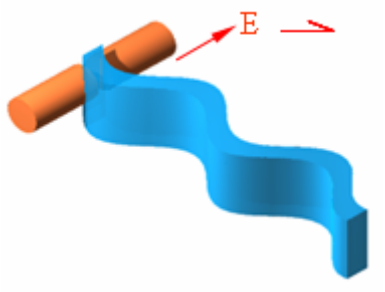


天線的極化

天線向周圍空間輻射電磁波。電磁波由電場和磁場構成。人們規定：電場的方向就是天線極化方向。一般使用的天線為單極化的。下圖示出了兩種基本的單極化的情況：垂直極化---是最常用的一種極化方式；水準極化---在一些場合也會被用到。



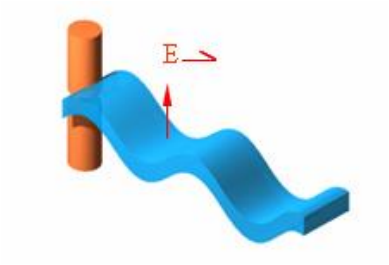
垂直極化



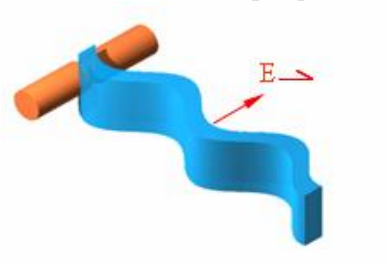
水準極化

雙極化天線

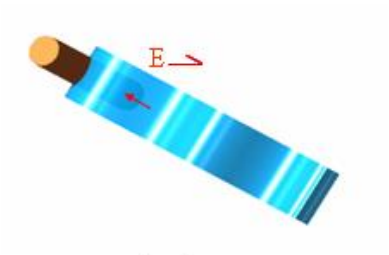
下圖示出了另兩種單極化的情況：±45° 極化，這樣將極化方式增加到四種，也就是說我們在安裝天線時多了兩種方式，但±45° 極化通常使用的較少。見下圖：



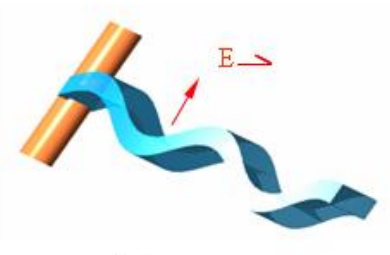
垂直极化



水平极化

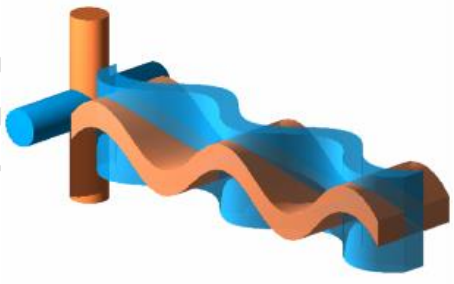


+45° 极化

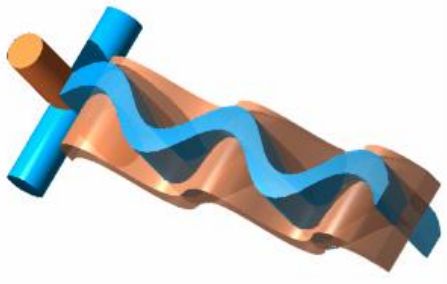


-45° 极化

把垂直極化和水準極化兩種極化的天線組合在一起，或者把 +45° 極化和 -45° 極化兩種極化的天線組合在一起，就構成了一種新的天線---雙極化天線。



±90° 雙極化



±45° 雙極化

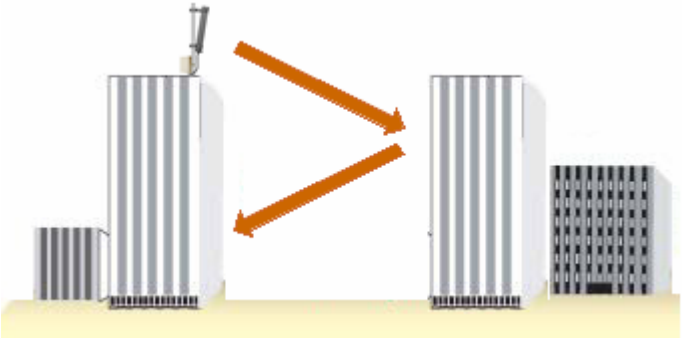
雙極化天線在 RouterOS 上主要應用在 bonding、Nstreme Dual 和 802.11n 的高頻寬傳輸應用上，例如 RB-SXT 集成的就是雙極化 5G 天線。

建議：密集城區和普通城區覆蓋優先選擇 ±45° 雙極化天線；一般郊區，農村和郊區可選擇±90° 雙極化天線。

電波的多徑傳播

在超短波、微波波段，電波在傳播過程中還會遇到障礙物(例如樓房、高大建築物或山丘等）對電波

產生反射。因此，到達接收天線的含有多種反射波（廣意地說，地面反射波也應包括在內），這種現象叫為多徑傳播。



由於多徑傳輸，使得信號場強的空間分佈變得相當複雜，波動很大，有的地方信號場強增強，有的地方信號場強減弱；也由於多徑傳輸的影響，還會使電波的極化方向發生變化。另外，不同的障礙物對電波的反射能力也不同。例如：鋼筋水泥建築物對超短波、微波的反射能力比磚牆強。我們應儘量克服多徑傳輸效應的負面影響，這也正是在通信品質要求較高的通信網中，人們常常採用空間分集技術或極化分集技術的緣由。

電波的繞射傳播

在傳播途徑中遇到大障礙物時，電波會繞過障礙物向前傳播，這種現象叫做電波的繞射。超短波、微波的頻率較高，波長短，繞射能力弱，在高大建築物後面信號強度小，形成所謂的“陰影區”。信號品質受到影響的程度，不僅和建築物的高度有關，和接收天線與建築物之間的距離有關，還和頻率有關。

例如有一個建築物，其高度為 10 米，在建築物後面距離 200 米處，接收的信號品質幾乎不受影響，但在 100 米處，接收信號場強比無建築物時明顯減弱。注意，誠如上面所說過的那樣，減弱程度還與信號頻率有關，對於 216 ～ 223 兆赫的電視射頻信號，接收信號場強比無建築物時低 16 dB，對於 900 兆赫的手機射頻信號，接收信號場強比無建築物時低 22dB .

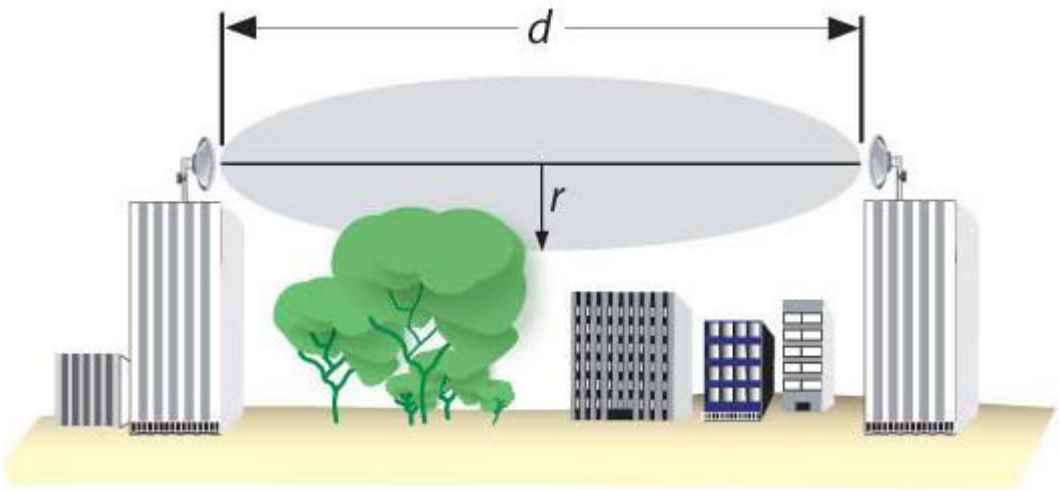
如果建築物高度增加到 50 米，則在距建築物 1000 米以內，接收信號的場強都將受到影響而減弱。也就是說，頻率越高、建築物越高、接收天線與建築物越近，信號強度與通信品質受影響程度越大；相反，頻率越低，建築物越矮、接收天線與建築物越遠，影響越小。

因此，選擇基站場地以及架設天線時，一定要考慮到繞射傳播可能產生的各種不利影響，注意到對繞射傳播起影響的各種因素。

菲涅爾區(Fresnel Zone)

Fresnel 區是一個視線區域的無線電波分佈範圍，這個區域必須無障礙，否則信號強度會被削減。例如 在一個 16 公里使用 5.8G 連接的無線，60%的 fresnel 區是一個 8.7 米的圓球區，在 2.4GHz 同樣的距離是 13.6 米。點對點和點對多點的環境儘量選擇周圍的制高點，在無線覆蓋如社區、城市等，將來尋找障礙較少的網站，這是為增加視距傳播，增加覆蓋距離和範圍，以便減少網站數量。

在收發天線之間連一條線，以這條線為軸心，以 R 為半徑的一個類似於管道的區域內，沒有障礙物的阻擋。如圖所示，這個管道稱為菲涅爾區(Fresnel Zone)，菲涅爾區是一個橢球體，收發天線位於橢球的兩個焦點上，



例如：在一個 16 公里使用 5.8G 連接的無線，60%的 fresnel 區是一個 8.7 米的圓球區，在 2.4GHz 同樣的距離是 13.6 米。

板狀天線高增益的形成

為提高板狀天線的增益，還可以進一步採用八個半波振子排陣 前面已指出，四個半波振子排成一個垂直放置的直線陣的增益約為 8 dB；一側加有一個反射板的四元式直線陣，即常規板狀天線，其增益約為 14 ~ 17 dB 。 一側加有一個反射板的八元式直線陣，即加長型板狀天線，其增益約為 16 ~ 19 dB . 不言而喻，加長型板狀天線的長度，為常規板狀天線的一倍，達 2.4 m 左右。

高增益柵狀拋物面天線

從性能價格比出發，人們常常選用柵狀拋物面天線作為主天線。由於拋物面具有良好的聚焦作用，所以拋物面天線集射能力強，直徑為 1.5 m 的柵狀拋物面天線，在 900 兆頻段，其增益即可達 G = 20 dB . 它特別適用於點對點的通信，例如它常常被選用為直放站的施主天線。

拋物面採用柵狀結構，一是為了減輕天線的重量，二是為了減少風的阻力。

拋物面天線一般都能給出 不低於 30 dB 的前後比 ，這也正是直放站系統防自激而對接收天線所提出的必須滿足的技術指標，當然使用碟形拋物面天線效果會比柵狀拋物面天線好，但成本會高出許多。

3.3 天線類型

看看 WLAN 常用的幾種天線類型，我們可以分為全向、磁區和定向幾種天線：

2.4/5G 全向天	2.4/5G 平板磁	2.4G 拋物面柵	5G 拋物面柵格	5G 拋物面碟型

線	區天線	格天線	天線	天線
				
用於全向覆蓋	可用於扇形區域的覆蓋或者點對多點傳輸	用於定向的 2.4G 點對點傳輸	用於定向的 5G 點對點傳輸	用於遠距離定向的 5G 點對點傳輸
2.4/5G 雙極化天線	2.4/5G 集成外殼天線	2.4G 室內覆蓋吸頂天線		
				
11n 的高頻寬傳輸	將設備集成在天線內	用於室內 WiFi 覆蓋		

點對點安裝

在安裝點目測周圍環境，判斷遠端目標的大概方向，天線對準目標，然後設備啟動後，如果設備啟動後未發現信號，可通過逐步轉動天線尋找信號，當發現信號後，雙方根據信號強度做微調，直到達到最好的信號強度為止。

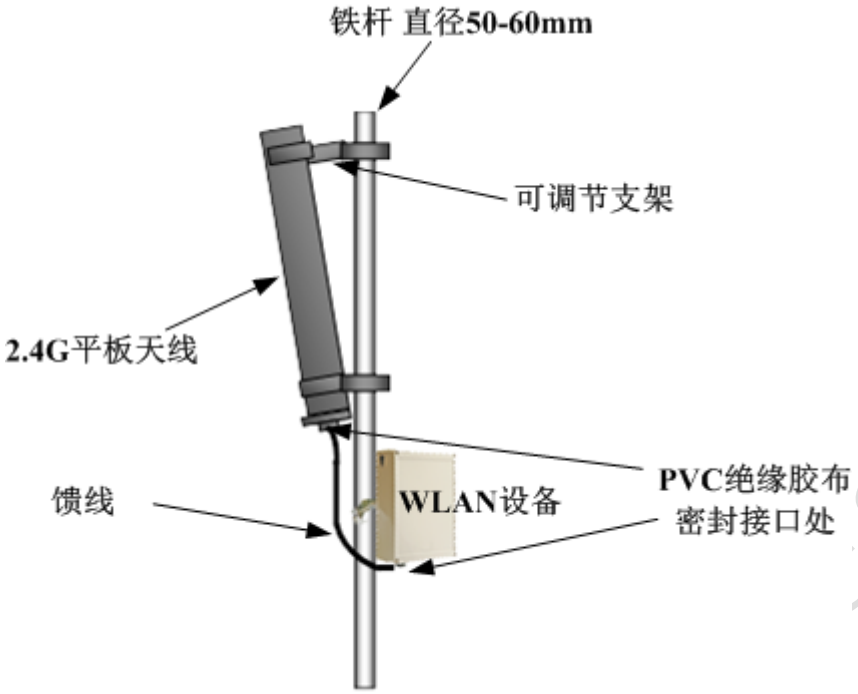
覆蓋安裝

- 天線高度：天線過高會降低天線附近的覆蓋電平（特別是垂直向下的）俗稱“塔下黑”，特別是全向天線最為明顯
- 天線高度過高容易造成越區覆蓋問題，影響網路品質。
- 天線方向角：天線主播瓣方向指向高使用者人群區域，可以加強該區域的信號強度。
- 相鄰天線的交叉覆蓋不宜超過 10%
- 需要考慮天線的俯仰角度，特別在高處調整俯仰調度，有利於附近平地的信號接收。

天線安裝

室外天線安裝需要相應的配套設施，確保天線在室外不受各種環境影響，如下圖的 2.4G 平板天線安裝

- 需要直接 50-60mm 的鐵杆，通過 U 形卡固定天線
- 使用相應高品質的饋線連接天線和 WLAN 設備
- WLAN 設備固定在天線的下方，儘量將設備安裝在較低位置，通過饋線連接
- 天線接頭和 WLAN 設備接頭通過 PVC 絕緣材料或者膠布包裹密封



室外設備實際安裝示意圖：

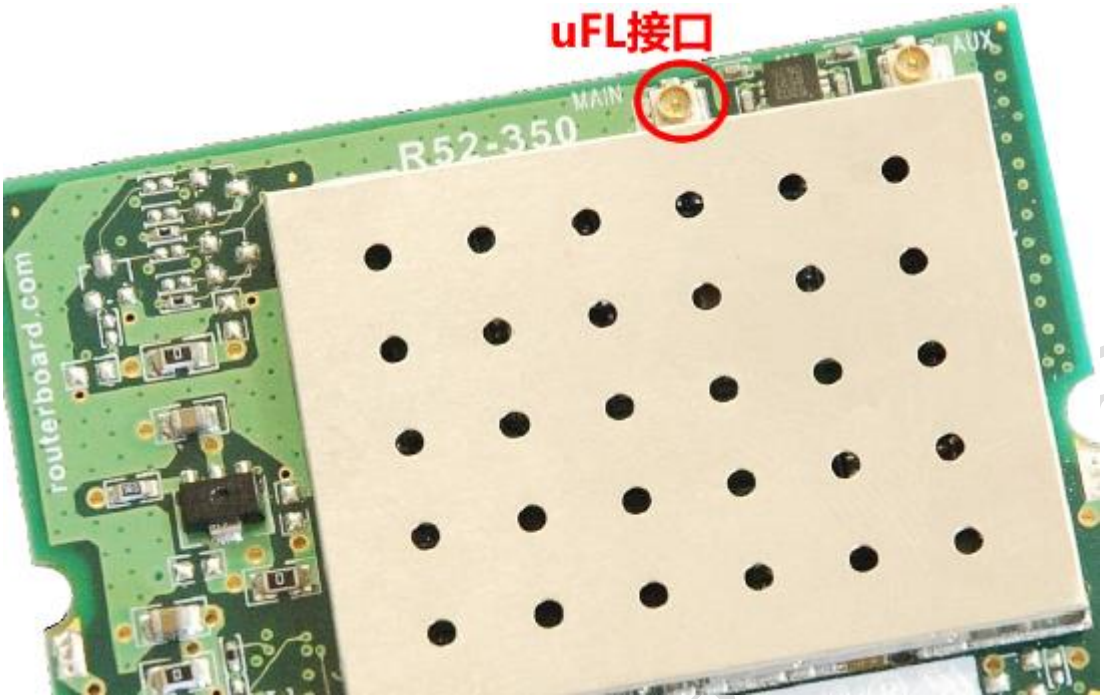
3.4 接頭與線材類型

這裡介紹下設備常用的接頭與線材類型，有助於你面對各種設備或網卡與天線連接時，能正確的選擇線材和介面類別型。

1、無線網卡介面，以及可能會涉及到的線材接頭類型：

無線網卡介面類別型	線材類型 1	線材類型 2
uFL 接頭	uFL to SMA 跳線	uFL to N 跳線
MMCX 接頭	MMCX to SMA 跳線	MMCX to N 跳線

下圖是無線網卡的 uFL 接頭：

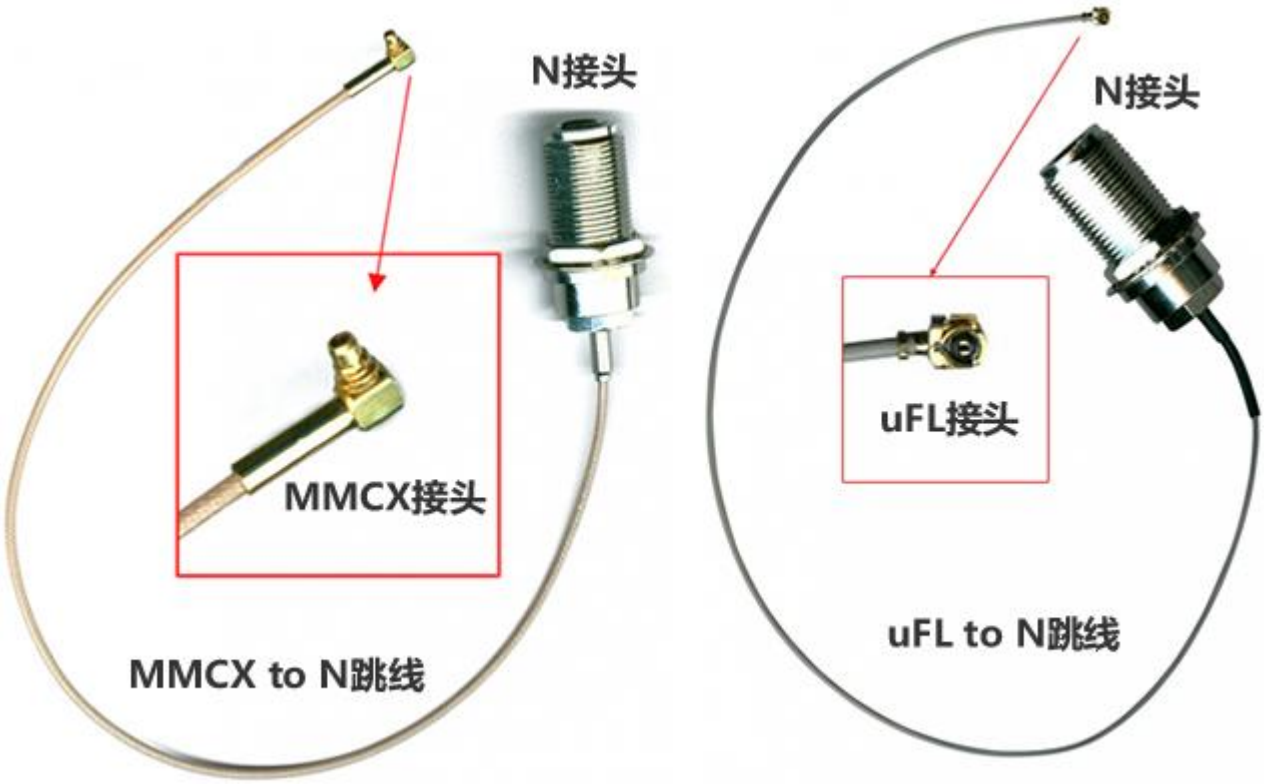


下圖是無線網卡的 MMCX 接頭

2、認識無線網卡的接頭，現在認識下從無線網卡連接到天線的介面和線材

天線介面或線材連接頭	線材類型 1
SMA 型（公頭，母頭）	SMA to N 饋線
N 型（公頭，母頭）	N to N 饋線

下面是 MMCX to N 跳線和 uFL to N 跳線，主要用於無線網卡與 N to N 饋線連接



下面是 SMA 接頭（公頭與母頭）和 SMA 的饋線：

下面是 N to N 饋線，主要用於設備和天線連接：

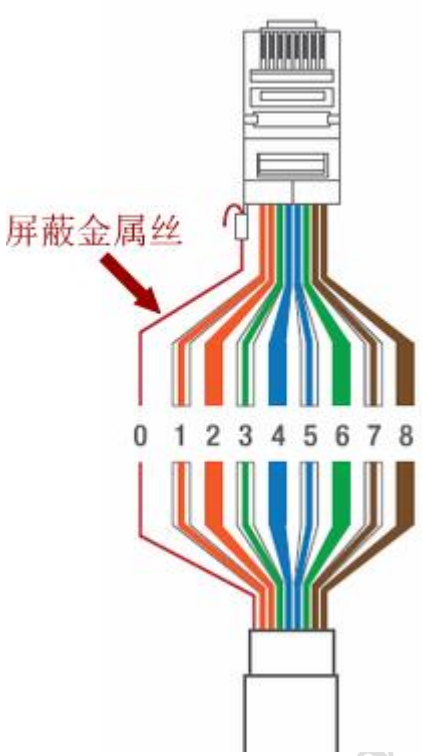
3.5 RouterBOARD 設備接地

遮罩雙絞線

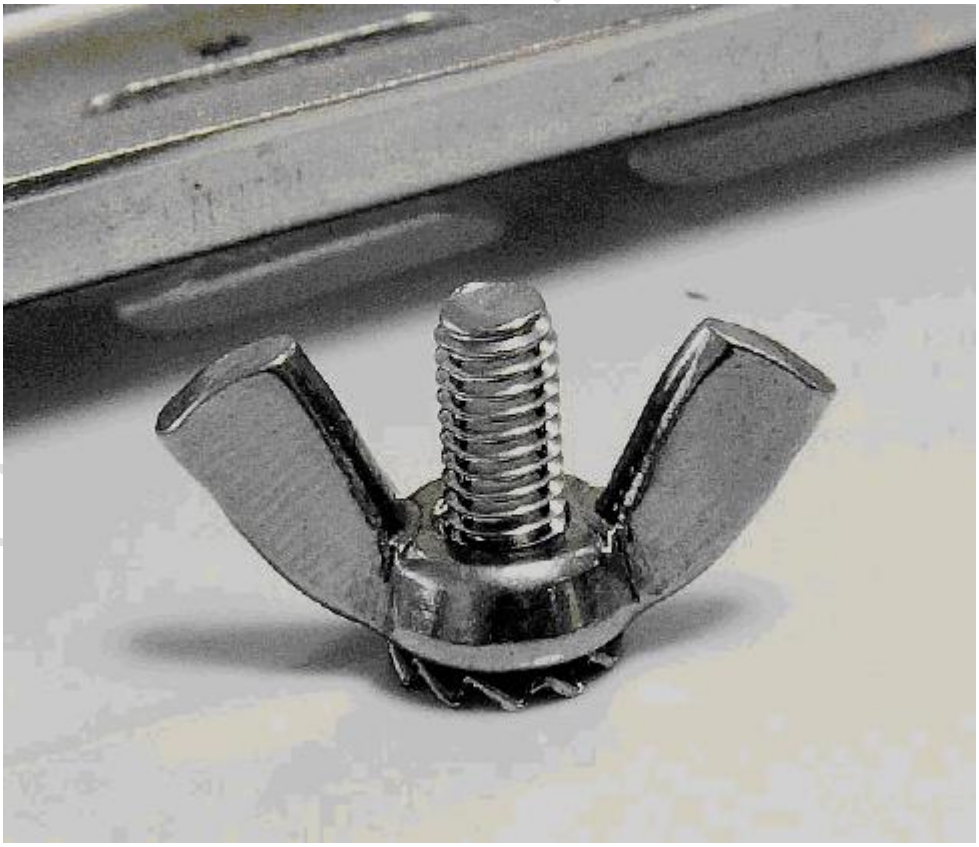
室外無線安裝在建築物頂端或鐵杆上，天線和無線設備都應當適當的接地，天線避雷器必須安裝在外接的饋線上（靠近天線或者天線接頭位置）防止設備損壞。注意天線避雷器如果沒有接地將沒有任何作用。

使用 7mm 直徑（1 AWG）的金屬線包裹在耐腐作材料接地。需要確定你接地是有效的，即需要連接到建築物或者附近的接地設施，例如：樓房的避雷針（樓房主水管也有一定的接地作用），如果是小設備可以選用較細的金屬線，RB 設備的遮罩雙絞線接地有以下要求

1、室外安裝要求使用室外的遮罩雙絞線，也要選擇遮罩的 RJ45 水晶頭或者帶有接地線介面的 RJ45 水晶頭



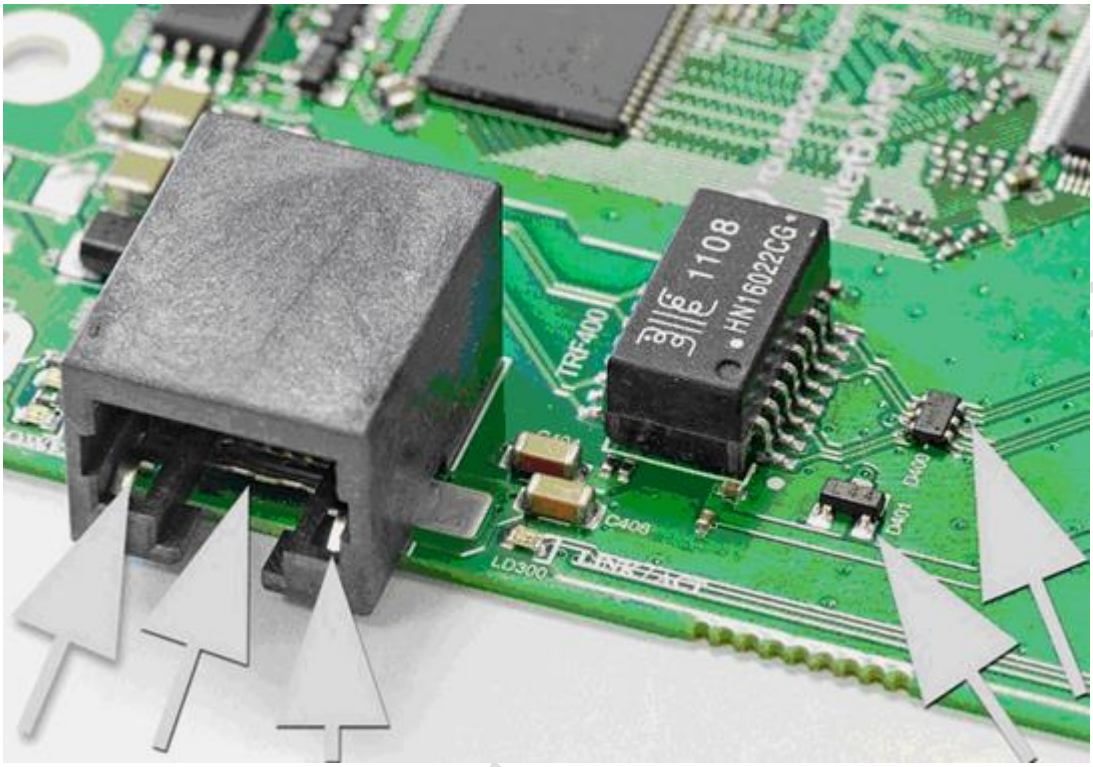
2、接地線還要連接到 RouterBOARD 上，通過 RouterBOARD 的螺絲孔位與機殼的孔位元連接，一般金屬機殼螺絲孔位元是直接可以接地的,塑膠室外機殼金屬孔位元都會與外部的接地金屬片相連。



接地線應連接到室外殼的接地金屬螺絲上

3、以太網避雷器不建議使用，需要考慮是否影響 POE 連接，以太網避雷器可能影響 POE 使用，造成 POE 電壓衰減。如果使用以太網避雷器請放置在室外，並且和 RouterBOARD 接地線一起接地主要考慮。

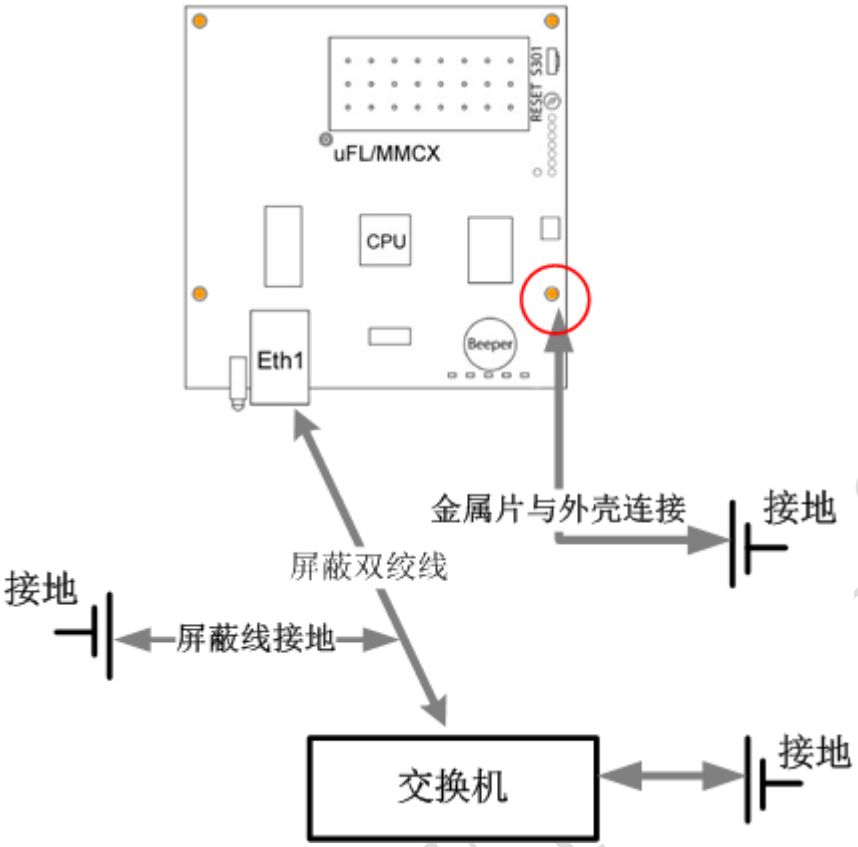
RouterBOARD 的 ESD（防靜電）保護



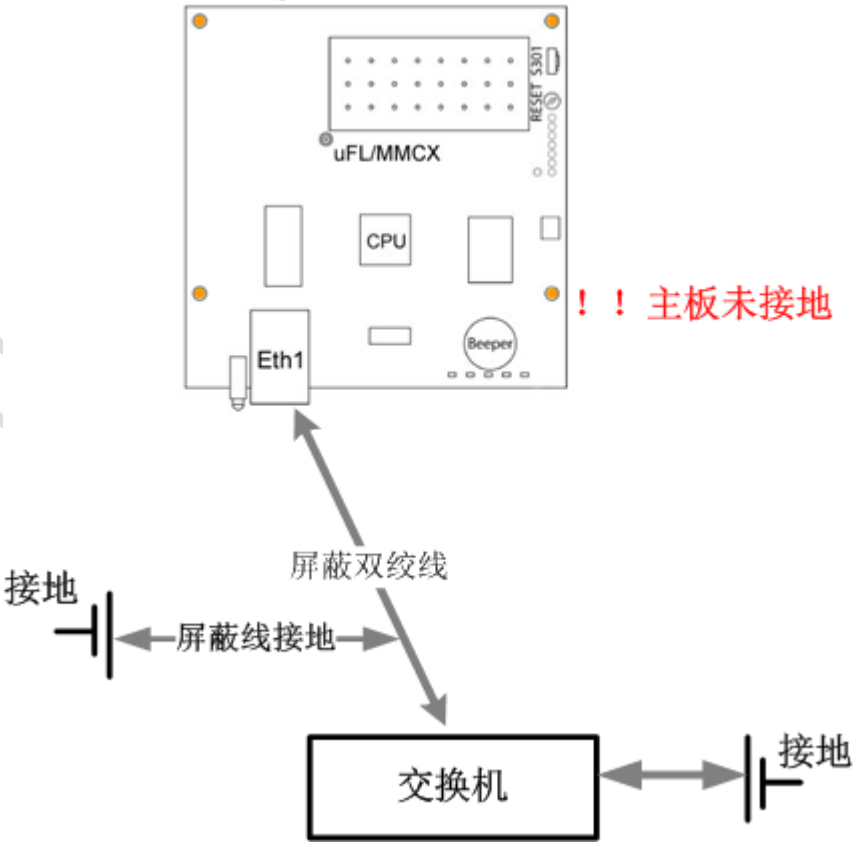
1. 三個箭頭標記了以太網口的接地金屬片，RouterBOARD 以太網口左右兩邊的 2 個金屬片，用於遮罩雙絞線接地
2. 中間箭頭指向的金屬片是主機板的接地片，主機板通過螺絲安裝孔金屬與外殼的接地相連接（如果你外殼安裝孔位沒有接地設計，你可以用一條金屬線連接相應的接地設施）
3. 在圖的右側兩個箭頭為 ESD 防靜電保護晶片，如果在沒有使用遮罩雙絞線時，ESD 晶片可以保護 CPU 和其他部分主機板部件

如果僅使用遮罩雙絞線保護有效性很低的，這樣主機板自身沒有接地。你需要兩種方式都採用，才能有效保護設備，下面有兩種方式，推薦使用第一種方式：

方式一（遮罩雙絞線 + 設備接地）：



方式二：僅有遮罩雙絞線：



注意！即使你沒有將無線設備接地，僅使用了遮罩雙絞線，但你應將雙絞線連接對端設備交換機、PC 或 RouterBOARD 等室內設備接地。

這裡說明下接地能有效降低室外雷擊或感應電損壞的幾率，但也不是 100%保證不被損壞，但有效地保護是必須的。

3.5 WLAN 網卡和饋線損壞檢測

控制 WLAN 信號接收發送、處理和解碼是由 WLAN 晶片完成，如 AR5414，AR9220 等，功率放大器（PA）是將 WLAN 信號放大元件，是將 WLAN 信號功率放大，通常我們說的無線網路功率 100mW、300mW 和 500mW 就是由功率放大器完成，功率放大器就像擴音喇叭的作用。

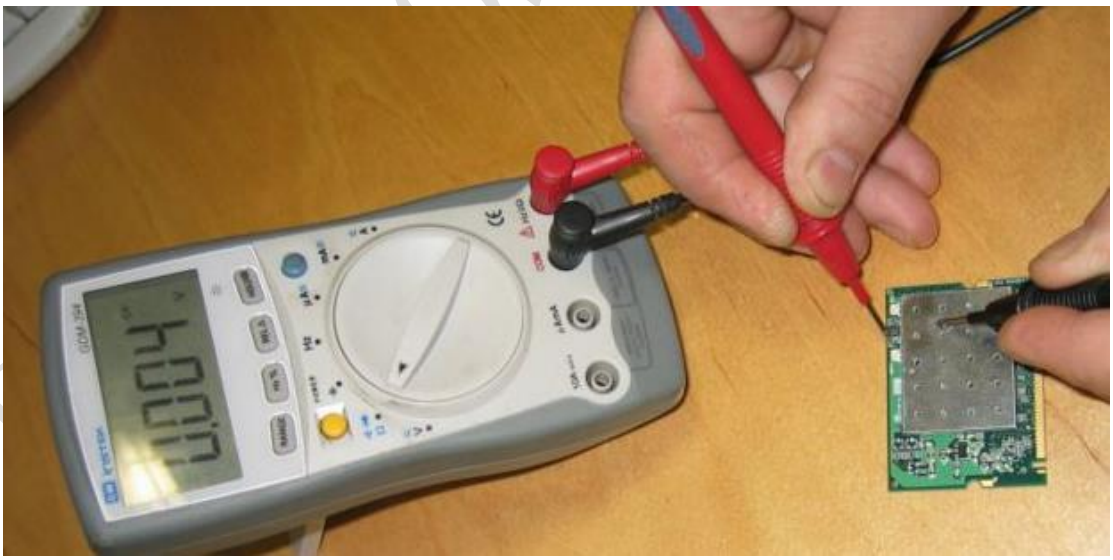
如果功率放大部分損害，直接導致 WLAN 網卡的信號發射和接收變弱，無法連接到其他 WLAN 設備。通過當把一張無線網卡插入 RouterBOARD，如果立馬發現無線網卡手感變得發燙時，可能 PA 已經損壞。 這樣的情況可能是使用者造成，也可能是出廠問題。

R52, R52Hn 和 R52H ESD 損壞測試

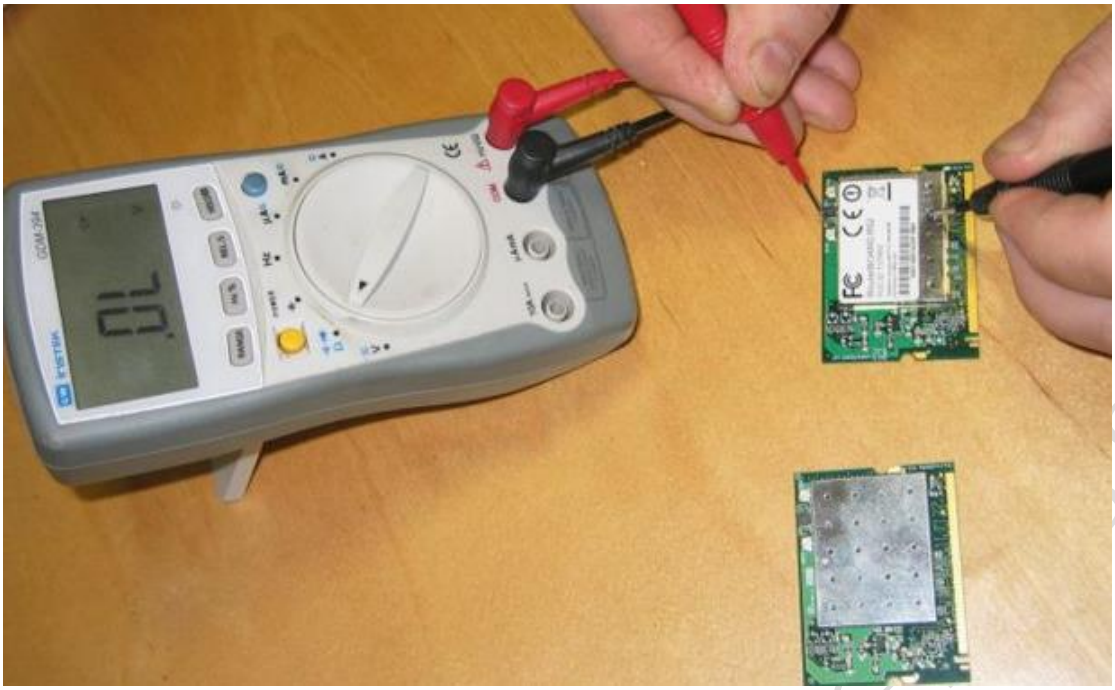
在無線網卡有一個靜電保護器，用於靜電保護，通常稱為 ESD，ESD（Electro-Static discharge）的意思是“靜電釋放”。ESD 是 20 世紀中期以來形成的以研究靜電的產生、危害及靜電防護等的學科。因此，國際上習慣將用於靜電防護的器材統稱為 ESD，中文名稱為靜電阻抗器。

通常不正確的接地會導致無線網卡在雷暴或其他靜電環境損壞。如果 R52 或 R52H 在雷暴天氣後網卡不能正常工作，可以使用萬用表檢測。下面的實例可以檢測是否損壞：

首先將萬用表調整到電阻檔，測試電阻，通過測試靜電保護器，如果發現有電阻值，說明靜電保護器已經被擊穿，即損壞，這樣情況並不能保證能修復網卡，如下圖：。



正常的網卡測試如下：這張圖的萬用表顯示的 OL（電阻無窮大），萬用表還有顯示為 1，也是同樣的表示未短路



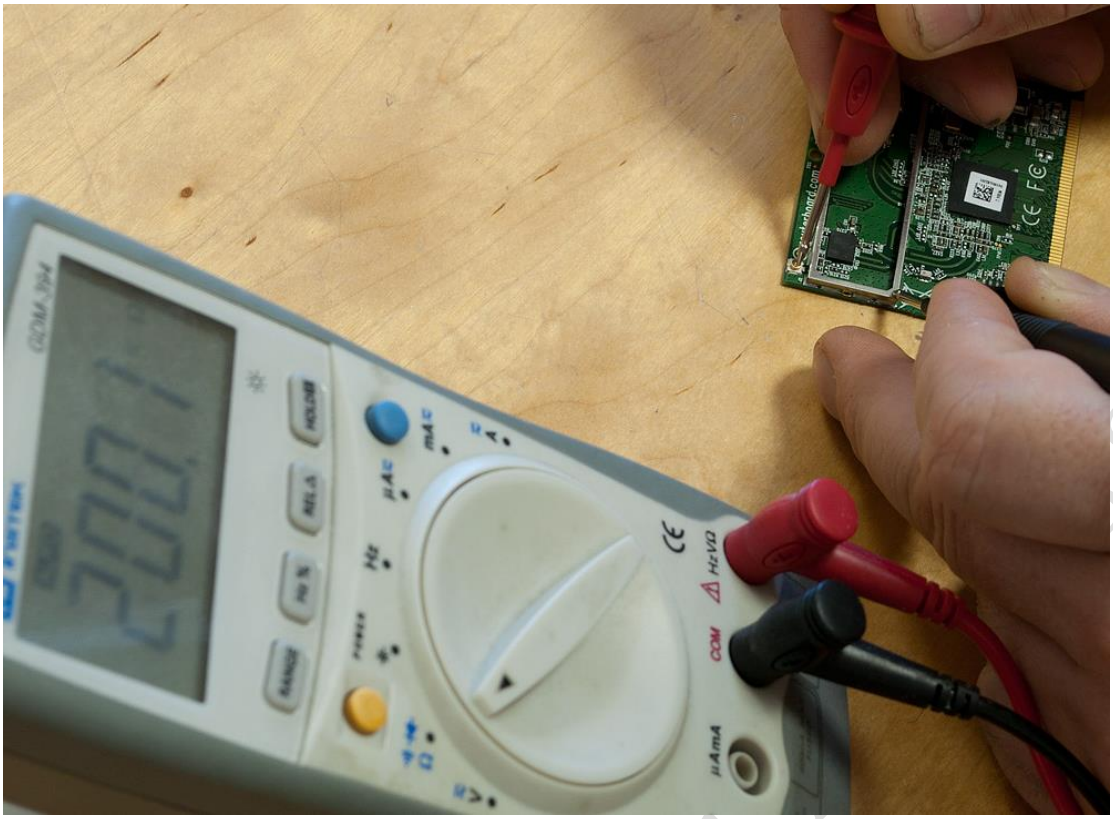
R52Hn 網卡測試 Chain 0 的位置：

R52Hn 網卡測試 chain 1 的位置：

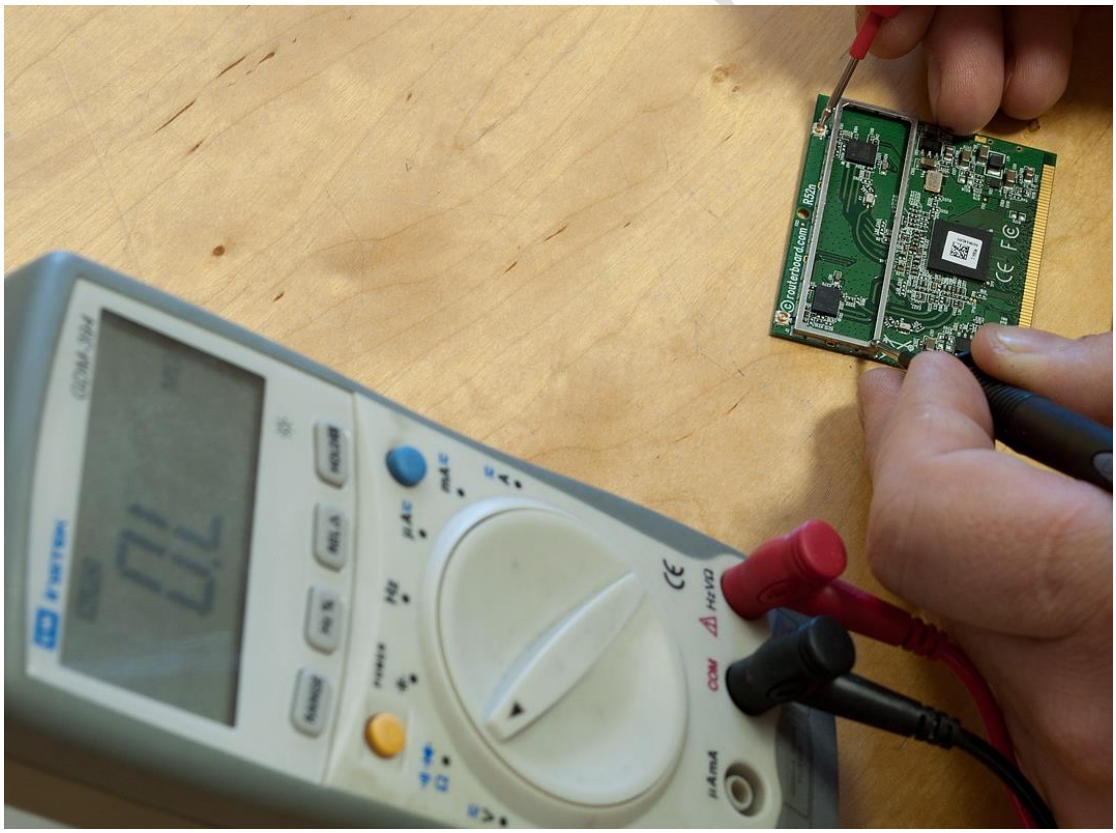
R52n 天線電路損壞測試

下面的圖片展示了如何測試天線電路是否損壞，如果電阻低於無窮大說明網卡被擊穿，這樣情況是幾乎無法修復。按照官方的意思就是沒有必要返廠維修了。

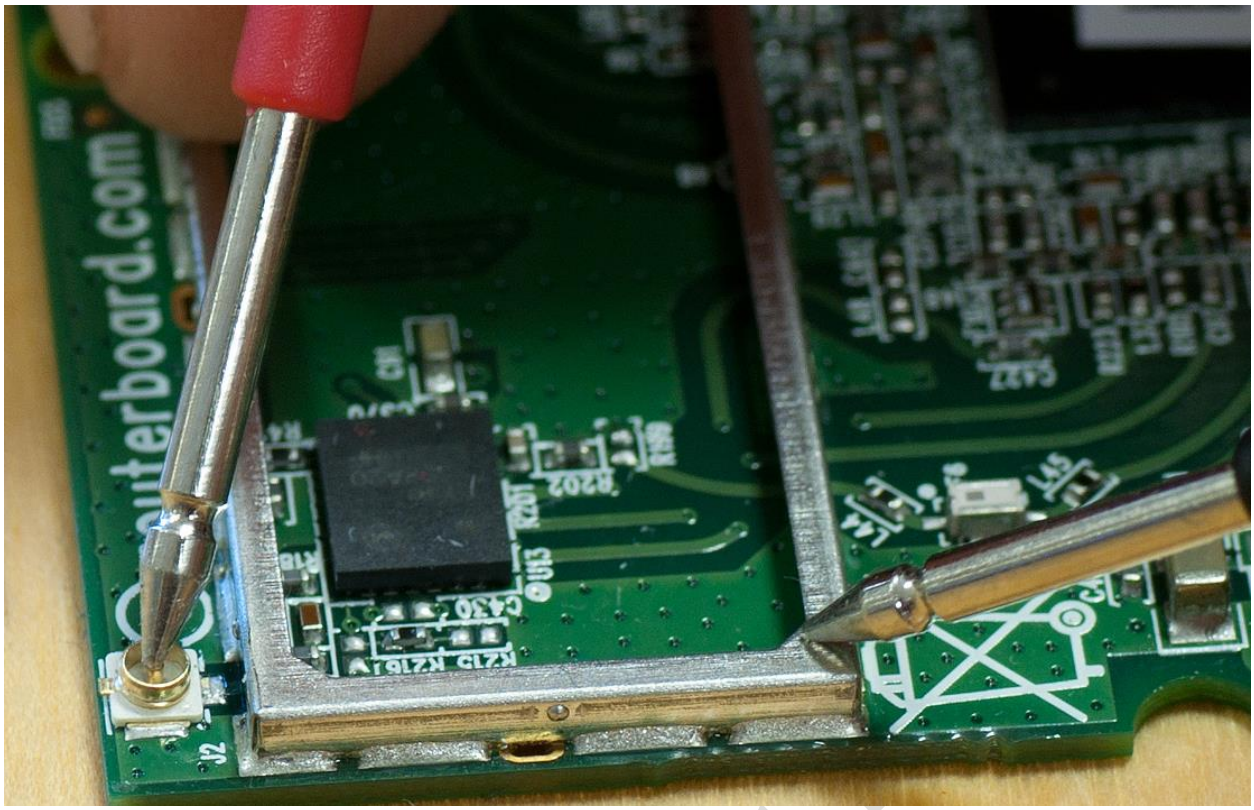
測試天線 chain 圖片，下面是損壞的無線網卡：



下面這個 chain 是正常：

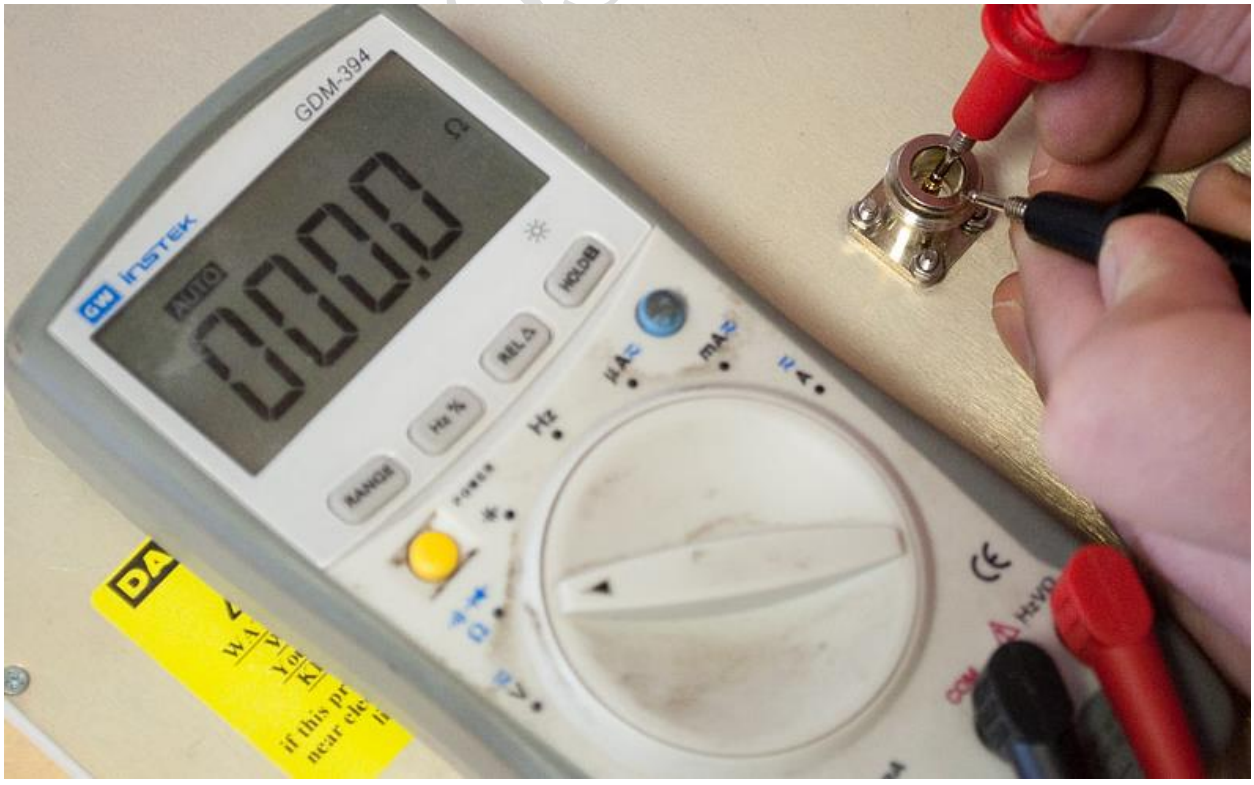


下面這張圖展示了測試區域，一支筆測試 chain 介面的芯，一支測試金屬蓋表皮

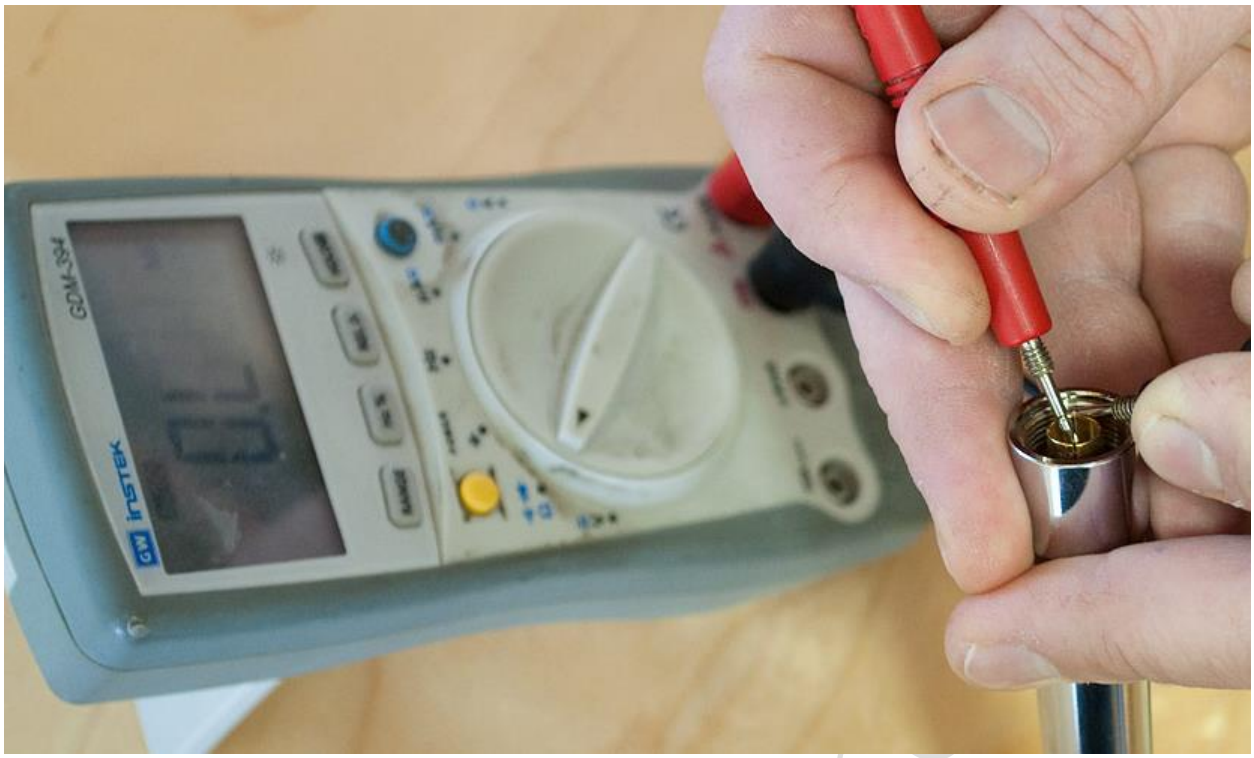


饋線短路測試

饋線用於連接天線和網卡，製作饋線或安裝時可能會造成損壞，通常測試饋線是否短路也是使用萬用表的電阻檔，測試方法類似，我們通過測試饋線接頭，如果出現如下圖，表示饋線或接頭已經短路

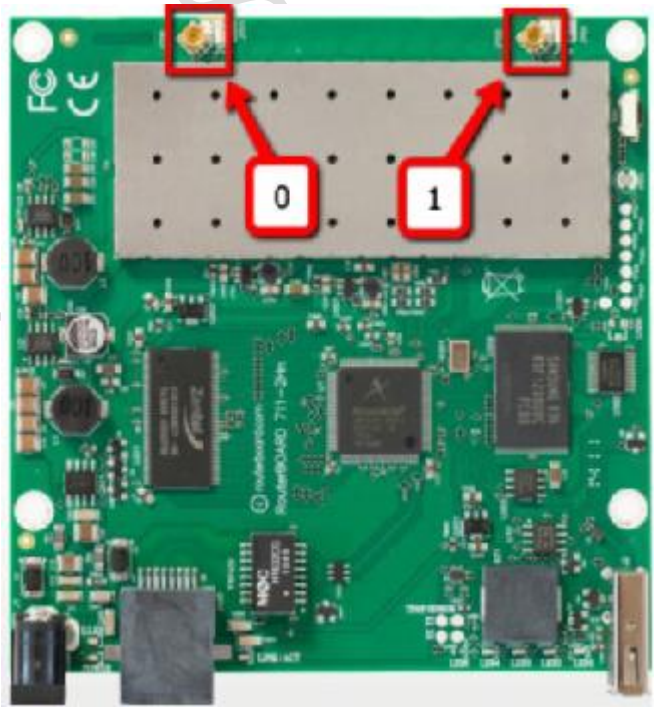


下面情況則表示饋線和接頭正常，沒有短路：

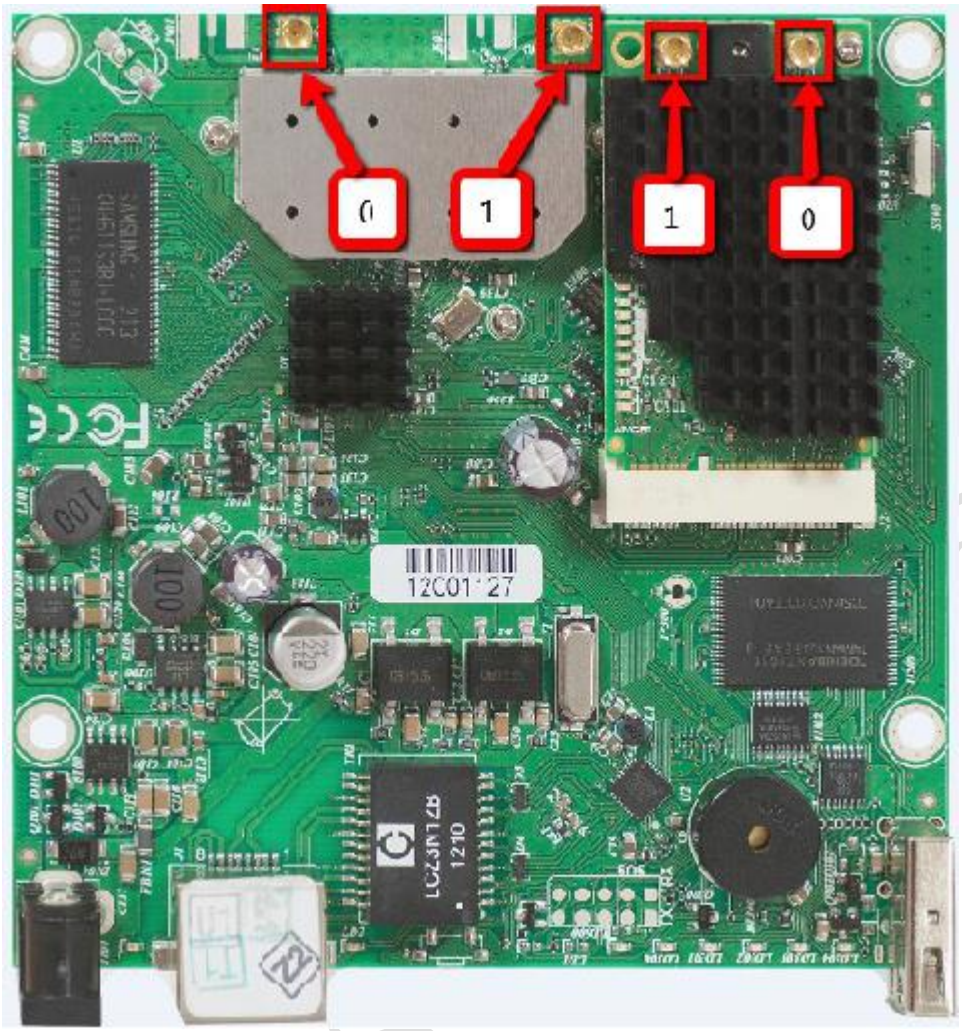


注意事項

1、RouterBOARD 第一次啟動時無線網卡預設配置腳本會啟用，11n 網卡的兩個 chains 會同時開啟，因此確保兩個天線連接頭與天線、至少要和跳線連接，避免造成功率放大器輸出損壞。注意這張 RB911 的 11n 無線網卡 chain0 和 chain1 的位置



2、下面是 RB912 上安裝了 RB11e 無線網卡，與板載的網卡 chain0 和 chain1 是相反的



3.6 RouterOS 支援的無線網卡

RouterOS 主要以支援 Atheros 廠商晶片為主，也支援部分其他廠商的晶片，所以你在挑選網卡是，一定要注意無線網卡的晶片。下面是 RouterOS 支援的大部分網卡晶片驅動（網卡資料僅供參考），如果需要使用 Superchannel 功能都必須採用 Atheros 廠商的晶片。

這裡主要介紹 Atheros 晶片的無線網卡，當前市面上主要流行的 abg 網卡主要是 AR5212 和 AR5414 兩種晶片，AR5414 在功耗和性能方面要優於 AR5212。abgn 方面主要以 AR9220 系列為主。Atheros 晶片提供無線信號的處理，而發射功率是由各個 OEM 工廠生產時，所使用的功率放大控制器決定，所以無線網卡的發射功率不是由晶片決定，我們需要看每個 OEM 廠商的資料確定具體參數。

下面是一張 MikroTik 的 R52Hn 的無線網卡，採用的是 AR9220 的晶片，支援 802.11abgn 屬於雙頻無線網卡，當然只能同時工作在一個頻率上，要麼 bgn 模式，要麼 an 模式。

下面是 R52Hn 參數的簡單介紹

R52Hn	
802.11a	支持
802.11b	支持
802.11g	支持
802.11n	支持
無線介面	MMCX
介面類別型	miniPCI
晶片	AR9220
輸出功率	25dBm @ abgn
5GHz	4920 - 6100GHz（5MHz step）
2GHz	2.192 - 2.539GHz (5 MHz step)
重量	20g
工作溫度	-50℃ to +60℃

這些參數是一張網卡的基本資訊，如果要看他的具體性能，我們還要瞭解他的發射功率和接收靈敏度的資料，如下

下面是在不同協議下的接收靈敏度和發射功率的頻寬情況

在 802.11b 協定下 1Mbit 頻寬要求的信號接收靈敏度為-93dBm，而此時發射功率可以達到 24dBm；11Mbit 情況下同樣是收靈敏度為-93dBm，而此時發射功率可以達到 24dBm。

802.11b	RX Sensitivity 接收靈敏度	TX Power 發射功率
1Mbit	-93	24
11Mbit	-93	24

在 802.11g 協定下 6Mbit 頻寬要求的信號接收靈敏度為-94dBm，而此時發射功率可以達到 25dBm；54Mbit 情況下同樣是收靈敏度為-81dBm，而此時發射功率可以達到 22dBm。

802.11g	RX Sensitivity 接收靈敏度	TX Power 發射功率
6Mbit	-94	25
54Mbit	-81	22

在 802.11bgn 下 MCS0 20MHz 為-94dBm，發射功率可以達到 25dBm，以下依次類推

802.11bgn 2.4GHz	RX Sensitivity 接收靈敏度	TX Power 發射功率
MCS0 20MHz	-94	25
MCS0 40MHz	-92	24
MCS7 20MHz	-78	21
MCS7 40MHz	-75	20

802.11a 協定情況下

802.11a	RX Sensitivity 接收靈敏度	TX Power 發射功率
6Mbit	-97	25
54Mbit	-80	21

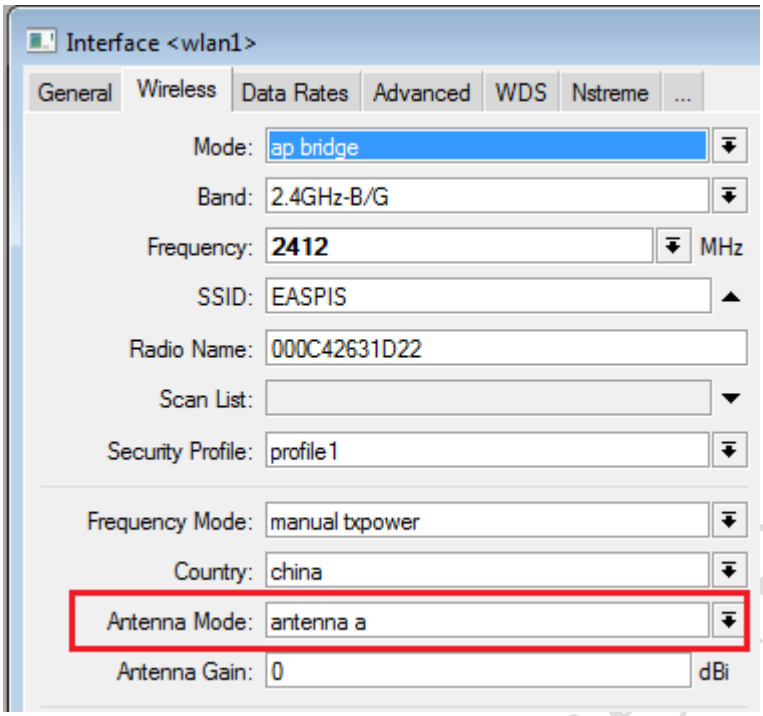
802.11an 協定情況下

802.11an 5GHz	RX Sensitivity 接收靈敏度	TX Power 發射功率
MCS0 20MHz	-97	24
MCS0 40MHz	-92	22
MCS7 20MHz	-77	18
MCS7 40MHz	-74	17

無線網卡都是時分工作方式，即接收和發送不是在同一時間完成的，是把時間劃分開，一段時間用於發射信號，一段時間用於接收信號，即半雙工模式。802.11abg 網卡和 802.11n 的網卡在介面上有所不同，一般情況下不管是 11abg 網卡，還是 11n 網卡介面都會是 2 個，但他們有所不同。

11abg 代表網卡是 AR5414，介面都有標明 MAIN 和 AUX，即一個主介面，一個輔助介面，預設情況是網卡採用 MAIN 發生和接收信號，AUX 是不工作的，如下圖

11abg 網卡是採用 MAIN 發送還是 AUX 發送，我們可以在 RouterOS 中設置，預設是採用 MAIN 介面。如下圖在 RouteroS 無線網卡中的 wireless 參數中的 antenna Mode 中設置，默認是選擇 antenna-a，即 MAIN 介面，如果選擇 antenna-b 則是 AUX 介面



11n 網卡的區別

而 11n 的網卡不同，因為採用了 MIMO 技術，所以分為 1×1 和 2×2，甚至 3×3 模式，即 150Mbit、300Mbit 和 450Mbit 的區別。如果 11n 有 1 個介面代表 1×1，有 2 個介面就是 2×2，每個介面都是一個獨立的信號源，不會像 11abg 網卡區分 MAIN 和 AUX 的介面，只能由一個介面發射和接收信號。

無線網卡功率換算

通常我們可以看到無線網卡發射功率有 dBm 和 mW（毫瓦），即增益和功率兩個參數，不過這兩個參數是可以換算的，如下面是 dBm 和 mW 對照表

dBm	mW		dBm	mW
0	1.0 mW		26	400mW
1	1.3 mW		27	500mW
2	1.6 mW		28	640mW
3	2.0 mW		29	800mW
4	2.5 mW		30	1.0W
5	3.2 mW		31	1.3W
6	4.0 mW		32	1.6W
7	5.0 mW		33	2.0W
8	6.0 mW		34	2.5W
9	8.0 mW		35	3.0W
10	10 mW		36	4.0W
11	13 mW		37	5.0W
12	16 mW		38	6.0W
13	20 mW		39	8.0W

14	25 mW		40	10W
15	32 mW		41	13W
16	40 mW		42	16W
17	50 mW		43	20W
18	64 mW		44	25W
19	80 mW		45	32W
20	100 mW		46	40W
21	128 mW		47	50W
22	160 mW		48	64W
23	200 mW		49	80W
24	250 mW		50	100W
25	320 mW		60	1000W

3.7 RouterOS WLAN 構建常見問題

1、我應該把中心 AP 放在那裡？

中心 AP 應該被放在一個地區的制高點，使得周圍的用戶都在視距範圍內，例如高層建築的屋頂，鐵塔等

2、構建一個中心基站需要什麼？

中心基站設備組成包括 MikroTik 無線路由器，全向天線或者磁區天線與設備連接的饋線、電源等，MikroTik 路由器連接有線網路， 路由器配置為橋接模式，用於連接無線和有線網路，通過全向天線將信號發送到周邊的客戶。

3、一個中心基站能連接多少個用戶端？

支援 2007 個用戶端，然而實際情況並不是如此，需要根據系統的性能和承載能力。實際環境中終端 PC 的數量，頻寬情況和信號連接狀態都會影響，802.11a 下支持 20-30 個左右客戶比較合適，802.11bg 下接入端最好在 10-20 個用戶端，如果你通過流量控制和資料過濾就能更好的對他們進行管理。

4、我需要連接一個用戶端的網路，應該怎麼做？

你需要一個用戶端設備（CPE），例如一個 MikroTik 無線路由設備、以太網介面、定向天線、低損耗饋線。MikroTik 無線路由器可以為本地用戶端的網路提供需多功能，如防火牆、NAT、流量控制、DHCP 服務等。定向天線應該安裝在可以看見中心基站的位置。

5、每個系統的傳送速率如何？

RouterOS 支援 802.11abgn 無線傳輸協定，2.4GHz 在 802.11b 模式下，資料傳輸是 11Mbps。然而實際輸送量在 5-6Mbps。5GHz 在 802.11a 模式和 2.4GHz 的 802.11g 模式下，資料傳輸為 54Mbps。

5GHz 的 802.11a 模式下，為得到理想的頻寬，在中心基站和用戶端最好使用 800MHz 的 CPU。同樣 RouterBOARD 系列建議使用 400 系列和 600 系列。所有用戶都可以分配到相同的頻寬。

6、能否限制每個用戶的頻寬？

是的，可以限制每一個使用者的頻寬速度，通過 RouterOS 的 queue 選項，如果你是 bridge 橋接無線網卡和有線網卡，請將 bridge setting 裡的 use-ip-firewall 選項開啟。

7、中心基站與用戶端之間無線傳輸最大距離能達到多少？

最大距離和天線、饋線、傳輸功率和信號接收的靈敏度、周圍環境和天線安放的位置等有關係。
在 2.4GHz, 中心基站和用戶端通常不會超過 10-12km.
在 5GHz,我們已測試中心基站使用 17dBi 平板天線，用戶端使用 30dBi 圓盤拋物面天線連接距離在 25 公里，實際傳輸速率 10Mbps

8、我能否從設備使用更長的饋線連接到天線？

可以，但無線傳輸距離和信號會受到影響。

9、我是否使用功率放大器增加距離？

可以，功率放大器有增強功率的作用，增加傳輸距離。同樣他也可以連接饋線，增加饋線的傳輸距離。

10、無線連接是否要求在視線範圍內？

是的，視距範圍內總是被需要的，直接能看到對方，即兩個連接點中間不能存在障礙物。

11、什麼是 Fresnel 區？

Fresnel 區是一個視線區域的無線電波分佈範圍，這個區域必須無障礙，否則信號強度會被削減。例如 在一個 16 公里使用 5.8G 連接的無線，60%的 fresnel 區是一個 8.7 米的圓球區，在 2.4GHz 同樣的距離是 13.6 米。

12、我是否可以將兩個無線網路橋接？

是的，能使用 MikroTik 無線路由器建立透明橋接在兩個設備間，具體可查看 RouterOS 技術文檔。

13、安裝一個 Wlan 無線系統需要多長時間？

一個基本的無線系統，如包含 3-5 個用戶端的系統，在人員足夠的情況下需大概 1-2 天時間

14、Wlan 運行在 Station 模式下是否能做橋接？

不能，station 模式不支援橋接功能，station 應用於三層的 IP 通信連接。

15、Wlan 橋接模式一般使用哪種？

一般 RouterOS 的橋接模式選擇 ap-bridge 對 station-wds，需要開啟 WDS 選項，並設置預設的橋接參數。

16、802.11n 能使用 wds 模式嗎？

在 5.0 前 RouterOS 對 802.11n 僅支援 EoIP 隧道的傳輸模式，在 5.0 後啟用 Nv2 協定後支援高頻寬傳輸的 WDS 模式。

17、RouterOS 最大的 5G 傳輸頻寬能達到多少

我們所測試到的單網卡，最大單向頻寬在 5G-Turbo 模式下，可以達到 75Mbps，雙向頻寬在 40Mbps 左右，當然使用 802.11n 的 5G 傳輸，可以獲得更高的頻寬，合適的環境和設備下可以得到近 200Mbps 的

頻寬。

18、mode=bridge 模式支援那種連接方式

採用 bridge 模式只能支援與 ap-bridge、station-wds 和 bridge 連接的通信，即只支援點對點無線連接，如果你採用 RB411 無線設備 RouterOS 是 L3 級，那麼 2 個 RB411 點對點通信只能使用使用 bridge 模式。在 5.0 版本後出現的 station-bridge 模式也是可以和 bridge 通信

19、什麼是 Nstreme

Nstreme 是 MikroTik 獨立開發的一套無線傳輸協議，是將多個幀進行重組，即將資料量較小的幀重新組合成大的幀進行轉發，提高資料傳輸的效率，有助於 Wlan 無線傳輸頻寬的提升，5.0 後 Nstreme 改進版本 Nv2（Nstreme version2）採用 TDMA 技術有效的支援了 11n 的高頻寬傳輸。

20、什麼是 Nstreme Dual

MikroTik 開發的雙向傳輸協定，即每個設備採用兩個無線模組，一個無線模組做 tx（發送），一個無線模組做 rx（接收）把資料接收發送分離成兩個無線傳輸的方式，有助於提高無線傳輸的頻寬和效率。

21、WLAN 與 WiFi 區別

WLAN 是 Wireless Local Area Network 的縮寫，指應用無線通訊技術將電腦設備互聯起來，構成可以互相通信和實現資源分享的網路體系。無線局域網本質的特點是不再使用通信電纜將電腦與網路連接起來，而是通過無線的方式連接，從而使網路的構建和終端的移動更加靈活。Wi-Fi(WirelessFidelity)，無線保真技術與藍牙技術一樣，同屬於在辦公室和家庭中使用的短距離無線技術。WI-FI 是 WLAN 的一個標準 WLAN 是無線局域網，無線局域網是由無線設備構成的，包括無線路由器或其他發射裝置以及各種例如筆記本、平板電腦、手機等網路終端，設備之間是通過 WiFi 無線技術連接的。

22、RB751U

RB751U 集成一個 2GHz 802.11bgn 無線網卡，並內置了 PIF 2.5dBi 天線。同時提供一個外接天線的 MMCX 接頭。由於該設備提供內置和外置天線，所以需要特別說明天線在 wireless 選項中的配置如下：

- Chain0

one antenna for TX
one antenna for RX

- Chain1

one antenna for TX/RX
MMCX 外接天線介面

如果啟用 MMCX 介面，需設置天線模式為 antenna-b，在 wireless HT 菜單下禁用內置的 Chain1 天線

3.8 WiFi 覆蓋

WiFi 覆蓋上網，是現在主流，三大運營商、其他小運營商、公共場所、咖啡廳、酒店、高校等等，都在用 WiFi，還有一些餐飲行業演變的無線點餐系統，都在使用 WiFi 覆蓋，大多採用的是 802.11bg 協定，而 802.11bgn（基於 2.4G）也在逐步增長。

對於 WiFi 覆蓋我們採用的是 802.11bgn 協定，即 2.4G 頻率（2412MHz~2462MHz，共 11 頻道，再加上另外 3 個非中國標準的頻道 2467，2472，2484），就算頻率有 14 個頻道，也只有 3 個不相互干擾的頻道，如此少的乾淨頻道，根本不能滿足現在的需求，再加上環境的頻率干擾，對 WiFi 品質造成極大的影響。這個問題在 2003 年後的成都已經比較嚴重，2.4G 的資料傳輸在城區傳輸很困難，雖然覆蓋相對好些，但用戶多了也凸顯問題。

當然我們解決這個問題，可以選擇的最簡單的方法就是加大設備的發射功率，蓋過其他的設備和干擾，但這個方法也被大多廠商採用，自然設備之間的抗干擾就此抵消。其他廠商就開始尋找其他方法，例如智慧天線和修改 802.11 協定的訪問（CSMA/CD）為 TDMA 等，我分析下 3 種特點

發射功率

我們傳統意義上，認為設備發射功率高，可以提高傳輸距離和覆蓋範圍，但我們緊緊理解到的是發射功率，忽略了接收靈敏度，即設備接收到使用者端的信號強度，這個參數很關鍵，我們雖然加大了發射功率，用戶端收到了，但回傳給設備，設備卻無法收到，就像一個人說話加了擴音器聲音特別大，但下面人說話他卻什麼都聽不到，一樣的沒用，我們不僅要提高功率，還要注意接收靈敏度，RouterOS 上大多是擴展網卡，我們需要選擇好的無線網卡，既要看發射功率，又要看接收靈敏度，至於網卡參數就看廠商是否憑良心說話了，還需要自己對比測試就知道

下面是R52H的參數

協議	輸出功率	接收靈敏度
IEEE 802.11a:	24dBm	-90dBm @ 6Mbps
	19dBm	-70dBm @ 54Mbps
IEEE 802.11b:	25dBm	-92dBm @ 1Mbps
	25dBm	-87dBm @ 11Mbps
IEEE 802.11g:	25dBm	-90dBm @ 6Mbps
	20dBm	-70dBm @ 54Mbps

某廠家網卡

協議	輸出功率	接收靈敏度
IEEE 802.11b:		-96dBm @ 1Mbps
		-90dBm @ 11Mbps
IEEE 802.11g:		-93dBm @ 6Mbps
		-74dBm @ 54Mbps

從上面的表我們可以對比下，不同網卡的接收靈敏度，在 802.11b 協定下-92dBm，R52H 速率為 1Mbps，而另外一個廠商的網卡的接收靈敏度在 1Mbps 是-96dBm，對比下 11g 協議也可以看出差別

在發射功率上，我們不僅要看發射功率，還要看看接收靈敏度，不同的 WiFi 都會寫明這些參數，RouterOS 只要不是集成無線網卡的 RB 設備，都可以更換其他廠商的 miniPCI 無線網卡，給予靈活的發射功率選擇，這點是 RouterOS 的優勢，但選擇無線網卡就是一個費心思的工作！

802.11 協議優化改進

我們知道標準的 802.11 協議訪問採用的是（CSMA/CA）方式，即利用它檢測和避免當兩個或兩個以上的網路設備需要進行資料傳送時網路上的衝突。這個方式並非像以太網一樣，有線以太網我們是可以看到各有 2 條線（1、2、3、6）進行接受和發送資料的，而 WiFi 則並不是，他採用的同一頻率向使用者發射資料，通過

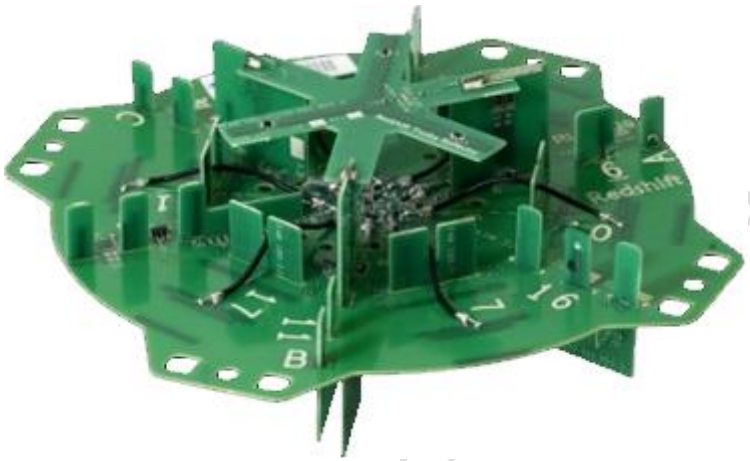
時分間隔利用同一頻率來接收使用者回傳資料，所以為什麼有 CSMA/CA 利用 ACK 信號來避免衝突的發生，也就是說，只有當用戶端收到網路上返回的 ACK 信號後才確認送出的資料已經正確到達目的地址。

當使用者多後，在同一頻率下為避免頻率佔用時間和採用載波監聽衝突檢查造成的效率下降，採用了時分多址技術，我們可以在 RouterOS 的 Nstreme 選項裡找到 disable CSMA 和 Enable Polling 的選項，即禁用 CSMA，啟用 Polling，MikroTik 早就採用輪詢權杖的方式來解決衝突的問題，用來提高多點訪問的問題，但這個技術僅適用於設備與設備之間，後來的 Nv2 協議，引入 TDMA 技術也是如此。

智慧天線

一部分有實力的廠商開始引入智慧天線，這種方式是採用多個全向天線，組成天線陣列，如果你是一個軍事發燒友，應該知道相控陣雷達，原理就和相控陣雷達一樣，我採用多個天線發射和接收使用者信號，相控陣雷達就像昆蟲的複眼一樣，多個天線模組組成，在同一平面掃描，實現同時跟蹤多個目標

當多使用者接入無線網路後，我們對使用者的信號和訪問進行處理，得到他在各個天線上的信號情況，當其中一個天線信號品質相對於其他天線要好，設備就將該使用者的資料連接轉移到指定信號好的天線，達到無線信號的優化，其實這樣說也就是把文章做在了天線上，但其實也是通過軟體和硬體結合實現的



不過智慧天線大多採用的是全向天線，主要是接收來自各個方向的 WiFi 信號，包括各種反射回來的多路徑信號，將這些信號收集處理，給 AP 設備優化使用者信號，這種反射多路徑信號在室內和建築結構複雜的環境有很大的優勢，能提高 AP 的覆蓋範圍，但在室外空曠地帶，他就無法和定向天線的 AP 比較，因為全向天線方向性差，制約了覆蓋距離，室外空曠地帶沒有那麼多反射點，智慧天線也不能發揮他的優勢，所以要看情況而定，智慧天線的優勢更多集中在室內

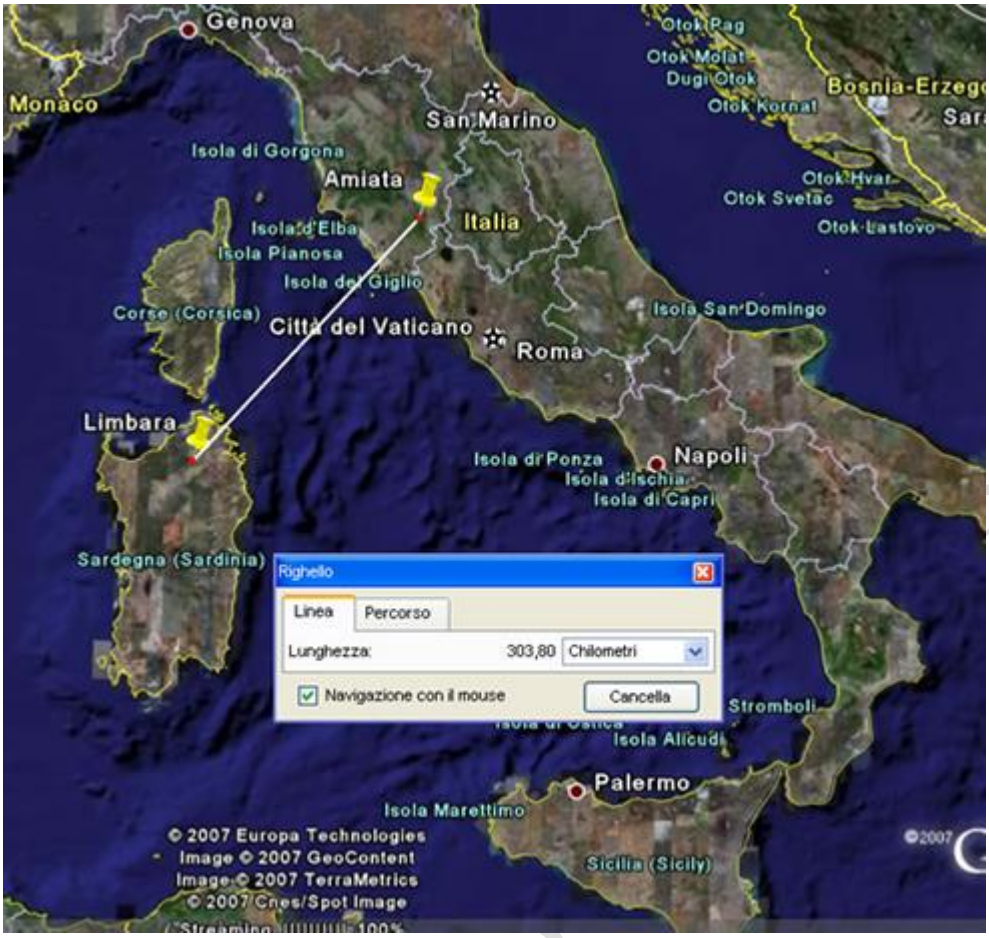
RouterOS 當然無法實現智慧天線的技術，因為他一套模組組合的系統，非一套完整的 AP 設備解決方案，就算有也沒有天線技術的支援。

3.9 WLAN 無線資料傳輸

WLAN 無線傳輸，和覆蓋不一樣，最簡單是點對點的無線傳輸，也許對 RouterOS 無線瞭解的人應該知道下面的資訊：

在 2007 年 6 月 16 日，在義大利已經實現一個 304 公里的無線連接（100 公里陸地和 200 公里海平面），在 Amiata mount (1734asl)與 Limbara mount (1300 ca asl) 之間。

採用一對基於 RouterOS 配置的無線主機板，一對 XR5 802.11a 600mW 無線網卡，一對自製 120cm 圓盤式高增益天線，使用水準極化，兩端的信號強度從-58 dBm 到-62dBm，速率在 12 到 48Mbps



300 公里的資料傳輸，的確很牛！當然 RouterOS 運用了自己的私有協議 Nstreme，優化 ACK 時間、壓縮幀資料、多點訪問時採用輪詢技術等，新的 Nstreme Version 2（Nv2），之前提到引入了 TDMA 協定，實現更多的設備接入，優化 11n 的傳輸，而且降低了系統資源消耗。如果用 Nv2 協議 11n 的頻寬可以達到近 200Mbps 的 TCP 資料傳輸，這也是 RouterOS 最大的優勢。

為什麼 RouterOS 在 11n 的產品上前期都集中在 5G 的 802.11an 上，也是他主攻傳輸市場的原因，如果做 WiFi 覆蓋的確沒有太多技術優勢,但也並非否定 RouterOS 的 WiFi 覆蓋，如果加上大功率，高接收靈敏度的無線網卡，同樣也能得到較好的覆蓋效果，只是與智慧天線設備在室內覆蓋和複雜結構環境下，就要差點了。至於周圍環境干擾嚴重的情況下，是任何無線設備都不能回避的問題。

第四章 RouterOS 無線功能介紹

要求功能包： wireless

RouterOS 無線協定遵循 IEEE 802.11 標準，並完全支持 802.11a、802.11b、802.11g 和 802.11n，並增加了如 WPA、WEP 和 AES 加密，Wireless Distribution System (WDS)，Dynamic Frequency selection（DFS 動態頻率選擇），Virtual Access Point（虛擬 AP），以及 MikroTik 的 Nstreme 和 NV2 私有協議等等。

無線能工作在多個模式下：station（client），access point（AP）和 wireless bridge 等，station 模式也可以分為多個模式，關於完整的介紹，可以參考 station 模式介紹

4.1 RouterOS 無線介紹

MikroTik RouterOS 當前所支援的協定：

- **2.4ghz-b** - IEEE 802.11b
- **2.4ghz-b/g** - IEEE 802.11b 與 IEEE 802.11g
- **2.4ghz-g-turbo** - IEEE 802.11g 支持 108 Mbit
- **2.4ghz-only-g** - IEEE 802.11g 支持 54 Mbit
- **5ghz** - IEEE 802.11a 支持 54 Mbit
- **5ghz-turbo** - IEEE 802.11a 支持 108 Mbit
- **2ghz-b/g/n** - IEEE802.11bgn (基於 4.0beta3 以上版本)分別相容 2.4GHz 頻段的 11Mbit、54Mbit 和 150Mbit~450Mbit
- **2ghz-onlyn** - IEEE802.11n (基於 4.0beta3 以上版本)僅支援 2.4GHz 頻段的 150Mbit~450Mbit
- **5ghz-a/n** - IEEE802.11an (基於 4.0beta3 以上版本)分別相容 5GHz 頻段的 54Mbit 和 150Mbit~450Mbit
- **5ghz-onlyn** - IEEE802.11n (基於 4.0beta3 以上版本)僅支援 5GHz 頻段的 150Mbit~450Mbit

MikroTik 提供了強大的無線設置功能，同樣產品包括針對 Wlan 開發的 RouterBOARD 硬體，型號從 RB100、RB500、RB400、RB600、RB700、RB800、RB900 和一體無線設備等（具體型號和參數可以登錄官網 www.routerboard.com）能應用在 802.11abgn/ac 的無線點對點、點對多點、漫遊和覆蓋等方案中。

4.2 RouterOS 支援的 WLAN 連接方式

MikroTik RouterOS 提供了多無線連接方式：**點對點連接、點對多點連、無線接力、無線漫遊等、獨有的 bonding、Nstreme、Nstreme2 協定、Nv2 協定和 Mesh 網狀網路**。

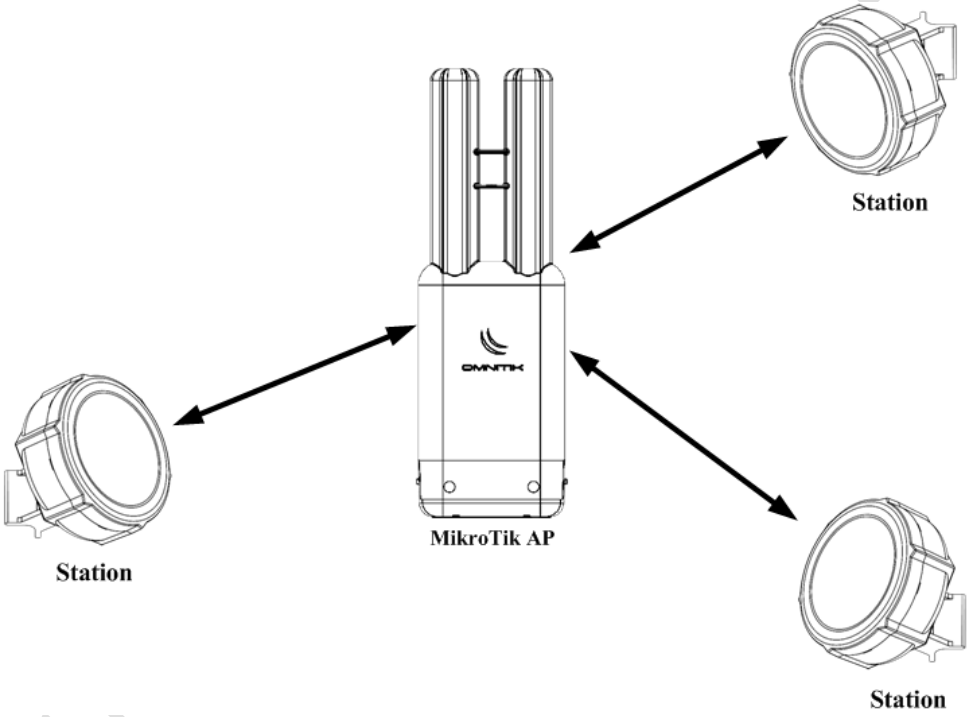
點對點連接

當兩個網路之間可採用點對點的無線連接方式。只需在每個網段中都安裝一個 AP，通過點對點傳輸實現網路信號的傳輸和互聯。在點對點連接方式中，天線最好全部採用定向天線，已得到更好的信號和頻寬。

點對點方式有兩種一中是我們常見的的橋接模式，我們可以採用 ap-bridge 或者 birdge 方式，在這裡我們推薦使用 ap-bridge 與 station-wds 的橋接方式。

點對多點連接

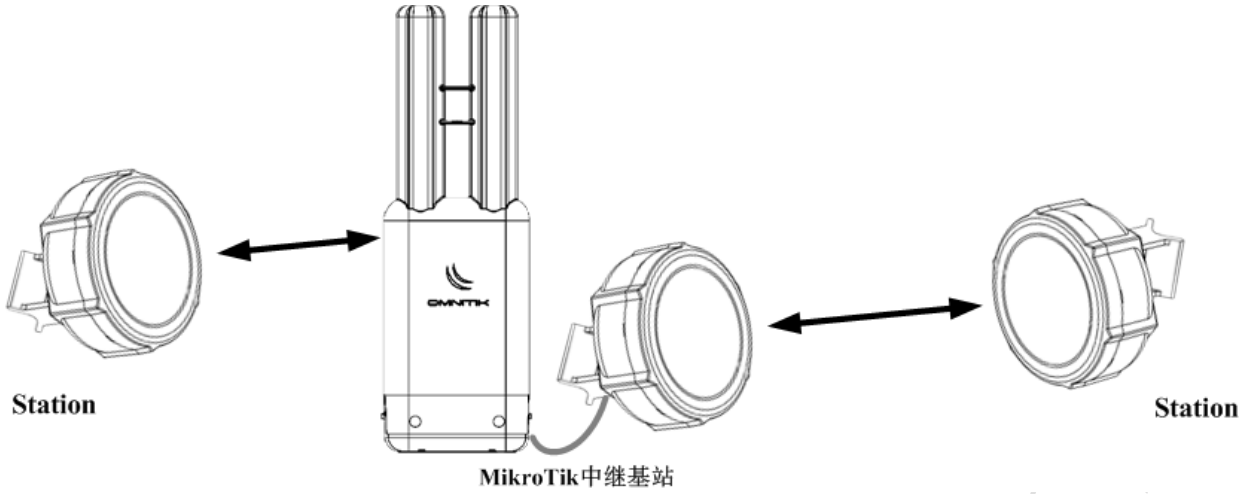
當三個或三個以上的網路之間採用光纖或雙絞線等有線方式難以連接時，可採用點對多點的無線連接方式。只需在每個網段中都安裝一個 AP，即可實現網段之間點到點連接，也可以實現有線主幹的擴展。如下圖



在點對多點連接方式中，一個 AP 設置為中心的 ap-bridge，其他接收機站則全部設置為 station 或 station-wds。在點對多點連接方式中，中心點一般採用全向天線或者扇形面的天線，用戶端則最好採用定向天線。

無線中繼

當兩個網路間的距離已經超過無線網路產品所允許的最大傳輸距離時，或者雖然兩個網路間的距離並不遙遠，在兩個網路之間有建築或其它物體阻擋，可以尋找一個中繼點實現傳輸信號的接力，如下圖



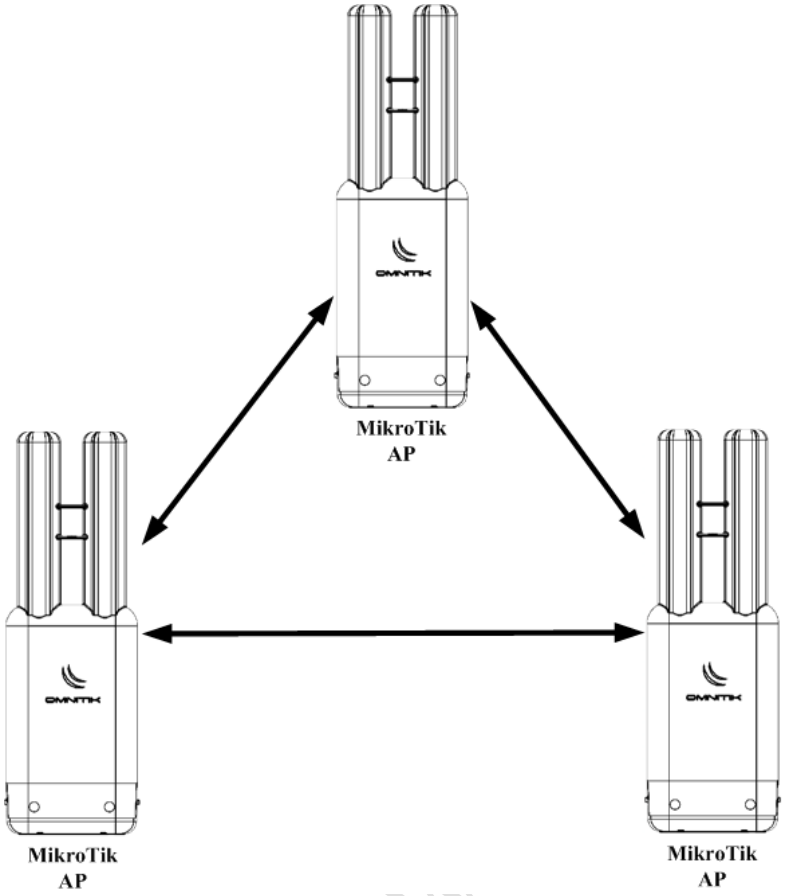
無線接力工作方式與點對多點非常類是，但無線接力是指在一個設備上添加兩張或兩張以上的網卡，做中繼傳輸，他和點對點或點對多點不同的地方在於，一個 AP 上有多個 wlan 網卡介面，需要將它們做成橋接或者路由方式，多無線網路有助於提高資料的轉發量。

無線漫遊（WDS）

多個中心基站設備可以為在網路範圍內各個位置之間漫遊的移動式無線客戶機工作站設備服務。 多基站配置中的漫遊無線工作站具有以下功能：

- 在需要時自動在基站設備之間切換，從而保持與網路的無線連接。
- 只要在網路中的基站設備的無線範圍內，就可以與基礎架構進行通信。

在城市某區域或者在網路跨度很大的大型企業中，人們可能需要完全的移動上網需求，此時，可以在網路中設置多個 AP，使裝備有無線網卡的移動終端實現如手機般的漫遊功能（如圖 4）。使用無線漫遊方案，隨時隨地訪問他們所需要的網路資源。

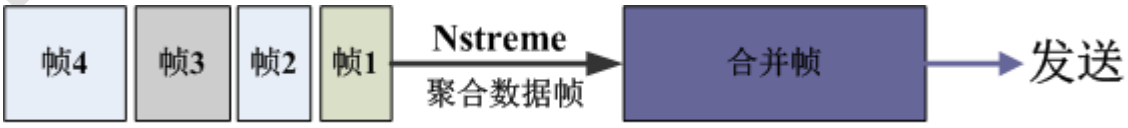


這就是所謂的無縫漫遊，在移動的同時保持連接。原因很簡單，AP 除具有橋接器功能外，還具有傳遞功能。這種傳遞功能可以將移動的工作站從一個 AP“傳遞”給下一個 AP，以保證在移動工作站和有線主幹之間總能保持穩定的連接，從而實現漫遊功能。

Nstreme 與 Nstreme v2

這功能屬於 MikroTik 專有的無線協議，在長距離和頻寬上有非常突出的表現， 隨著無線協定的發展從早期支援 802.11abg 的 Nstreme 協定開始，到現在的 Nstreme v2 全面支援 802.11n 協議，同時 Nstreme v2 引入了 TDMA 技術，能更好的支援點對多點的連結也相容 802.11abg 協議。

Nstreme 協議主要特點是通過將多個幀重組後，將多個幀重新組合成一個巨幀一次發送，目的是減少無線傳輸時過多的 ACK 請求，一些小幀過多的消耗無線連結的資源，通過 Nstreme 協議合併後，減少了 ACK 重複發送的請求，特別是在信號較差、距離較遠的情況下。



早期的 Nstreme 協定支援三種模式：

- Point-to-Point mode – 點對點模式通過在每一個點架設一個無線設備實現
- Dual radio Point-to-Point mode (nstreme2) – 這個協議是通過在每一個點架設兩套無線設備，同時分別一對做接收和一對做發送，實現雙向的通信。能做到高速連接。

- Point-to-Multipoint – 點對多點模式的客戶論旬（類似 AP 控制的權杖環）

Nstreme 協議是專用於長距離無線傳輸，正常的無線連接在長距離傳輸時，會產生高傳輸延遲。使用 Nstreme 協定後這個問題被消除。 當我們要求高速資料傳輸的時候，需要保證足夠的上下行頻寬時，我們可以選擇 Nstreme Dual 的模式，一對網卡做上行，一對做下行資料傳輸。

Nstreme 協議更多應用在 802.11abg 上，隨著 802.11n 的出現新的 Nstreme v2 協議提高了傳輸頻寬和多點的性能。

Nstreme Version 2 (Nv2)

Nv2 是 MikroTik 為 802.11n 優化的私有無線協議，基於 TDMA 技術（Time Division Mulitple Access 時分多址），Nv2 基於 TDMA 好處在具有更大的輸送量、低延遲、適用於點對多點網路連接。

TDMA 是一個頻段訪問共用網路，允許多個使用者在同頻率下通過在不同時間段信號間隔訪問方式，在屬於他自己的時間間隔內，每一次使用者傳輸一連串的資料。這個允許多個網站共用相同的傳輸介質，在一段時間內使用一部分的頻率通道

工作在 Atheros 晶片上，AR5212 和更新的 AR5414，802.11n 系列晶片包括 R52n、R52Hn、AR9220 和 AR9300 系列晶片等，從 RouterOS v5.0beta5,你可以在 wireless 功能表下配置 Nv2，Nv2 協議限制了 511 個用戶端。

Nv2 最重要的好處：

- 增加傳輸速率，特別是在 802.11n 模式下
- 更多的客戶連接到點對多點網路（PTM）
- 更低的延遲
- 沒有傳輸距離限制

從 RouterOS v5.0beta5,你可以在 wireless 功能表下配置 Nv2，Nv2 協議限制了 511 個用戶端

下面是 Nstreme 協定支援在不同協定下與其他標準 WLAN 無線產品的速率比較情況：

標準	工作頻段	頻率佔用空間	理論最大速率	標準 WLAN	Nstreme
802.11b	2.4GHz： 2312-2599MHz	5MHz	11Mbps	5.5Mbps	7Mbps
802.11g G-Turbo	2.4GHz 2312-2599MHz	5MHz 44MHz	54Mbps 108Mbps	27Mbps 54Mbps	37Mbps 68Mbps
802.11a A-Turbo	5GHz： 4920-6100MHz	5MHz 10MHz 20MHz 40MHz	13.5Mbps 27Mbps 54Mbps 108Mbps	6.75Mbps 13.5Mbps 27Mbps 54Mbps	9Mbps 18Mbps 37Mbps 74Mbps
802.11n	2.4/5Ghz： 2312-2599Mhz 4920-6100Mhz	5MHz 10MHz	37.5Mbps 75Mbps	18.5Mbps 37.5Mbps	28Mbps ‘ 50Mbps

		20MHz 40MHz(2×20MHz) 60MHz(3×20MHz)	150Mbps 300Mbps 450Mbps	75Mbps 150Mbps ？？	100Mbps 200Mbps ？？

Mesh 無線網狀網路

MikroTik 支援的 STP 和 HWMP+兩種方式的 Mesh：

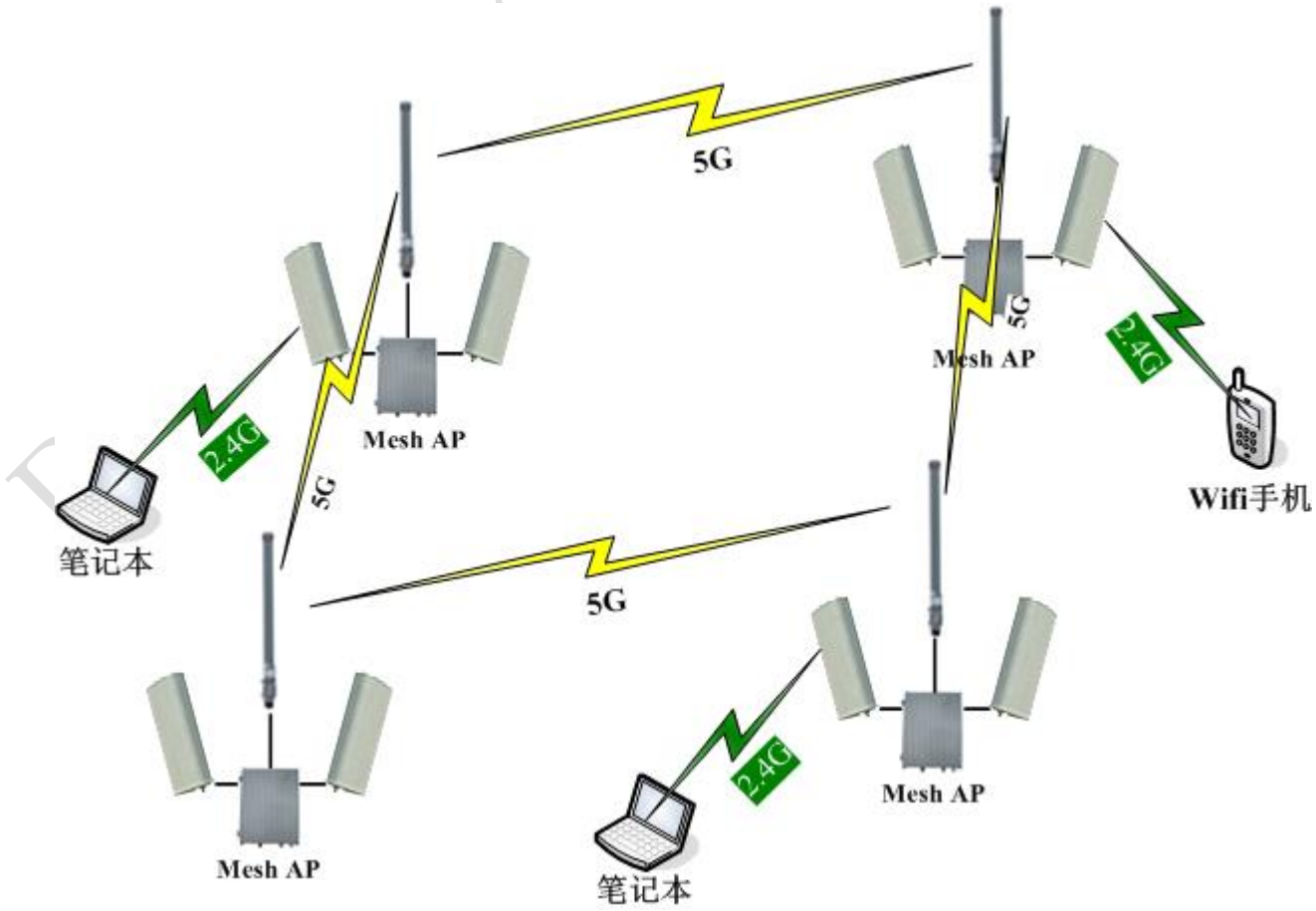
STP 生成樹協議的英文縮寫。該協定可應用於環路網路，通過一定的演算法實現路徑冗余，從而避免報文在環路網路中的增生和無限迴圈。在 v3.0 後支援快速生成樹協定 RSTP，在收斂速度上更快，通過優先順序劃分線路的優先路徑。

HWMP+是 MikroTik 為無線網狀網路 Mesh 定義的 2 層路由式通訊協定。基於 IEEE802.11s 草案 Hybrid Wireless Mesh Protocol (HWMP)，能用於替代 STP 生成樹協議確保環路的最優路徑。這種分散式系統不僅能應用到無線分佈系統（WDS）。HWMP+網狀網路同樣也支援以太網介面的網狀網路，因此你可以用於簡單的以太網分佈系統，或者同時連接 WDS 和以太網。構建一個大型的無線網路，如城市 Wlan 網路。

STP 和 HWMP+協定的 Mesh 在 MikroTik 特性：

每台 AP 的無線模組發射模式都配置為 ap-bridge，根據需要配置頻率和 SSID。骨幹與覆蓋 SSID 不同，但骨幹與骨幹之間 SSID 相同，覆蓋也一樣。

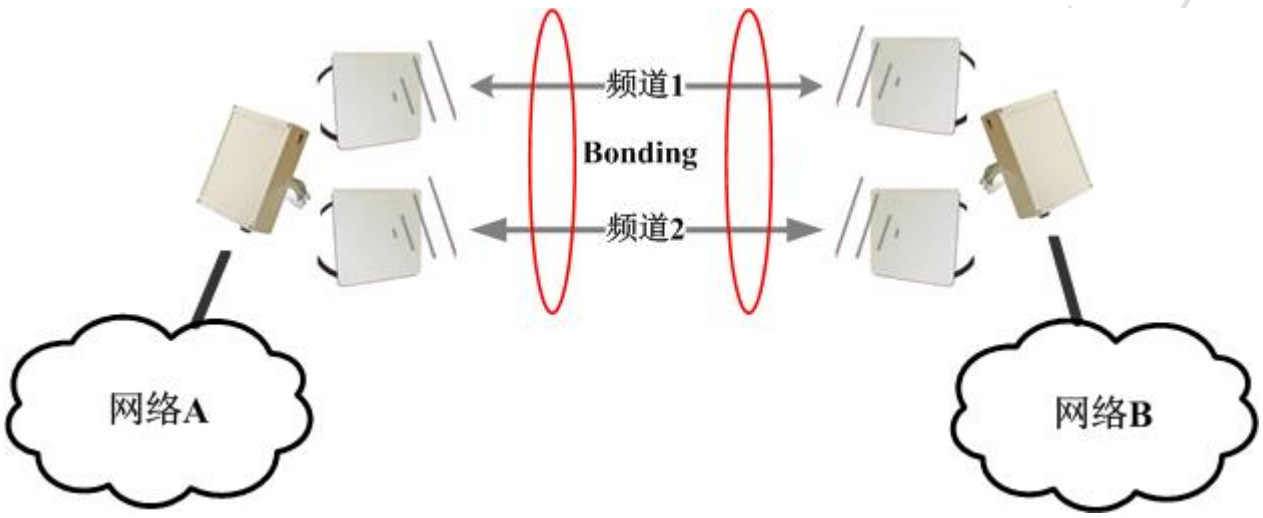
WDS 模式：v2.9 選擇 dynamic，v3.0 選擇 dynamic-mesh（v3.0 的 dynamic-mesh 效率要比 v2.9 的高）配置 Bridge 或者 Mesh 參數，並將指定的介面定義入 Bridge 或 Mesh 中，但 Bridge 和 Mesh 不能被同時使用。



MikroTik bonding 功能

Bonding 是通過彙聚多個介面到一個虛擬的連結上，這種方式可以獲得更高的頻寬或提供失效轉移接管。 Bongding 操作必須用於二層鏈路層，不支援三層 IP 層的應用。（關於 RouterOS bonding 介紹請參考《RouterOS 中文網路教程》）

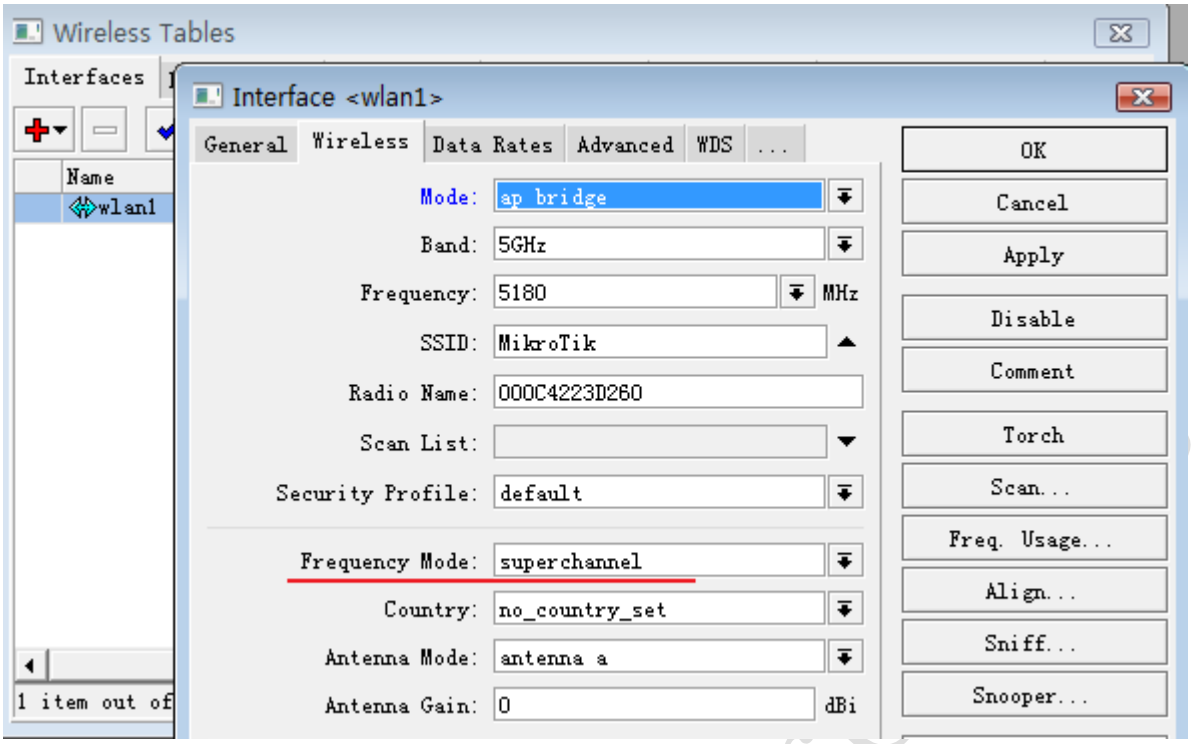
通過 Bonding 功能，可以將兩條或者兩條以上的無線鏈路綁定在一起，起到將無線網路合併頻寬的作用，如下圖，是為了提高網路 A 和網路 B 直接的頻寬，將 2 條無線綁定在一起，也可用起到線路備份的作用：



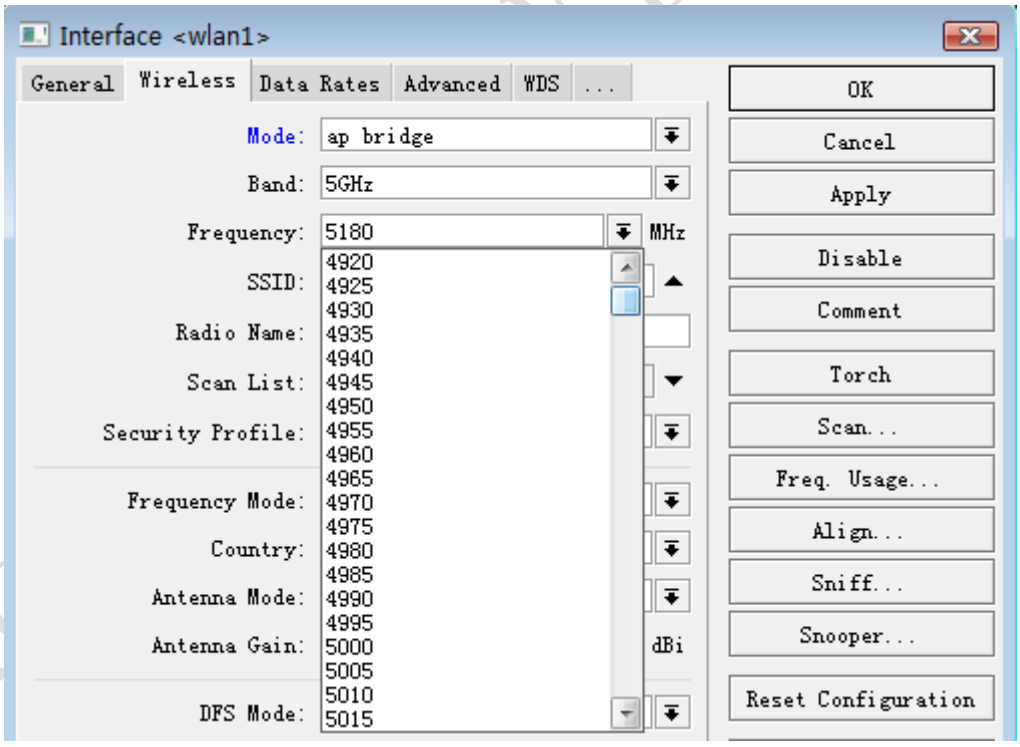
MikroTik Superchannel

MikoTik 的 Superchannel 模式是，增加了無線的發射頻率，能夠獲取更多的無線通道，避免 WLAN 因為較少的通道而帶來的干擾問題，RouterOS 支持的 2G 和 5G 頻率範圍：

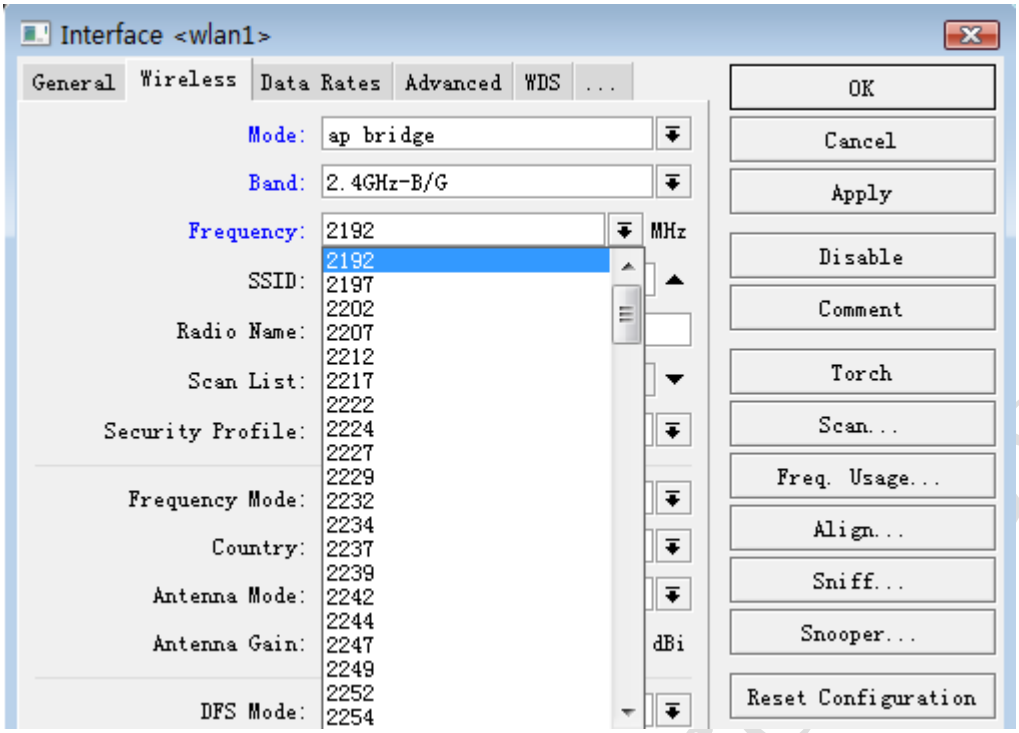
我們可以進入 wireless 目錄下選擇 wlan1 後，點擊 Advance Mode，看到 Frequency-Mode 選項，並選擇為 superchannel，這樣我們便可以將超級頻道啟動（升級超級頻道需要另外註冊 key）如下圖：



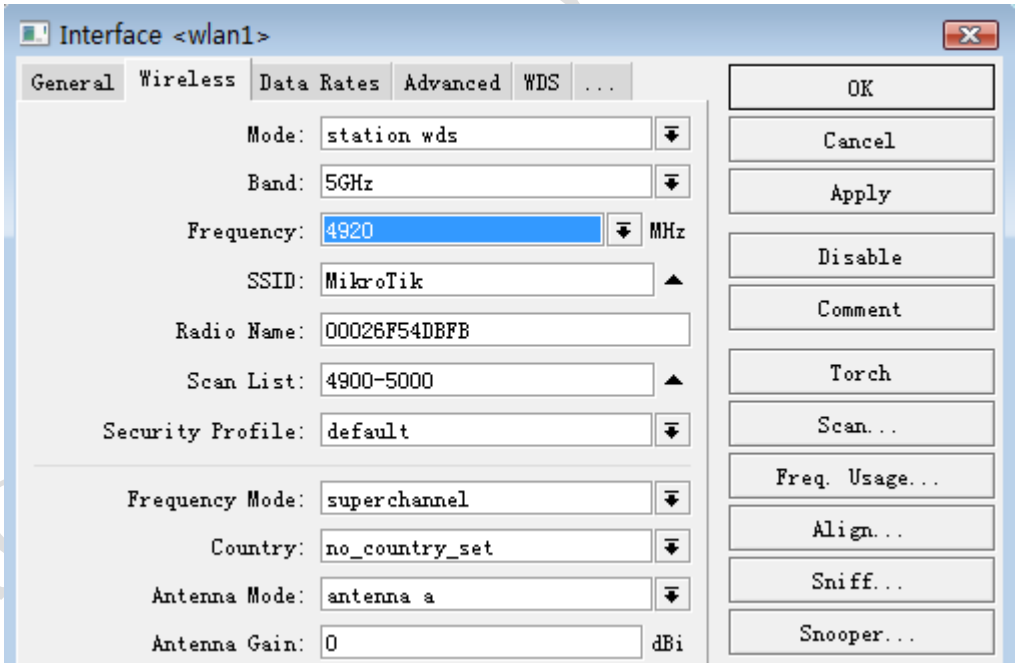
啟動超級頻道後，我們可以看到 5GHz 頻道下可以從 4920MHz~6100MHz，每 5MHz 一跳，如下圖



2.4G 頻段，是從 2192MHz~2539MHz，也是每 5MHz 一跳，如下圖：



注意：RouterOS 設置 superchannel 時，當 AP 端設備使用 ap-bridge 設置為超級頻道後，在 station 或者 station-wds 搜索頻道時，只會按照預設頻率搜索，搜索時間會很長，可能會花費 1～2 分鐘時間，為了提高連接效率，我們需要在 scan-list 設置搜索頻率，如下圖，設置 4900-5000MHz 的頻率搜索範圍



4.3 RouterOS 802.11 協議

802.11 二層橋接限制

802.11 數據框架格式

首先要說明的是 802.11 的框架格式很特別，它的長度是可變的。不同功能的資料幀長度會不一樣。這一特性說明 802.11 資料幀顯得更加靈活，然而，也會更加複雜。802.11 的資料幀長度不定主要是由於以下幾點決定的

mac 位址數目不定，根據框架類型不同，mac 802.11 的 mac 位址數會不一樣。比如說 ACK 幀僅有一個 mac 位址，而資料幀有 3 個 mac 位址，在 WDS 模式（下面要提到）下，幀頭竟然有 4 個 mac 地址。我們看看 802.3 的幀結構

802.3 MAC 框架格式。

Preamble	SFD	Dst	Src	Length	Data	FCS

幀內容：

- Preamble（前導序列）：由 62bit 交替出現的 0,1 序列組成。設置目的,接收端實體層同步位元時鐘。
- SFD（起始域）：“11”表示有用資料開始。
- Dst(目的地址域):由 6 位元組組成,表目的節點位址。
- Src(源地址域):由 6 位元組組成,表源節點位址。
- Length(長度域):由 2 位元組組成。,資料欄長度。
- Data(資料欄):46 位元組~1500 位元組之間。
- FCS(校驗域):4 位元組組成。

802.11 幀結構

Frame Control	Duration	RA	TA	DA	Seq	SA	Data	FCS

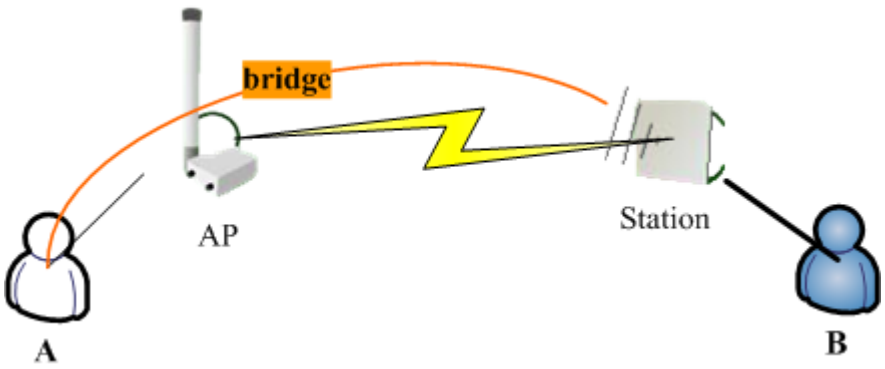
我們可能會碰到以下類型的 mac 地址

- RA(receiver address)：無線網路中，該資料幀的接收者
- TA(transmitter address)：無線網路中，該資料幀的發送者
- DA(destine address)：該幀的目的 mac 位址
- SA(source address)：該幀的源 mac 地址

這裡的 DA 和 SA 含義和普通以太網中的 Dst 和 Src 位址含義一樣，在無線網路中可能我們需要通過 AP 把資料發送到其它網路內的某台主機中。但是有的人會奇怪，直接在 RA 中填這台主機的 mac 位址不就久好了麼。但是請注意 RA 的含義，說的是無線網路中的接收者，不是網路中的接收者，也就是說這台目的主機不再無線網路範圍內。在這種情況下我們的 RA 只是一個中轉，所以需要多出一個 DA 欄位來指明該幀的最終目的地，當然，如果有了 DA 那必須有 SA，因為若目的主機要回應的話，SA 欄位是必不可少的。(假設沒有 SA 欄位，那麼目的主機回應的資料包就只能發送到源主機所屬的 AP 上了)

我們要知道 Access Point 是一個訪問節點，即 AP 是一個用於終端設備網路連接的節點，終端設備包含了各種 PC、筆記本和各種設備，我們可以稱他們為 station，但後來我們需要通過 802.11 協定實現橋接器傳輸，但是最初 802.11 協議的 AP 被期望能通過無線橋接二層的幀，但 station 設備並沒有考慮二層的橋接

我們考慮一下網路：



這裡 A 到 AP 與 station 到 B 都是通過以太網連接，但是 AP 到 Station 是通過無線。根據 802.11，AP 能透傳二層橋接在 A 和 Station 之間，但不能實現橋接傳輸在 AP 和 B 之間，或者 A 和 B 之間。

802.11 協定標準指定幀在 station 和 AP 設備之間 只能傳輸幀的頭部包含 3 個 MAC 位址

幀傳輸從 AP 到 station 包含以下 MAC 位址

- TA(無線發送者地址) - AP 的地址
- DA(目標位址)- station 設備的位址，也是無線接收者的位址
- SA(源地址) - 發送者的源 MAC 地址

幀傳輸從 station 到 AP 包含以下 MAC 位址：

- RA(無線接收者地址)- AP 的 MAC 地址
- DA(目標位址) - 達到的目標位址
- SA(源位址) - station 設備的位址，即無線傳輸者 MAC 位址

這樣每個幀都包含無線發送者和接收者的 MAC 位址，包含 3 個 mac 位址的框架格式是不能適應透明的二層橋接，因為 station 不能發送與自己不同的源 MAC 位址，例如從 B 來的幀，並且同時 AP 也不能在幀裡包含 B 的位址

在 WDS 模式下，一個 AP 互相連接的系統資料幀會有 4 個位址，RA，TA 表示接收端和發送端，這兩個位址用於無線傳輸的時候。還有 2 個位址是 DA 和 SA，分別跟以太網中一樣表示源位址和目的地址

- RA(receiver address)：無線網路中，該資料幀的接收者
- TA(transmitter address)：無線網路中，該資料幀的發送者
- DA(destine address)：該幀的目的 mac 位址
- SA(source address)：該幀的源 mac 地址

這樣的幀在無線連接裡包含了必要的二層橋接器資訊，但不幸的是 802.11 協議沒有指明 WDS 如何建立和維護無線網路的連接，因此任何使用 4 個 MAC 位址的幀需要進行明確的執行。

4.4 基本無線速率和 MCS 速率

基本速率是無線 AP 設備要求終端必須滿足的速率條件，在滿足這個條件下才能連接到 AP，下面是關於 802.11n 的 MCS 對應速率表，MCS 0～7 使用單條空間流，即 MIMO 為 1×1，當 MCS＝7 時，速率值最大。MCS 8～15 使用兩條空間流，即 MIMO 為 2×2，當 MCS＝15 時，速率值最大。

MCS 對應速率表（頻寬為 20MHz）

MCS 索引	空間流數量	調製方式	速率(Mb/s)	
			800ns GI	400ns GI
0	1	BPSK	6.5	7.2
1	1	QPSK	13.0	14.4
2	1	QPSK	19.5	21.7
3	1	16-QAM	26.0	28.9
4	1	16-QAM	39.0	43.3
5	1	64-QAM	52.0	57.8
6	1	64-QAM	58.5	65.0
7	1	64-QAM	65.0	72.2
8	2	BPSK	13.0	14.4
9	2	QPSK	26.0	28.9
10	2	QPSK	39.0	43.3
11	2	16-QAM	52.0	57.8
12	2	16-QAM	78.0	86.7
13	2	64-QAM	104.0	115.6
14	2	64-QAM	117.0	130.0
15	2	64-QAM	130.0	144.4

MCS 對應速率表（頻寬為 40MHz）

MCS 索引	空間流數量	調製方式	速率(Mb/s)	
			800ns GI	400ns GI
0	1	BPSK	13.5	15.0
1	1	QPSK	27.0	30.0
2	1	QPSK	40.5	45.0
3	1	16-QAM	54.0	60.0
4	1	16-QAM	81.0	90.0
5	1	64-QAM	108.0	120.0
6	1	64-QAM	121.5	135.0
7	1	64-QAM	135.0	150.0
8	2	BPSK	27.0	30.0

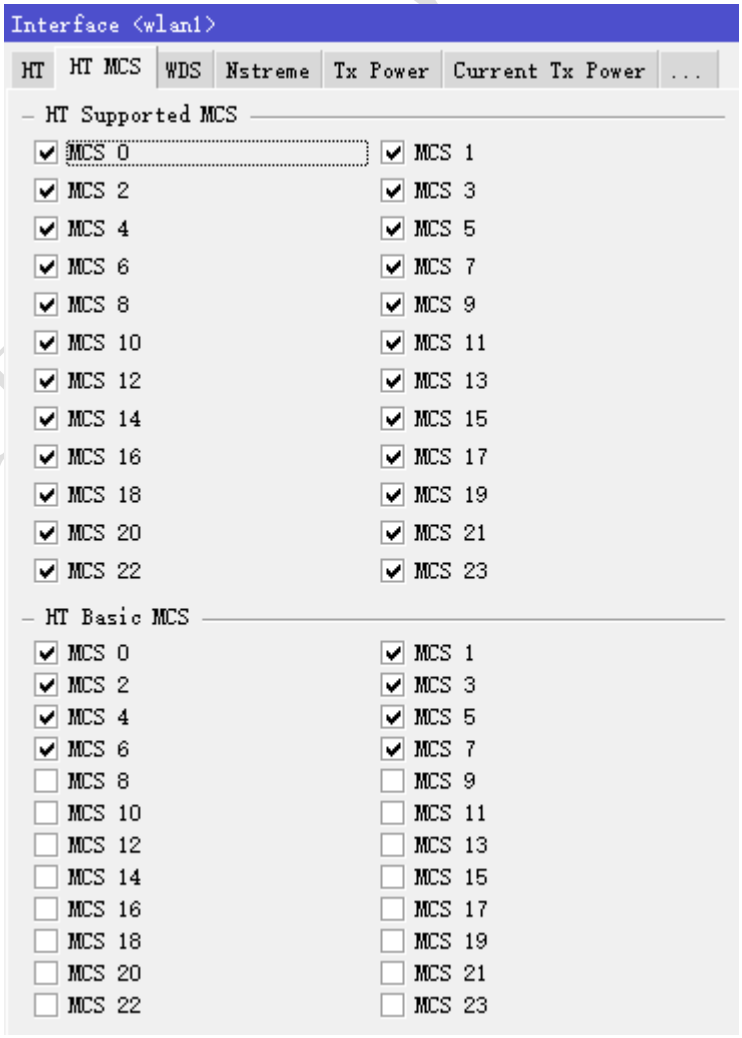
MCS 索引	空間流數量	調製方式	速率(Mb/s)	
			800ns GI	400ns GI
9	2	QPSK	54.0	60.0
10	2	QPSK	81.0	90.0
11	2	16-QAM	108.0	120.0
12	2	16-QAM	162.0	180.0
13	2	64-QAM	216.0	240.0
14	2	64-QAM	243.0	270.0
15	2	64-QAM	270.0	300.0

在 RouterOS 裡 MCS 速率配置分為兩類：**Basic-MCS** 和 **Support-MCS**：

Basic -MCS：基本 MCS 是指 AP 正常工作所必須支援的 MCS 速率集，終端設備必須滿足 AP 所配置的基本 MCS 速率才能夠與 AP 進行關聯。

Support-MCS：支援 MCS 速率集是在滿足 AP 的 basic-MCS 集的條件下，AP 所能夠支援的更高的速率，你可以配置 support-MCS 速率選擇更高的速率與 AP 進行關聯。

在 RouterOS 可以看到如下配置選項：



當 rate-set=configured 時，即無線的速率手動調節，下面是每種協定可以選擇的速率對應關係

802.11 協議	可選設置
2.4ghz-b	basic-b, supported-b
2.4ghz-b/g, 2.4ghz-onlyg	basic-b, supported-b, basic-a/g, supported-a/g
2.4ghz-onlyn 2.4ghz-b/g/n	basic-b, supported-b, basic-a/g, supported-a/g, ht-basic-mcs, ht-supported-mcs
2.4ghz-g/n	basic-a/g,supported-a/g,ht-basic-mcs,ht-supported-mcs
5ghz-a	basic-a/g,supported-a/g
5ghz-a/n, 5ghz-onlyn	basic-a/g,supported-a/g,ht-basic-mcs,ht-supported-mcs
5ghz-a/n/ac, 5ghz-onlyac	basic-a/g,supported-a/g,ht-basic-mcs,ht-supported-mcs,vht-basic-mcs,vht-supported-mcs

當你設置 rate-set=configured 時，無線速率不再由 RouterOS 自訂，而是手動調整，該操作需要謹慎，特別是在 WiFi 覆蓋場景，特別是 basic-rate 參數非常關鍵，即 AP 對終端設備基本速率的要求，如果把 basic-rate 調整過高，會導致部分設備無法連接。

4.5 RouterOS 各種 station 模式

該模式在 wds-mode 關閉下使用

這個是標準模式，在 station 模式下不支援二層橋接，如果試圖將無線網卡放入 bridge 將不能獲得預期的結果，但從另一方面考慮如果二層橋接不是必須的解決方式，可以選擇路由和 MPLS 交換，這個模式支援所有的 ROS 無線協定

station-wds

該模式在 wds-mode 開啟下使用

這個模式僅能與 RouterOS AP 工作（一些 WRT 軟體支援），為此需要通過協商連接，所以 AP 端需要為對應的 station 建立一個獨立的 WDS 介面，這個介面能通過點對點連接 AP 和對應的，無論是從 AP 到 station、還是 station 到 AP 或者終端之間的轉發，都會保留二層 MAC 位址

這個模式支援所有的 RouterOS 無線傳輸協定（非 RouterOS 設備不支援），當使用標準 802.11 協定這個模式使用 4 個位址框架格式，如果是 MikroTik 私有協議（Nstreme 或 Nv2）將採用內部方法

這個模式能安全的使用二層橋接，即 AP 創建獨立的 WDS 介面能使用橋接防火牆和 RSTP 環路探測和回避等

station-pseudobridge

這個模式從無線連接觀點看與標準 station 模式連接相同，不過有限的支援二層橋接

基於 MAC 位址轉換的 IPv4 資料傳輸，當幀發向 AP 通過 MAC 地址清單替換 IPv4 資料的源 MAC 位址（為了能使用 3 個 MAC 位址的框架格式），反過來替換發現 station 的目標 MAC 位址，也同樣能建立 VLAN 幀的封裝（但不能支持 PPPoE 透傳）

station-pseudobridge-clone

這個模式和 station-pseudobridge 模式相同，This mode is the same as station-pseudobridge mode, 除了連接到 AP 使用“clone”MAC 位址外，

station-bridge

這個模式僅能工作在基於 RouterOS 的 AP，提供對 station 設備透明二層橋接的支援，RouterOS AP 接受當啟用 bridge 模式時用戶端採用 station-bridge 模式的連接，即這個模式支援 AP 使用 bridge 模式，用戶端使用 station-bridge，這樣兩個 L3 的設備可以實現點對點的橋接器傳輸（主要應用於 Nv2 傳輸），這個模式同樣能實現安全的二層橋接，即 AP 創建獨立的 WDS 介面能使用橋接防火牆和 RSTP 環路探測和回避

Station Roaming

Station Roaming 模式是在 RouterOS v6.35 開始支援，winbox 管理操作介面是在 v6.38.3 版本添加。該模式僅支援 802.11 無線協定，且配置為 station 模式下使用，當 RouterOS 無線用戶端使用 802.11 無線協定連接到 AP ，並在指定的時鐘週期執行背景掃描。當背景掃描找到一個較好的 AP 信號，會嘗試漫遊到該 AP。掃描時間間隔在無線信號變得很差時，會縮短掃描時間，但當無線用戶端信號很好時，時間間隔會增長。

下面是配置實例

```
[admin@MikroTik] /interface wireless> set 0 mode=station-wds station-roaming=enabled
wireless-protocol=802.11
```

測試操作丟包 1 個，切換標準無法知道，以下是根據 RouterOS 的 wireless debug 日誌看到的情況，切換過程

```
16:05:07 wireless,debug wlan1: start background scan
16:05:10 wireless,debug wlan1: background scan complete, must select network
16:05:10 wireless,debug wlan1: no network that satisfies connect-list,  by default
 choose with strongest signal
16:05:10 wireless,debug wlan1: found better AP E4:8D:8C:BD:14:D1
16:05:10 wireless,info E4:8D:8C:60:B6:CD@wlan1: lost connection, roaming
16:05:10 wireless,debug wlan1: connect to better AP E4:8D:8C:BD:14:D1
16:05:10 wireless,info E4:8D:8C:BD:14:D1@wlan1 established connection on 2462000, SSID mik1
```

4.6 Repeater 中繼器

從 RouterOS 6.35 加入了 Repeater 無線中繼功能，Wireless repeater 允許無線網卡從 AP 端接收信號，並能使用相同的無線網卡複製信號給其他用戶端連接。Wireless repeater 通過配置無線網卡連接 AP 使用 station-bridge 或 station-pseudobridge 選項，並創建一個虛擬 AP 介面，然後創建一個 bridge 介面，並將兩個介面（主網卡和虛擬網卡）添加到 bridge 中。

如果你的 AP 支援 **button-enabled WPS** 模式，你可以使用自動設置命令

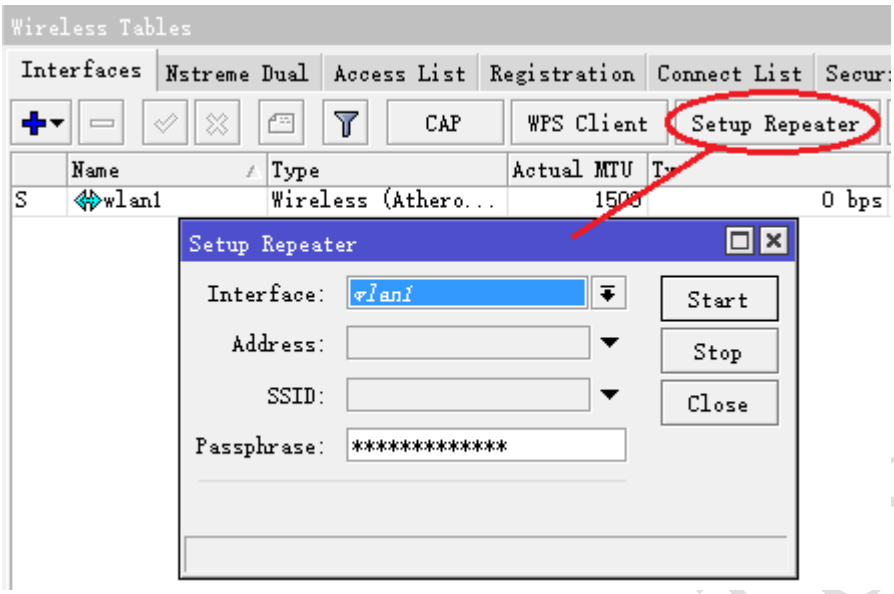
```
/interface wireless setup-repeater wlan1
```

setup-repeater 命令有以下步驟

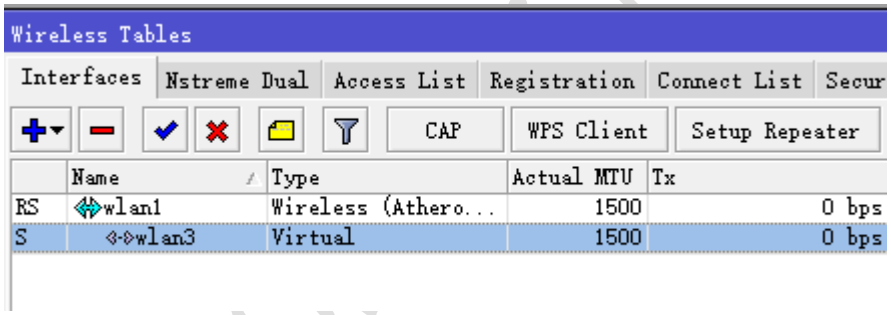
- 當按鈕被按下後，搜索 WPS AP
- 從 AP 獲取 SSID，key，頻道
- 重定主無線網卡配置
- 創建新的 bridge 介面，並刪除主無線網卡和虛擬網卡在其他 bridge 端口下的配置
- 刪除所有添加到該主無線網卡的虛擬網卡
- 創建 security profile，取規則是 "<interfacename>-<ssid>-repeater"，如果有相同 security profile 存在，將無法創建新規則，只做配置更新設置
- 配置主無線網卡，介面模式選擇方式：如果對端 AP 支援橋接模式，選擇使用 station-bridge，如果 AP 支援 WDS 模式，選擇使用 station-wds，如果 AP 之前兩種模式都不是則選擇使用 station-pseudobridge
- 創建虛擬 AP 介面，並使用相同的 SSID 和 security profile
- 如果主無線網卡沒有在其他 bridge 裡，會創建一個新的 bridge 介面，並將主無線網卡添加進去
- 將創建的虛擬 AP 介面，添加到與主無線網卡同一個 bridge 下
- 如果你的 AP 不支持 WPS，只能通過手動設置，使用以下參數
- address - AP 的 MAC 地址
- ssid - 中繼器連接 AP 的 SSID
- passphrase - AP 的驗證密碼-如果對應 AP 的密碼指定，將搜索 AP，並在/interface wireless security-profile 創建指定密碼的安全性原則。如果密碼未指定，將通過 WPS 找到密碼

setup-repeater 手動配置過程，可以通過搜索查找當前周圍的 AP，至少需要在除 interface 選項外，設置一個參數，即對端 AP 的連接密碼。

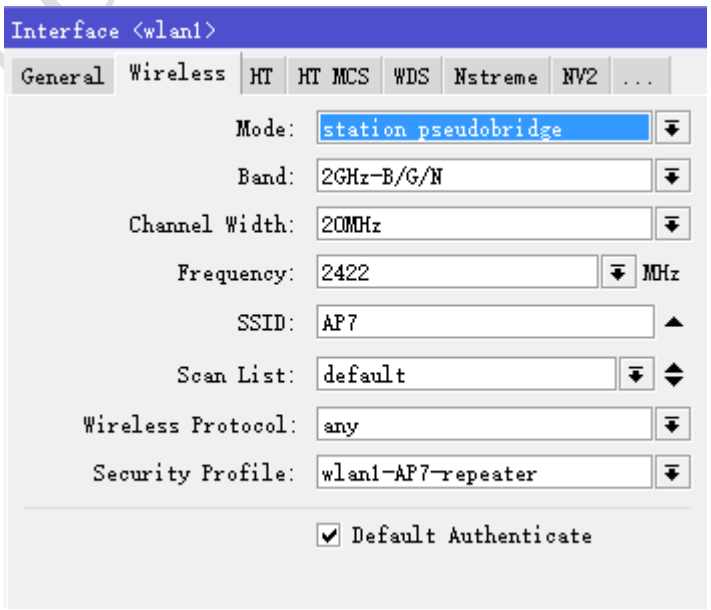
根據中繼器配置要求填寫入連接 AP 的密碼資訊填



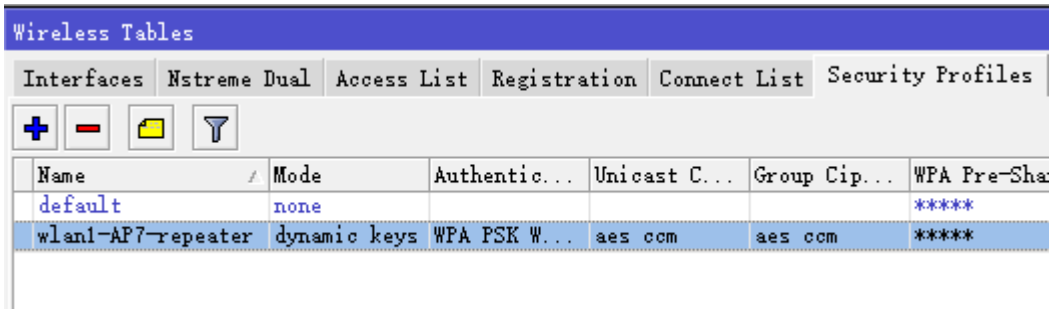
配置完成後，點擊 start，開始搜索 AP，同時 RouterOS 會重定當前無線網卡的配置，並嘗試連接對應 AP，連接成功後，會自動創建一個虛擬 AP，如下圖創建了 wlan3 的虛擬 AP，並從屬於 wlan1 物理無線網卡



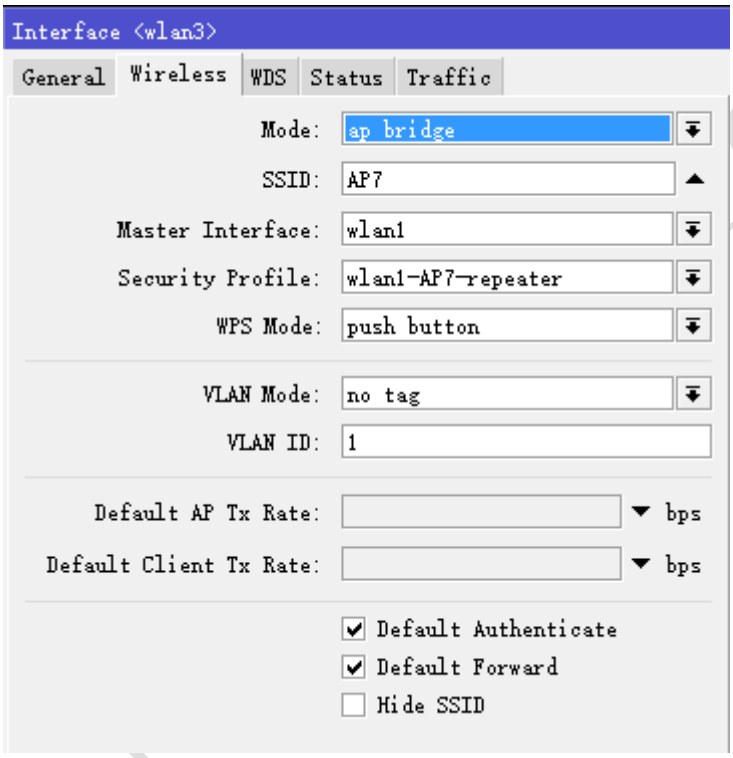
從下面物理網卡的配置可以看到，RouterOS 在重定網卡後，重新配置 mode=pseudobridge，並添加了安全性原則為 wlan1-AP7-repeater



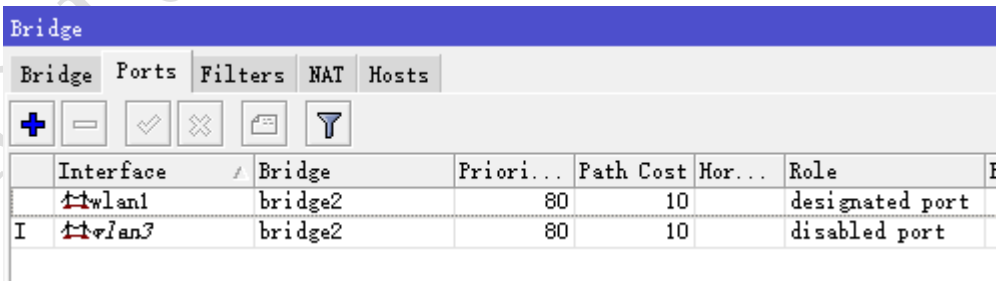
我們可以在 security-profiles 查看到新增的安全性原則



虛擬無線網卡 wlan3 配置，模式為 ap-bridge，SSID 為相同的 AP7，安全性原則為：wlan1-AP7-repeater



然後我們在 bridge 可以查看到 RouterOS 會自動為中繼無線添加兩個介面的 bridge 橋接配置，wlan1 和 wlan3 添加到 bridge2 介面下，完成橋接



4.7 RouterOS Wireless 基本參數介紹

屬性	描述
adaptive-noise-immunity (*ap-and-client-mode* / *client-mode* / *none*; 默認: none)	自我調整噪音免疫，該屬性僅基於 Atheros 廠商晶片，且是 AR5212 晶片或更高網卡晶片能生效

antenna-gain (整型 *[0..4294967295]*; 默認: 0)	天線增益，單位 dBi，被用於計算最大傳輸功率，主要根據各個國家對發生功率要求限制決定
antenna-mode (*ant-a* / *ant-b* / *rxa-txb* / *txa-rxb*; 默認: *ant-a*)	設置天線使用船速和接收方式，僅 802.11abg 可以選 *ant-a* - 僅用'a'天線介面（網卡 main） *ant-b* -僅用'b'天線介面（網卡 aux） *txa-rxb* - 使用'a'天線發射，'b'天線接收 *rxa-txb* -使用'a'天線接收，'b'天線發射
area (*字元*; 默認:)	識別無線網路工作組，該值通過 AP 宣佈，並匹配其他 AP 的 connect-list 下的 area-prefix。這個是一個專屬區域擴展
arp (*disabled* / *enabled* / *proxy-arp* / *reply-only*; 默認: enabled)	請參考 ARP 位址解析通訊協定
band (*2ghz-b* / *2ghz-b/g* / *2ghz-b/g/n* / *2ghz-onlyg* / *2ghz-onlyn* / *5ghz-a* / *5ghz-a/n* / *5ghz-onlyn*; 默認:)	定義無線頻段和頻寬速率
basic-rates-a/g (*12Mbps* / *18Mbps* / *24Mbps* / *36Mbps* / *48Mbps* / *54Mbps* / *6Mbps* / *9Mbps*; 默認: 6Mbps)	定義 a/g 頻寬速率
basic-rates-b (*11Mbps* / *1Mbps* / *2Mbps* / *5.5Mbps*; 默認: 1Mbps)	定義 b 頻寬速率
bridge-mode (*disabled* / *enabled*; 默認: enabled)	僅 AP-bridge 和 bridge 模式可選
burst-time (*整型*/ *disabled*; 默認:)	
channel-width (*10mhz* / *20/40mhz-ht-above* / *20/40mhz-ht-below* / *20mhz* / *40mhz-turbo* / *5mhz*; 默認: 20mhz)	允許 ht 使用高於和低於 20MHz 擴展頻率。擴展頻率允許 11n 設備使用 40MHz 的頻譜從而增加最大輸送量
comment (*字元*; 默認:)	介面注釋描述
compression (*yes* / *no*; 默認 no)	設置屬性為 yes，即允許硬體壓縮。無線網卡必須支援硬體壓縮功能。連接設備沒有使用壓縮仍然能工作。
country (*name of the country* / *no_country_set*; 默認: no_country_set)	各個國家可以獲得的頻段範圍、每個頻率最大發射功率。也可以自訂 scan-list（用於其他頻段或超級頻段搜索）. 值選擇為 *no_country_set* 是遵循 FCC 頻道設置。
default-ap-tx-limit (整型 *[0..4294967295]*; 默認: 0)	ap-tx-limit 值對 client 的頻寬速率起作用（僅支援 MikroTik 設備），將不會匹配 access-list 條目
default-authentication (*yes* / *no*; 默認: yes)	AP 模式可以選，該值是否驗證用戶端，如果設置為 yes，將不會匹配任何 access-list 列表的條目，設置為 no 將匹配 access-list 條目。對於 station mode 連接到 AP，該值設置為 yes 將不會匹配 connect-list 的任何條目。
default-client-tx-limit (*整型 [0..4294967295]*; 默認: 0)	Client-tx-limit 值對 client 的頻寬速率起作用（僅支援 MikroTik 設備），將不會匹配 access-list 條目

default-forwarding (*yes* / *no*; 默認: yes)	該值設置為 yes 預設轉發用戶端資料，將不會匹配 access-list 下任何條目。如果設置為 no 將匹配 access-list 條目
dfs-mode (*no-radar-detect* / *none* / *radar-detec*; 默認: none)	DFS (Dynamic Frequency Selection)，動態頻率選擇技術，用於探測網路內的頻率使用情況，動態選擇頻率 *none* - 禁用 DFS。 *no-radar-detect* - 從 scan-list 選擇指定範圍探測頻率。'wds-slave'模式該設置不能生效。 *radar-detect* - 選擇最低可用的網路頻率，並選擇在 60 秒內沒有探測到的頻率。另外，選擇不同頻率，這個設置要求國家的註冊通過 該屬性僅支援 AP 模式
disable-running-check (*yes* / *no*; 默認: no)	當設置為 yes，網卡將總是運行標記
disabled (*yes* / *no*; 默認: yes)	是否禁用無線網卡
disconnect-timeout (*time [0s..15s]*; 默認: 3s)	在幀在最低的資料率下發送已失敗，被用於探測發送 3 次失敗信號的間隔週期，在此時“3 * (hw-retries + 1)”。在整個 on-fail-retry-time 週期內 disconnect-timeout 資料包發送將被重新發。如果沒有幀在整個 diconnect-timeout 週期內沒有成功，無線連接關閉，並在 log 事件中記錄"extensive data loss"大量資料丟失。當幀成功發送該計時器會重設。
distance (*integer* / *dynamic* / *indoors*; 默認: dynamic)	連接距離，通常計算 ack 值，即多長時間等待確認單播幀。dynamic 值控制 AP 探測，並使用最小 ack 時間連接用戶端，ack 不能被用於 Nsteme 協議。
frame-lifetime (*integer [0..4294967295]*; 默認: 0)	當幀發送時間長度超過了 frame-lifetime 丟棄該幀，預設值為 0，幀被丟棄只會在連接關閉後。
frequency (*整型 [0..4294967295]*; 默認:)	頻率以 MHz 單位工作 頻率選擇依賴你選擇的 band，且與你選擇的 country（國家）和無線網卡功能參數有關 這個設置不能工作在任何 station 模式、wds-slave 模式和 DFS 啟用狀態下。
frequency-mode (*manual-txpower* / *regulatory-domain* / *superchannel*; 默認: manual-txpower)	這裡可以獲得三種頻率模式： *regulatory-domain*- 限制可獲得的頻率和功率，都根據你選擇各個 country 國家的要求 *manual-txpower* - 如同上，但沒有限制最大功率 *superchannel* - 性能測試模式（Conformance Testing Mode）. 允許支援無線網卡支援的所有頻率。 列出所有頻段可以獲得的頻率通過命令/wireless info print。superchannel 這個模式應該被用在有限範圍，或者你有特殊的許可在特定的地區。在 v4.3 前被稱為 Custom Frequency Upgrade 或 Superchannel，需要申請註冊，v4.3 後不需要申請 key 升級該功能
frequency-offset (*整型*	如果無線網卡工作在不同的頻率，RouterOS 會顯示，並允

[-2147483648..2147483647]; 默認: 0)	許指定偏移值。頻率變頻器被集成在無線網卡內部。例如你的網卡工作在 4000MHz，但 RouterOS 顯示為 5000MHz，設置偏移量為 1000MHz，這樣 RouterOS 會顯示正確的值。這個值以 MHz 正負為單位。
hide-ssid (*yes* / *no*; 默認: no)	*yes* - AP 不會包含 SSID 在信號標示幀內，並不會回應廣播 SSID 的請求。 *no* - AP 將包含 SSID 在信號標示幀內，並回應廣播 SSID 的請求。 這個屬性僅適用於 AP 模式，設置該參數為 yes，用戶端軟體將不會顯示該網路名稱。修改這個參數別並不會提升無線網路性能，因為 SSID 會被包含在其他幀裡。
ht-ampdu-priorities (整型列表 *[0..7]*; 默認: 0)	AMPDU 優先發送的幀會得到協商和使用(匯總幀 並通告塊發送確認)，使用 AMPDU 會增加輸送量，但可能會增加延遲，因此可能會在即時傳輸上不那麼令人滿意（語音和視頻）。因此選擇預設的 AMPDU 可以獲得較好速率
ht-amsdu-limit (*整型 [0..8192]*; 默認: 8192)	當協商時設備被允許預設值最大 AMSDU。AMSDU 聚合能明顯增加小幀的輸送量，但可能在丟失聚合幀的情況下增加傳輸延遲。發送和接收 AMSDU 都會增加 CPU 負載。
ht-amsdu-threshold (*整型 [0..8192]*; 默認: 8192)	最大幀長度允許包含進 AMSDU。
ht-basic-mcs (*列表 (mcs-0 / mcs-1 / mcs-2 / mcs-3 / mcs-4 / mcs-5 / mcs-6 / mcs-7 / mcs-8 / mcs-9 / mcs-10 / mcs-11 / mcs-12 / mcs-13 / mcs-14 / mcs-15 / mcs-16 / mcs-17 / mcs-18 / mcs-19 / mcs-20 / mcs-21 / mcs-22 / mcs-23)*; 默認: mcs-0; mcs-1; mcs-2; mcs-3; mcs-4; mcs-5; mcs-6; mcs-7)	調製與編碼，這個是每個連接用戶端都必須支持（請參考 802.11n MCS 規範）
ht-guard-interval (*any* / *long*; 默認: any)	是否允許使用監視間隔 (參考 802.11n MCS 規範可以看到如何影響輸送量)。"any" 將會使用短距離或長距離之間，這個依賴於資料傳輸率"long" 將使用長距離。
ht-rxchains (整型列表*[0..2]*; 默認: 0)	那一個天線用於接收資料
ht-supported-mcs (*列表(mcs-0 / mcs-1 / mcs-2 / mcs-3 / mcs-4 / mcs-5 / mcs-6 / mcs-7 / mcs-8 / mcs-9 / mcs-10 / mcs-11 / mcs-12 / mcs-13 / mcs-14 / mcs-15 / mcs-16 / mcs-17 / mcs-18 / mcs-19 / mcs-20 / mcs-21 / mcs-22 / mcs-23)*; Default: mcs-0; mcs-1; mcs-2; mcs-3; mcs-4; mcs-5; mcs-6; mcs-7; mcs-8; mcs-9; mcs-10; mcs-11; mcs-12; mcs-13; mcs-14; mcs-15; mcs-16; mcs-17; mcs-18; mcs-19; mcs-20; mcs-21; mcs-22; mcs-23)	設備廣播所支援的調製和編碼方式。
ht-txchains (整型列表*[0..2]*; 默認: 0)	那一個天線用於傳輸資料
hw-fragmentation-threshold (*整型*	當傳輸基於無線介質時採用 byte 為單位，指定最大分片

[256..3000] / *disabled*; 默認: 0)	資料包長度。802.11 標準資料包分片允許在無線傳輸前將資料包分片傳輸從而增強傳輸成功率(只有在分片資料包沒有被正確重新發送傳輸). 注意分片資料包的傳輸比傳輸非分片資料包更有效，因為協議開銷並在兩端增加資源耗用（傳輸和接收）
hw-protection-mode (*cts-to-self* / *none* / *rts-cts*; 默認: none)	幀保護屬性，參考幀保護介紹
hw-protection-threshold (*integer [0..65535]*; 默認: 0)	幀保護屬性，參考幀保護介紹
hw-retries (*整型 [0..15]*; 默認: 7)	在不考慮一個傳輸失敗情況下，重新發送幀的次數。當一個傳輸失敗幀被重新發送的次數。資料率降低接近掉線，幀會再次發送。在 on-fail-retry-time 的時間內，支援最低速率傳輸到目的地，出現連續三次失敗，在失敗後，幀會繼續重發，直到重新傳輸成功，或者直到用戶端在 disconnect-timeout 值後斷開連接。如果 frame-lifetime 值到期，幀也會被丟棄掉
l2mtu (*整型 [0..65536]*; 默認: 2290)	
mac-address (*MAC*; 默認:)	無線網卡 MAC 位址
master-interface (*字元*; 默認:)	當啟用 virtual-ap（虛擬 AP）時，選擇的無線網卡，virtual-ap 只能工作在 master 網卡設置為 ap-bridge、bridge 和 wds-salve 模式下。該參數僅支持虛擬 AP。
max-station-count (*整型 [1..2007]*; 默認: 2007)	最大關聯的用戶端數量，AP 之間的 WDS 連接也包括在內，該屬性可以控制連接用戶端數量，限制連接用戶端或者避免 AP 連接過於飽和。
mode (*station* / *station-wds* / *ap-bridge* / *bridge* / *alignment-only* / *nstreme-dual-slave* / *wds-slave* / *station-pseudobridge* / *station-pseudobridge-clone* / *station-bridge*; 默認: station)	選擇設備的工作模式，station 或者 AP 等 Station 模式: *station* - 基本的 station 模式，尋找並連接到可用的 AP。（不支援橋接功能） *station-wds* - 類似 station 模式，但但能與 AP 創建 WDS 連接，AP 端必須配置啟用 WDS 模式才能與 station-wds 連接。 *station-pseudobridge* - 類似 station 模式，但會加上 MAC 位址翻譯進行傳輸，允許網卡做橋接（但橋接非完整意義的透傳，因為使用了 MAC 位址翻譯） *station-pseudobridge-clone* - 類似 *station-pseudobridge*, 但使用了 station-bridge-clone-mac 位址連接到 AP。 AP 模式: *ap-bridge* - 基本的訪問節點模式，支援各種類型的連接。 *bridge* - 類似 *ap-bridge*,但被限制連接一個用戶端 。 *wds-slave* - 類似 *ap-bridge*, 但要與 AP 相同的 SSID，且要啟用 WDS 連接。如果連接丟失或者無法建立，這是將會繼續搜索相同 SSID 的 AP。wds-slave 如同 AP，但不主

	動宣告自己 SSID。如果 dfs-mode 選擇 *radar-detect*，這時 AP 將啟用 hide-ssid ，將在雷達探測週期裡無法搜索 特殊模式: *alignment-only* - 將無線網卡放入一個連續的傳輸模式，被用於校準遠端天線 *nstreme-dual-slave* - 允許這個無線網卡選擇 nstreme-dual 設置。 MAC 位址翻譯在 pseudobridge 模式下，通過檢查和建立相關 IP 和 MAC 地址的對應關係表，所有資料包發送到 pseudobridge,並將收到的 MAC 位址存儲到位址翻譯清單中，通過 MAC 位址翻譯的形式轉發資料，因此超過一個主機的橋接網路不能建立二層的協議，如透傳 PPPOE 撥號; 注意：當前 pseudobridge 不支持 IPv6。 Virtual AP 不具備這個模式屬性，他們遵從 master 網卡的模式。
mtu (*整型 [0..65536]*; 默認: 1500)	最大傳輸單元
multicast-helper (*default* / *disabled* / *full*; 默認: **default**)	當設置為 full 參數時，將發送單播目標 MAC 位址，解決在無線傳輸的組播問題。該選項僅支援在 AP 端設置，而用戶端需要配置 **station-bridge** 模式。該功能從 v5.15 開始啟用 • disabled - 禁用 helper 功能，通過組播 MAC 地址發送組播包 • full - 所有組播包 MAC 位址都轉換為單播 MAC 位址優先發送 • default - 默認選擇當前的 disabled 設置。
name (*字元*; 默認:)	無線網卡名稱
noise-floor-threshold (*default* / *integer [-128..127]*; 默認: default)	這個屬性只能在 AR5211 晶片上生效
nv2-cell-radius (*整型 [10..200]*; 默認: 30)	設置衝突時間槽的影響長度，AP 分配給用戶端的初始連接，同樣也是評估到用戶端的距離，當設置過小，遠端的用戶端將會出現連接問題或者掉線（ranging timeout 錯誤）。這個設置雖然整個運行週期內影響可以被忽略，但為了獲得最大性能，在沒有必要的情況下請不要增加該設置，因此 AP 不會預留時間，實際上不會被使用，而是分配它實際資料傳輸。 對於 AP: 遠處用戶端距離，單位 km 對於 station：無效
nv2-noise-floor-offset (*default* / *整型 [0..20]*; 默認: default)	
nv2-preshared-key (*字元*; Default:)	
nv2-qos (*default* / *frame-priority*; 默認: default)	設置資料包優先順序機制，首先資料從優先順序高的佇列發送，這時低佇列優先級數據要等到 0 佇列的優先順序到

	達。當高優先順序的連接滿時，低優先級數據不會被發送。在 AP 上使用該參數請謹慎。 frame-priority - 手的設置能被 mangle 規則調用。 default - 默認設置為小包提供最低延遲的優先順序
nv2-queue-count (*integer [2..8]*; 默認: 2)	
nv2-security (*disabled* / *enabled*; 默認: disabled)	Nv2A 安全設置
on-fail-retry-time (*time [100ms..1s]*; 默認: 100ms)	極低的速率下，在第三次發生失敗後，等待指定間隔重試時間。
periodic-calibration (*default* / *disabled* / *enabled*; 默認: default)	如果 default-periodic-calibration 屬性被啟用，則可以設置默認啟用該週期校正，這個屬性類型依賴無線網卡的類型，這個屬性僅能用於 Atheros 晶片。
periodic-calibration-interval (*integer [1..10000]*; 默認: 60)	這個屬性僅能用於 Atheros 晶片的網卡
preamble-mode (*both* / *long* / *short*; 默認: both)	短報文模式是 802.11b 標準的選項，能減低每個幀的開銷 對於 AP: *long* - 不使用短報文 *short* - 宣佈短報文功能，不接受來至沒有這個功能用戶端的連結 *both* - 對雙方宣佈短報文功能 對於 station: *long* - 不使用短報文功能 *short* - 如果 AP 不支持短報文，將不連接 *both* - 如果 AP 支持啟用短報文
prism-cardtype (*100mW* / *200mW* / *30mW*; 默認:)	指定安裝的 prism 網卡的類型（現在 prism 網卡幾乎很少見到）
proprietary-extension (*post-2.9.25* / *pre-2.9.25*; Default: post-2.9.25)	管理幀中的一個資訊單元包含了 RouterOS 的私有資訊。這個參數控制如下資訊： *pre-2.9.25* - 這個代表舊的版本 2.9.25 前，能與高版本的 RouterOS 交互資訊，該模式與一些用戶端不相容，例如 intel 迅馳（Centrino）用戶端。 *post-2.9.25* - 使用標準方式，將相容更新的無線用戶端
radio-name (*字元*; 默認: MAC 地址)	設備的名稱描述，將在無線登記表顯示（registration table）遠端設備的 MAC 位址。
rate-selection (*advanced* / *legacy*; 默認: legacy)	
rate-set (*configured* / *default*; 默認: default)	可以獲得兩個選項： *default* - 使用默認所支援的設置，basic-rates 和 supported-rates 下的參數會被自動鎖定 *configured* - 使用 use values from basic-rates 和 supported-rates 下的參數，手動調製，注意 g 模式下，同時使用"rates-b"和"rates-a/g"屬性.

scan-list (可以通過逗號分隔頻率或者"-"定義頻率範圍/ *default*; 默認: default)	*default* 值指根據無線網卡支援，且頻率模式（frequency-mode)、當前國家（country）頻率規範等設置可以獲得的頻率範圍(可以通過 info 查看"/interface wireless info> pirnt"). 默認搜索列表 5ghz 頻段下，每間隔 20MHz 步進搜索，在 *5ghz-turbo* 頻段下每間隔 40MHz，2.4G 則間隔 5MHz。如果 scan-list 採用手動設定，所有指定的頻率都會被搜索（例如：scan-list=*default,5200-5245,2412-2427* 即會使用默認頻段搜索，並添加從 5200-5245 或 2412-2427 頻率範圍.）
security-profile (*字元*; 默認: default)	從 security-profiles 獲取加密策略
ssid (*字元 (0~32 字元)*; 默認: 根據 RouterOS 的/system identity 取值)	SSID (Service Set Identifier)服務集識別字的縮寫,它是用來區分一個無線網路接入點與另一個接入點的識別字
station-bridge-clone-mac (*MAC*; 默認:)	這個屬性僅在 *station-pseudobridge-clone* 模式下生效 通過使用指定的 MAC 位址連接到 AP，如果這個值為 *00:00:00:00:00:00*, station 將會首先使用無線網卡的 MAC 位址 當在 station 內部的設備需要傳輸資料到 AP 端，這些設備的 MAC 位址將會被替換成該指定的 MAC 地址發送
supported-rates-a/g (速率清單 *[12Mbps / 18Mbps / 24Mbps / 36Mbps / 48Mbps / 54Mbps / 6Mbps / 9Mbps]*; Default: 6Mbps; 9Mbps; 12Mbps; 18Mbps; 24Mbps; 36Mbps; 48Mbps; 54Mbps)	被支援的速率清單，被用於除 *2ghz-b* 的頻段
supported-rates-b (*list of rates [11Mbps / 1Mbps / 2Mbps / 5.5Mbps]*; Default: 1Mbps; 2Mbps; 5.5Mbps; 11Mbps)	支援的速率清單，被用於 *2ghz-b*, *2ghz-b/g* 和 *2ghz-b/g/n* 頻段 bands. 兩個設備連接使用的速率要求同時被設備所支援。這個屬性僅在 rate-set 被選擇下生效
tdma-debug (*整型 [0..4294967295]*; 默認: 0)	TDMA 調試
tdma-hw-test-mode (*整型 [0..4294967295]*; 默認:)	
tdma-override-rate (*12mbps / 18mbps / 24mbps / 36mbps / 48mbps / 54mbps / 6mbps / 9mbps / disabled / ht20-mcs... / ht40-mcs...*; 默認: disabled)	
tdma-override-size (*整型 [0..4294967295]*; 默認:)	
tdma-period-size (*整型 [1..10]*; 默認: 2)	指定 TDMA 以毫秒為週期，能幫助長距離的傳輸，有助於增加頻寬，但同時也會增加延遲
tdma-test-mode (*整型 [0..4294967295]*; 默認: 0)	

tx-power (; Default:)	無線網卡發射功率設置
update-stats-interval (; 默認:)	多長時間要求用戶端更新信號強度和 CCQ 值 打開 registration-table 同樣可以更新，這個屬於 RouterOS 擴展屬性
wds-cost-range (*disabled* / *time [10s..5h]*; 默認: disabled)	橋接（Bridge）的 WDS 連接成本開銷（Port cost）自動調整，該值通過測量鏈路的輸送量。如果超過 10%的變動，或者超過 20 秒沒有做調整，Port cost 每 5 秒被重新計算和調整 如果設置該參數為 0，即禁用自動成本調整 自動調整不能工作在手動配置的一個 bridge 端口下，即需要指定 wds-default-bridge
wds-default-bridge (*字元* / *none*; 默認: none)	當 WDS 連接已建立，並且 WDS 介面進入運行狀態(*running*),即可以將你預先設定的 bridge 介面添加到這個屬性中。當 WDS 連接斷開，會自動從 bridge 中將 WDS 介面刪除。如果這個 WDS 介面被手動添加到 bridge 中，將不會再次添加
wds-default-cost (*整型 [0..4294967295]*; 默認: 100)	初始化 bridge 介面的 WDS 連接成本（cost），注：有線以太網預設為 10，無線默認為 100
wds-ignore-ssid (*yes* / *no*; 默認: no)	默認為 no，兩個 AP 能創建在相同的頻率下，且都擁有相同的 SSID。如果這個屬性被設置為 yes，這時 SSID 將不會監測遠端的 SSID，這個屬性不會在用戶端為 station-wds 模式下生效，同樣不能工作在 wds-mode 為 *static-mesh* 或者 *dynamic-mesh*.
wds-mode (*disabled* / *dynamic* / *dynamic-mesh* / *static* / *static-mesh*; 默認: disabled)	控制 WDS 如何連接其他設備 (AP 和用戶端的 station-wds 模式)。 *disabled* - 不啟用 WDS 連接 *static* - 僅允許 WDS 連接通過手動配置，即需要添加 WDS 介面綁定對方 MAC 位址 *dynamic* - 允許 WDS 連接，不用手動增加 WDS 介面，自動建立連接，並且當連接斷開後會自動從無線網路清單中刪除 *-mesh* 模式，使用更好的方式在 AP 間建立連接，不相容非 mesh 的 AP 模式，這個方式是為避免單邊的 WDS 連接創建在兩個 AP 中的一個。 當 AP 或者 station 與其他 AP 建立 WDS 連接，通過使用 connect-list 檢查這個連接是否被允許；如果 station 通過 station-wds 模式連接到 AP，AP 可以通過 access-list 這個 station 是否被允許加入
wmm-support (*disabled* / *enabled* / *required*; 默認: disabled)	指定是否啟用 WMM

第五章 RouterOS WiFi 覆蓋配置

5.1 WiFi 覆蓋介紹

我們通常說的 WiFi 覆蓋指的是通過 AP 為終端使用者，如筆記型電腦、安裝無線網卡的臺式 PC、支援 WiFi 的手機、Pad 等終端設備。即這些終端設備接收到 AP 信號後，不再進行資料轉發，資料在此處理和終結。

WiFi 的覆蓋我們一般區分為室內和室外覆蓋，我們在之前的 RouterOS WiFi 分析提到 RouterOS 802.11 協議中優勢主要是 WLAN 設備間的傳輸，而對於終端使用者的覆蓋相對較弱，並不代表 RouterOS 覆蓋方面完敗與其他任何設備，只是相對於純 AP 設備覆蓋抗干擾性和優化上來說沒有特別明顯的優勢，但在各種功能上有無法比擬的優勢。

我們這裡講解的是 RouterOS 的基本 WLAN 覆蓋，這裡我稱為 WiFi 覆蓋，比較貼近流行的終端覆蓋應用，我們把 WiFi 覆蓋分為室內和室外，室內覆蓋和室外覆蓋區別在於環境結構上的區別，室內覆蓋和室外覆蓋都需要進行具體的分析！在我們之前的 RouterOS 分析中我已經提到關於 RouterOS 在覆蓋方面具體問題。

室內覆蓋

在室內覆蓋上，結構比較覆蓋，我們選擇室內的中間區域進行覆蓋，在室內安放多個 AP 時，考慮 AP 覆蓋的均勻性。室內覆蓋對 RouterOS 選擇無線網卡特別重要，即無線網卡的發射功率和接收靈敏度。我建議選擇 200-350mw 無線網卡，接收靈敏度的要求就特別考驗無線網卡了，很多無線網卡雖然發射功率很強，但接收靈敏就不是那麼如人意！發射功率代表你說話，接收靈敏度就代表你聽力！即使你聲音很大所有人都能聽到，但別人聽到後回復你，你卻聽不到，那這個通信交流也是失敗的。

室外覆蓋

室內覆蓋選擇天線多為全向天線，增益從 6-12dBi，我們選擇天線肯定希望發射增益越大越好，因為我們知道增益可以增加信號在某一方向的強度，如果是 1-2 個 AP 到可以選擇較強的信號，但多了就要考慮 AP 之間的相互干擾性，畢竟 AP 眾多就向之前提到的噪音相互干擾。

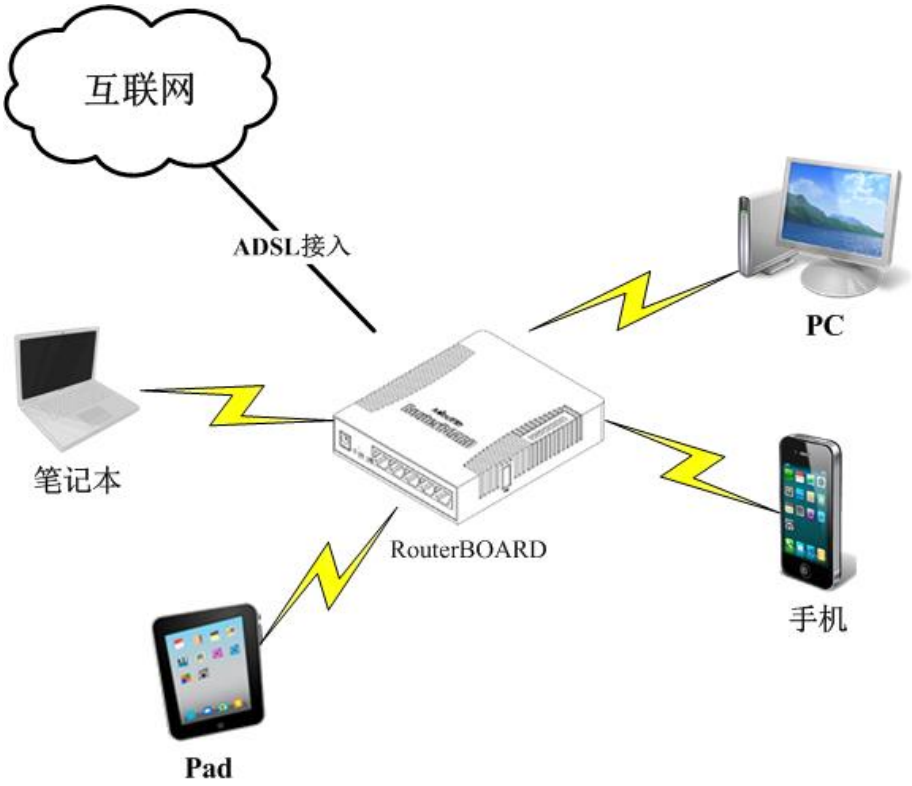
WiFi 的覆蓋一般有路由和橋接模式，大多廠商的 AP 設備預設採用橋接，但隨著市場和客戶需求的多樣化，AP 的功能也在增加。對於 RouterOS 來說你不用擔心他的功能，因為在某種程度上講 RouterOS 是“全能”的，這個可能初學者是最有體會的，你拿家用的 TP-LINK 無線路由器配置介面和 RouterOS 的 winbox 介面一對比，嚇你一跳！因為 RouterOS 集成了各種網路常見功能。如果你對 OSI 七層模型或者 TCP/IP 協定不太瞭解的話，可能理解路由和橋接模式比較困難，那建議你去百度或者 google 下，因為這對於無線網路應用非常重要，也是網路入門的基礎，我建議你仔細掌握資料連結層、網路層和傳輸層，什麼是交換機、橋接器；什麼是路由器等，這裡我就不再多講，你可以去參考下我的 RouterOS 網路教程，簡單講訴了基本的網路知識，畢竟這裡我們的重點是在講 RouterOS 的 WiFi 配置。

5.2 RouterOS WiFi 覆蓋事例

普通 WiFi 上網

RouterOS 無線覆蓋我們先從普通無線路由器最常見的路由方式說起，這個方式配置步驟和家庭的無線路由器一樣，但不同的是 RouterOS 配置過程和介面相對複雜，例如：我們辦公室內通過運營商光貓連接，通過

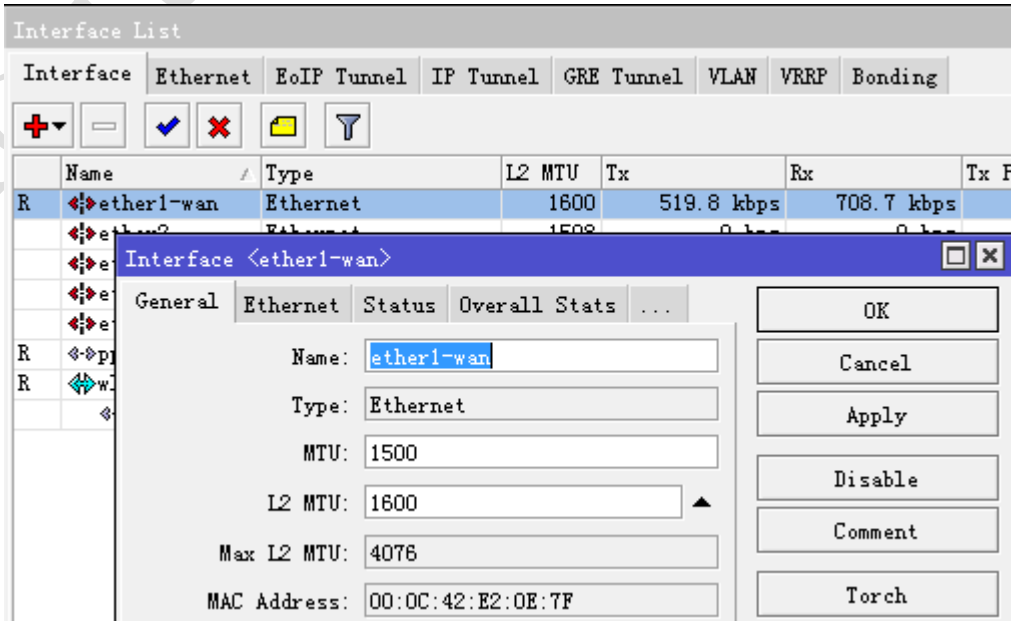
PPPoE 撥號上網，RouterOS 需要至少一個以太網口，然後通過一個 802.11bgn 的無線網卡覆蓋辦公室，如下圖：



普通的步驟是

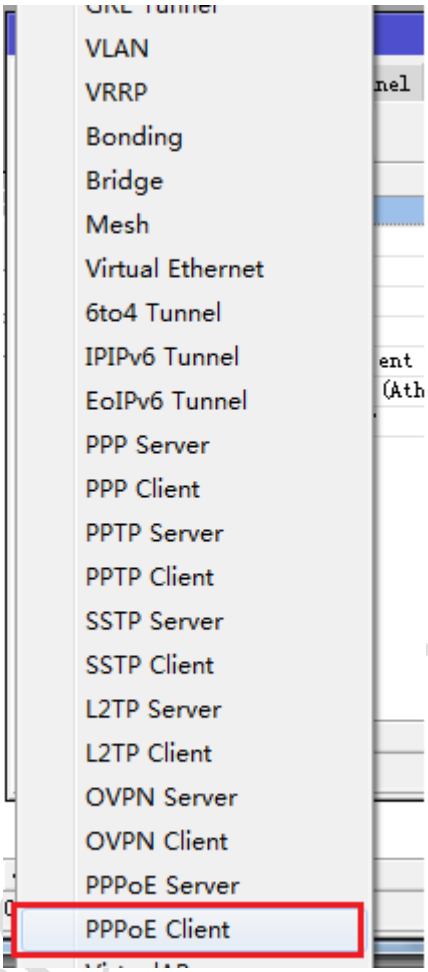
1、 配置 ADSL 撥號的 PPPoE
2、 配置我們 RouterOS 上的 wlan1 無線網卡採用 802.11bgn 參數
3、 配置 wlan1 無線網卡的 IP 位址為 192.168.100.1/24
4、 配置 DHCP 伺服器和啟用 DNS 緩存
5、 配置 ip firewall nat 的位址隱藏

那麼我們首先來配置 RouterOS 的 PPPoE 撥號，我們登錄 winbox 進入 interface 介面

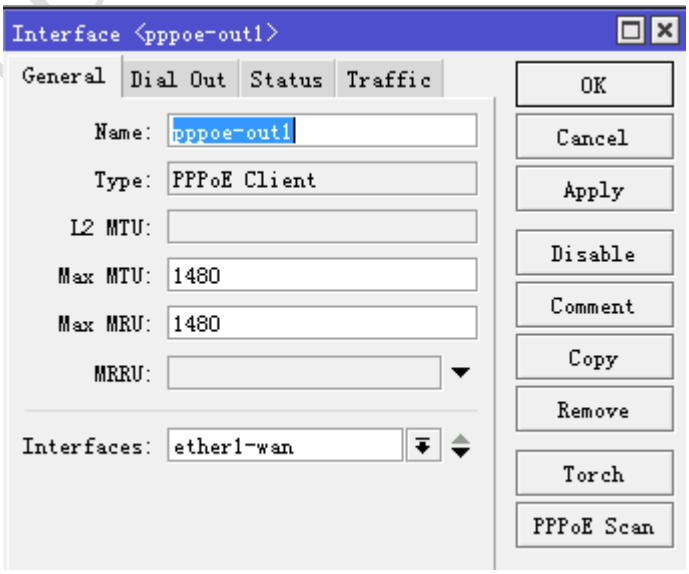


我習慣選擇 ether1 配置為外網接入網口，並取名為 ether1-wan

然後添加點擊加號添加 pppoe-client



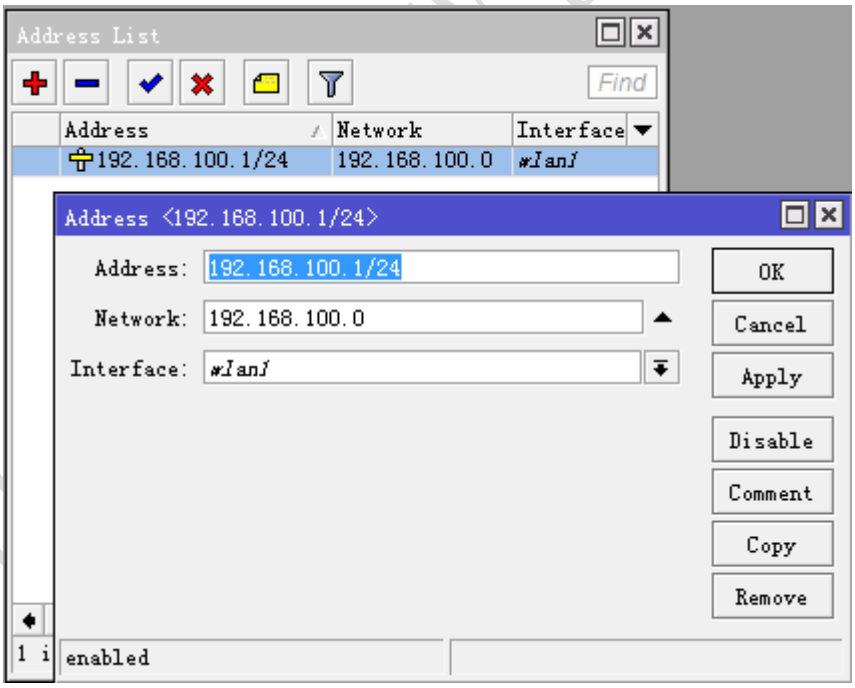
進入 pppoe-out1 的配置介面，選擇 interface=ether1-wan，其他參數默認



選擇 Dial-out 介面，設置帳號 88888888 和密碼 12345678，即 user 和 password，選擇 Add default router 添加默認路由，記住 user peer DNS 要選擇上，該參數是使用對端伺服器分配的 DNS

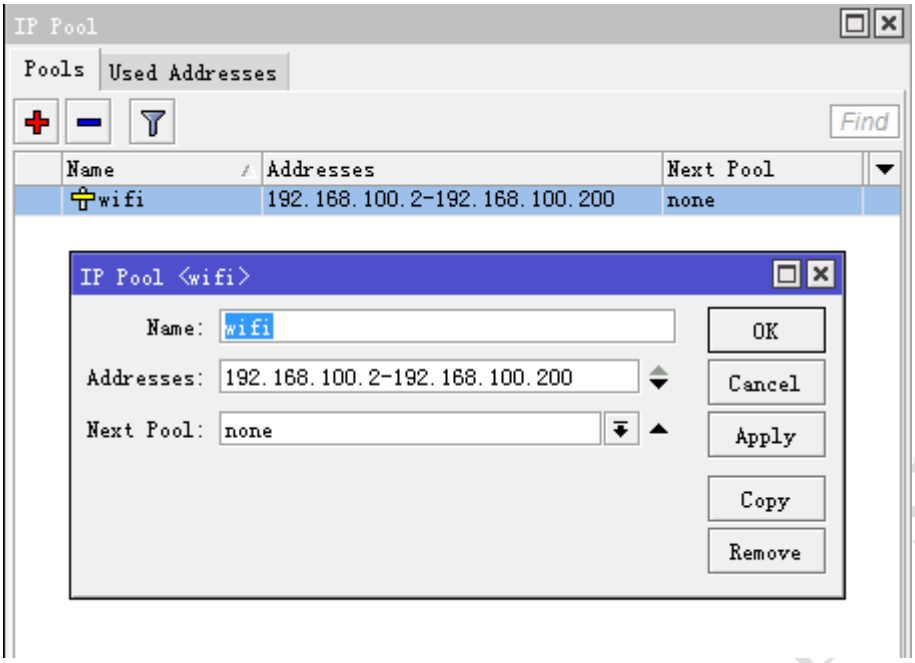
這樣 pppoe 撥號設置完成，當連接上 ADSL Moden 後，RouterOS 會自動撥號，並獲取 IP 位址

我們給 wlan1 網卡配置 IP 位址 192.168.100.1/24

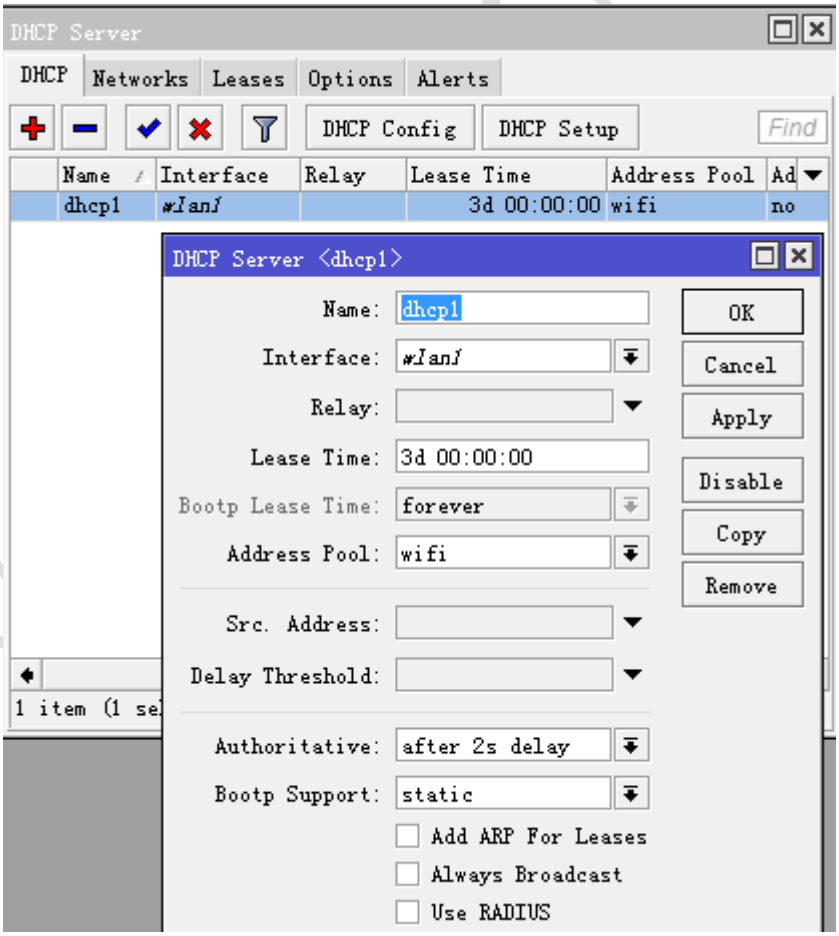


既然是 SOHO 的無線上網，用戶肯定不用手動分配 IP 位址，需通過配置 DHCP 服務，向使用者分配位址

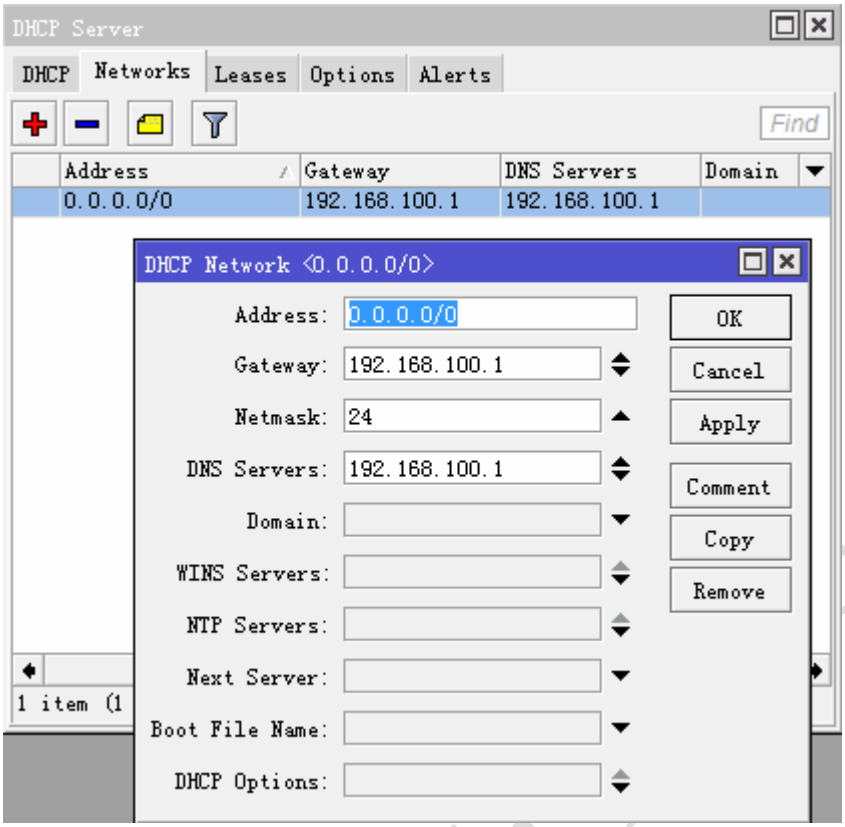
首先定義位址集區 pool，進入 ip pool 中添加分配給用戶的地址段 192.168.100.2-192.168.100.200



進入 ip dhcp-server 建立 DHCP 服務，選擇我們剛才配置的 wifi 位址集區

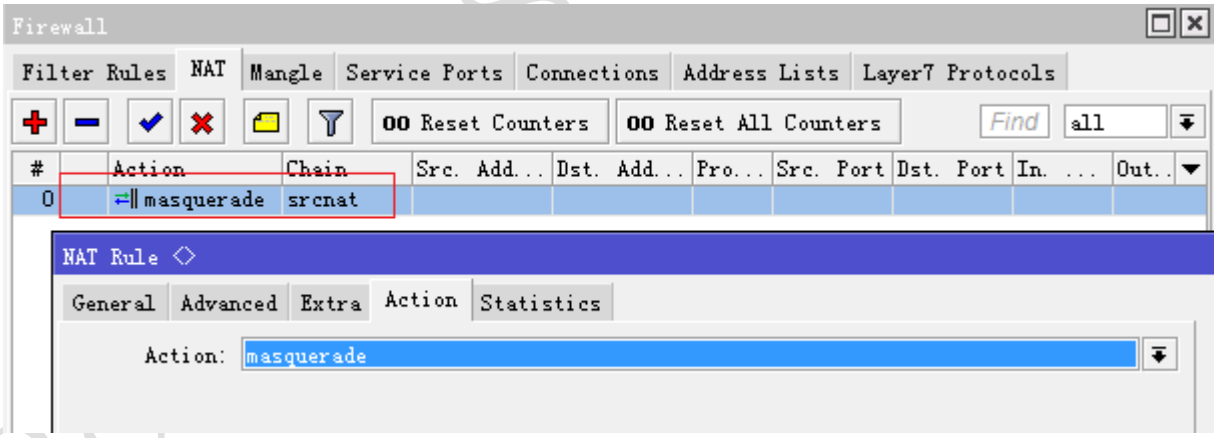


進入 network 選項，配置分配給使用者的的閘道、子網路遮罩和 DNS，這裡我們啟用了 DNS 緩存，直接使用閘道作為用戶 DNS

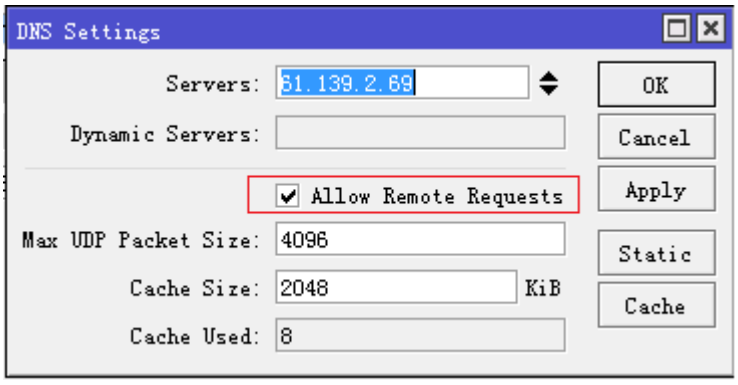


配置 nat 規則

進入 ip firewall nat，添加一條 srcnat 規則為 masquerade，轉換內網 IP 到外網



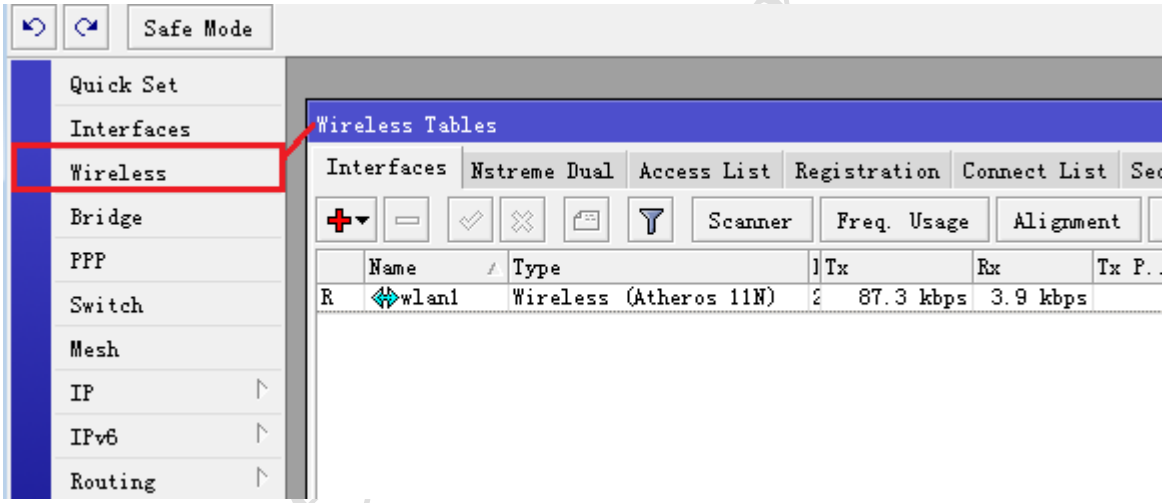
配置 DNS 緩存，即內網用戶可以使用 192.168.100.1 作為 DNS



無線網卡配置

下一步，我們需要配置無線網卡，在 RouterOS 中所有 802.11 協定的無線網卡都識別為 wlan，根據數量的多少順序編號，這裡我們只有一張無線網卡，預設名稱為 wlan1

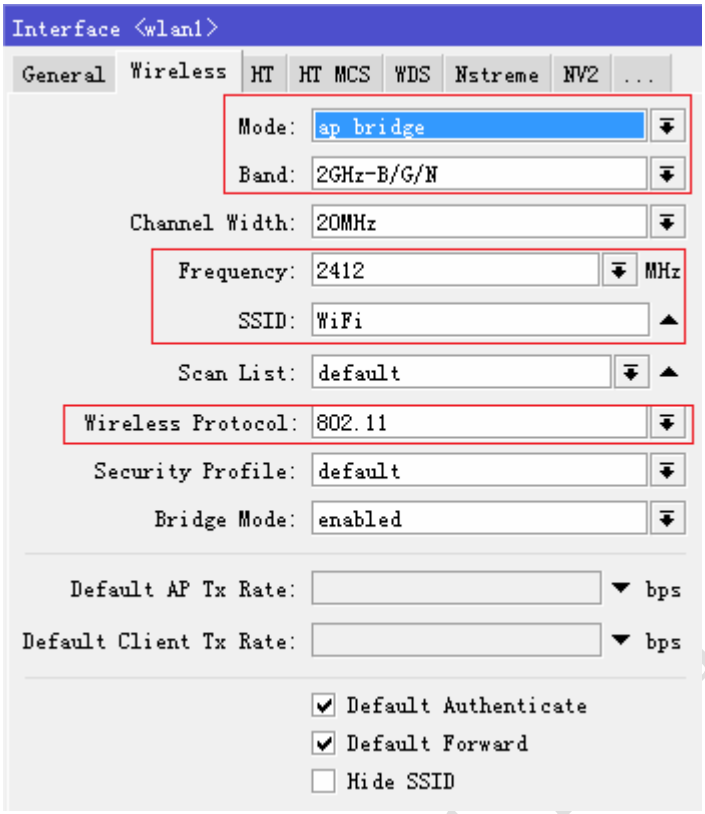
我們進入 Wireless 功能表，可以看到無線配置清單，裡面有一張 wlan1 網卡，從 Type 屬性可以看到無線網卡為 Atheros 11N 網卡



所有 WiFi 設備通用的配置基本上需要設置一下參數：

1、Mode：無線模式，覆蓋都採用 ap-bridge 模式；
2、Band：使用的頻段可以選擇 2GHz 或者 5GHz，普通的 WiFi 覆蓋通常採用 2GHz；
3、Frequency：無線發射頻率，選擇 1-11 頻段，特定的國家可以增加 3 個頻段；
4、SSID：即 Service Set Identifier 的縮寫，即無線網路名稱，建議不要設置中文，以免出現識別問題。

下面是進入 wlan1 的 Wireless 專案下的配置參數

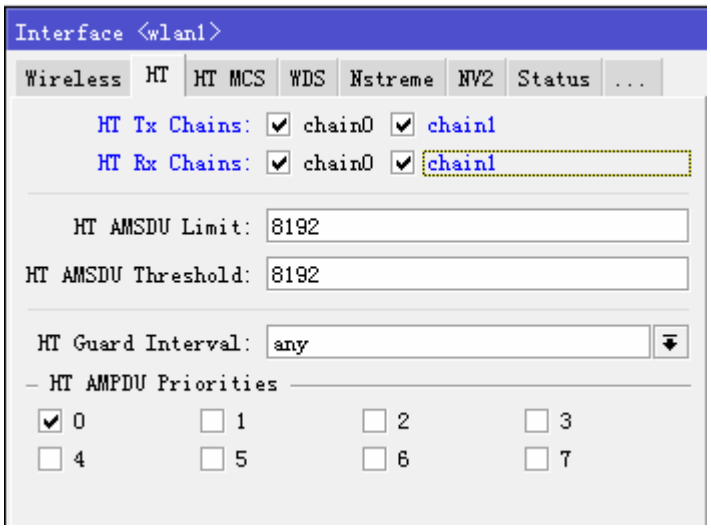


Wireless-protocol 參數

從 5.0rc1 開始加入了新的 wireless 設置參數 wireless-protocol，注意設置將根據所需要的無線網路模式，例如你需要相容那幾種模式，如下表：

值	AP	client
unspecified	基於老版本的 nstreme 或者 802.11	連接到老版本的 nstreme 或者 802.11
any	如同 unspecified	搜索所有匹配的網路，不論協定。
802.11	建立 802.11	只能連接到標準的 802.11 網路
nstreme	建立 Nstreme	只能連接到 Nstreme
nv2	建立 NV2	只能連接到 NV2
nv2-nstreme-802.11	建立 NV2	搜索 Nv2 網路，如果找到有適當的網路，並連接。 否則搜索 Nstreme 網路，如果找到有適當的網路，並連接。 否則搜索 802.11 網路，如果找到有適當的網路，並連接。
nv2-nstreme	建立 NV2	搜索 Nv2 網路，如果找到有適當的網路，並連接。 否則搜索 Nstreme 網路，如果找到有適當的網路，並連接。

採用 802.11n 的協定，需要配置 HT 參數，即採用 MIMO 是 1x1，還是 2x2 模式，如果你是 2x2 的 MIMO 設備，選擇 chain0 和 chain1

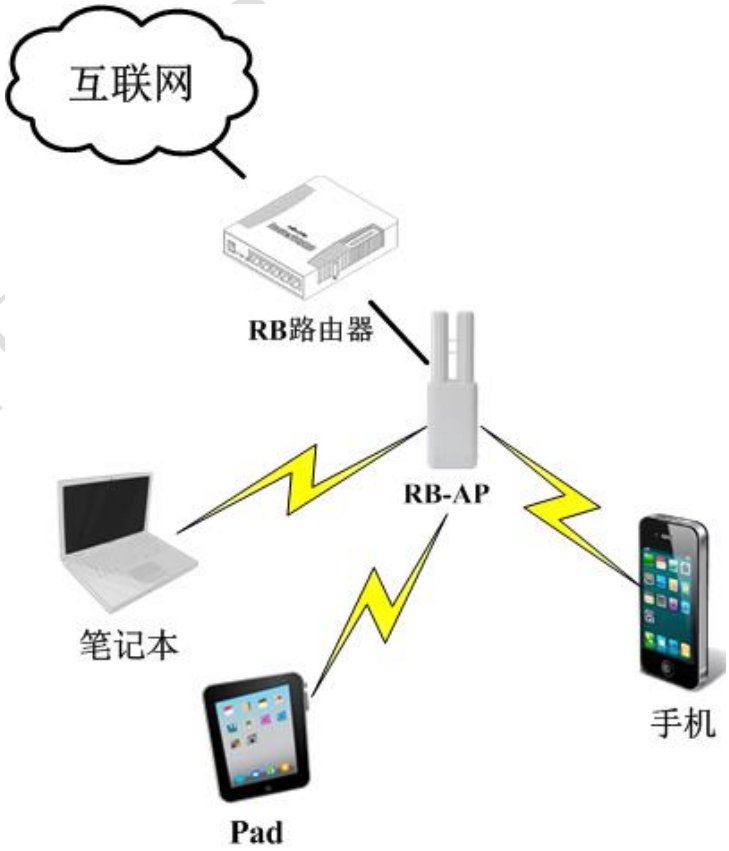


這樣一個 SOHO 的 WiFi 覆蓋就配置完成。

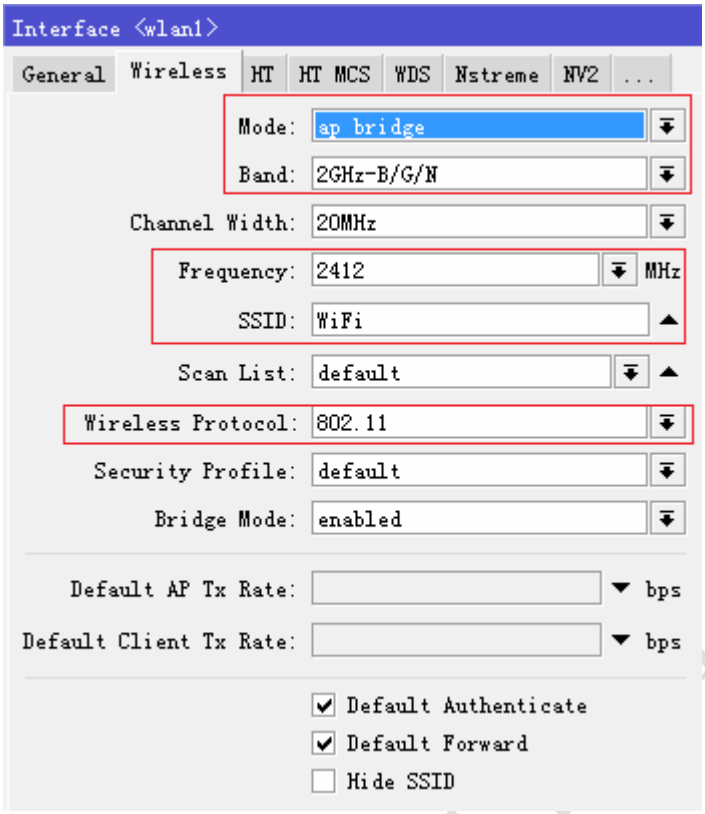
基於橋接器的覆蓋

橋接器覆蓋，是在室內或周圍空間較大，需要多個 AP 覆蓋時採用的方案，且接入使用者較多，每個設備僅需要做普通橋接器使用，配置比 SOHO 方式簡單。

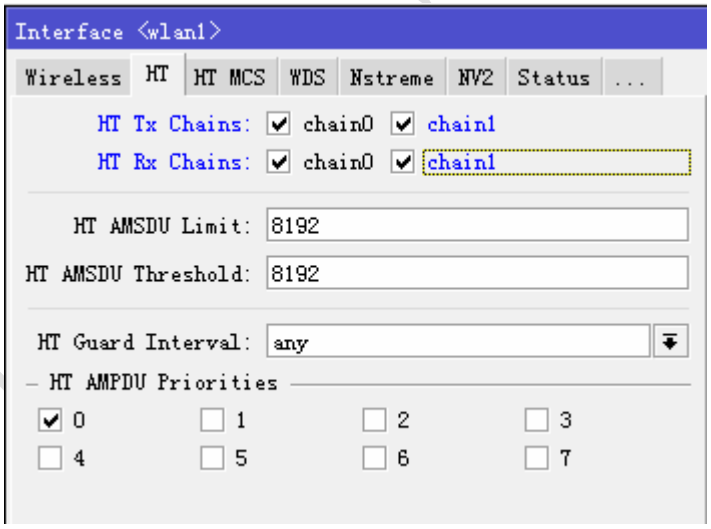
橋接器覆蓋與 SOHO 覆蓋區別是將路由和 WiFi 分離開，路由負責外網連接，AP 負責 WiFi 覆蓋



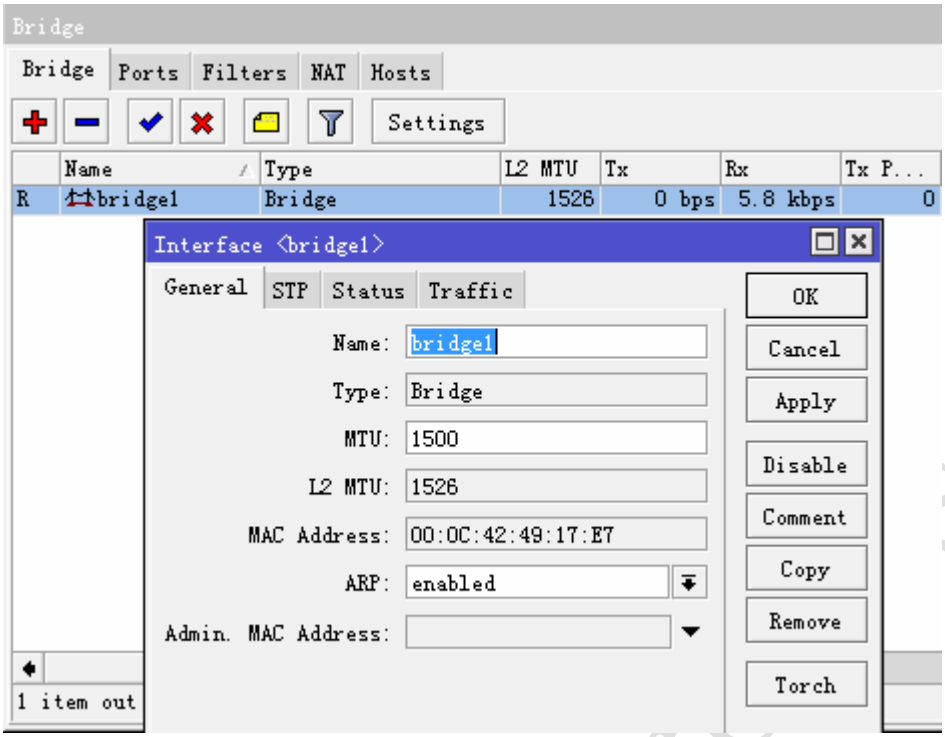
配置 wlan1 參數，基本與 SOHO 方式配置一樣：



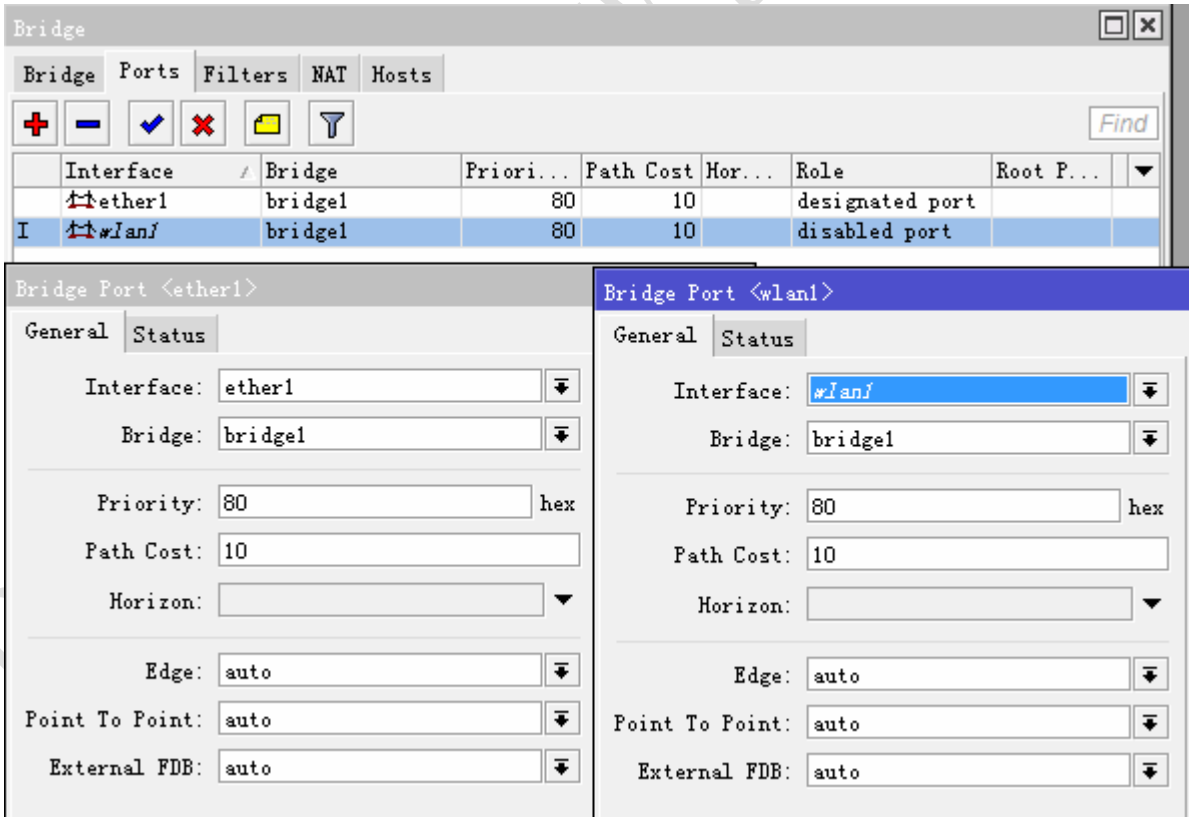
配置 HT 參數：



與 SOHO 的區別就在與配置橋接器，我們需要進入 bridge 功能表下，添加 bridge1 規則

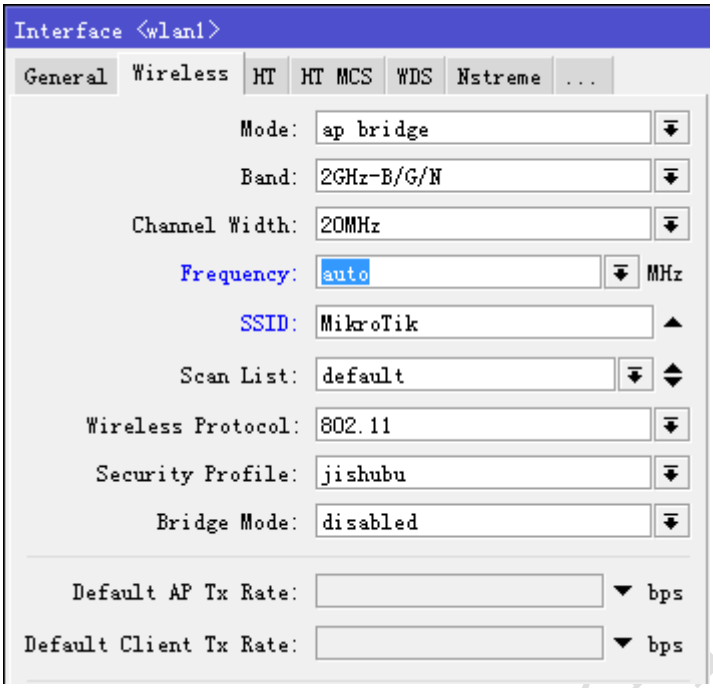


將 ether1 和 wlan1 添加到橋接器 bridge1 中



自動頻率選項

從 RouterOS v6.12 開始，你可以設置 AP 發射頻率為“auto”（自動），這個特性可以避開干擾，並提升你的無線網路性能。無線頻率 auto 設置簡化了操作，RouterOS 會檢查該無線網路區域，並選擇一個頻率，遠離區域中的其他無線網路頻率。設置如下



PS:這個功能只是方便大家設置網路，不過在複雜的無線網路環境中，也很難起到作用。

5.3 Access List 存取控制清單

操作路徑: /interface wireless access-list

Access list 用於 AP 限制其他設備的連接，通過訪問列表控制終端設備的各種參數。Access list 執行過程：

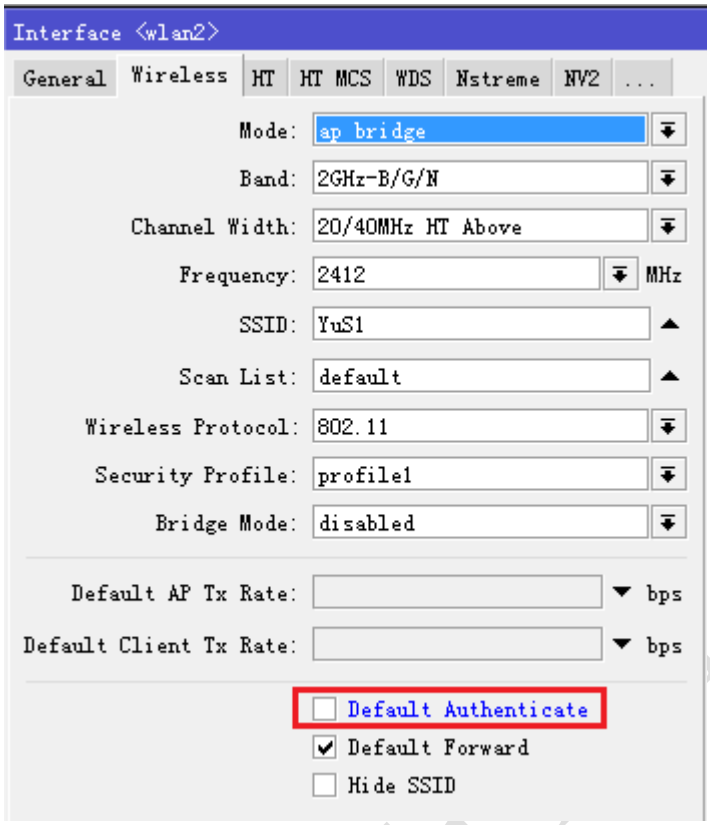
- Access list 規則通過迴圈檢測是否存在匹配的規則
- 被禁用的規則會被忽略執行
- 當存在多條相同規則時，僅匹配從上往下的第一條規則
- 如果沒有針對遠端連接的匹配的規則，會使用預設的無線配置
- 當在 access list 的規則中選擇了 **authentication**=*no*，那麼該匹配規則的無線用戶端將被拒絕連接

屬性	描述
ap-tx-limit (*整型 [0..4294967295]*; 默認: 0)	速率控制，限制發送到用戶端的速率，限制用戶端下行。當設置為 0 時，表示不限制速率，單位為 bits。
authentication (*yes* / *no*; 默認: yes)	用戶端驗證程式 *no* - 用戶端總是被拒絕連接 *yes* - 啟動驗證程式，指定該介面相應的 security-profile 安全性原則，如果沒有加密，使用預設配置
client-tx-limit (*整型 [0..4294967295]*; 默認: 0)	控制用戶端發出的速率，限制用戶端上行。當設置為 0 時，表示不限制速率，單位為 bits。 這個屬性僅支援 RouterOS 的用戶端

comment (*字元*; 默認:)	注釋說明
disabled (*yes* / *no*; 默認: no)	禁用規則
forwarding (*yes* / *no*; 默認: yes)	資料轉發，類似與用戶端之間資料隔離 *no* - 用戶端不能與其他用戶端交換資料，僅能與相連的 AP 通信 *yes* - 用戶端能通過相連的 AP 與其他用戶端交換資料
interface (*字元* / *all*; 默認: all)	當規則設置為 interface=*all*，即匹配所有的無線網卡，如果要匹配指定的網卡，可以通過該屬性選擇
mac-address (*MAC*; 默認: 00:00:00:00:00:00)	規則會匹配指定的用戶端 MAC 地址，即對用戶端 MAC 地址綁定。
management-protection-key (*string*; Default: "")	
private-algo (*104bit-wep* / *40bit-wep* / *aes-ccm* / *none* / *tkip*; 默認: none)	僅 WEP 加密模式支援
private-key (*字元*; 默認: "")	僅 WEP 加密模式支援
private-pre-shared-key (*字元*; 默認: "")	被用於 WPA 的 PSK 模式
signal-range (*數值範圍 - 指定數值範圍在 -120..120*; 默認: -120..120)	規則匹配用戶端信號強度是否在指定的範圍內 如果用戶端的信號超出了這個指定範圍, AP 將與用戶端斷開連接
time (*時間範圍,sun,mon,tue,wed,thu,fri,sat --一天的時間選擇*; 默認:)	規則匹配指定的時間週期 AP 將在指定的時間過後斷開與用戶端的連接。

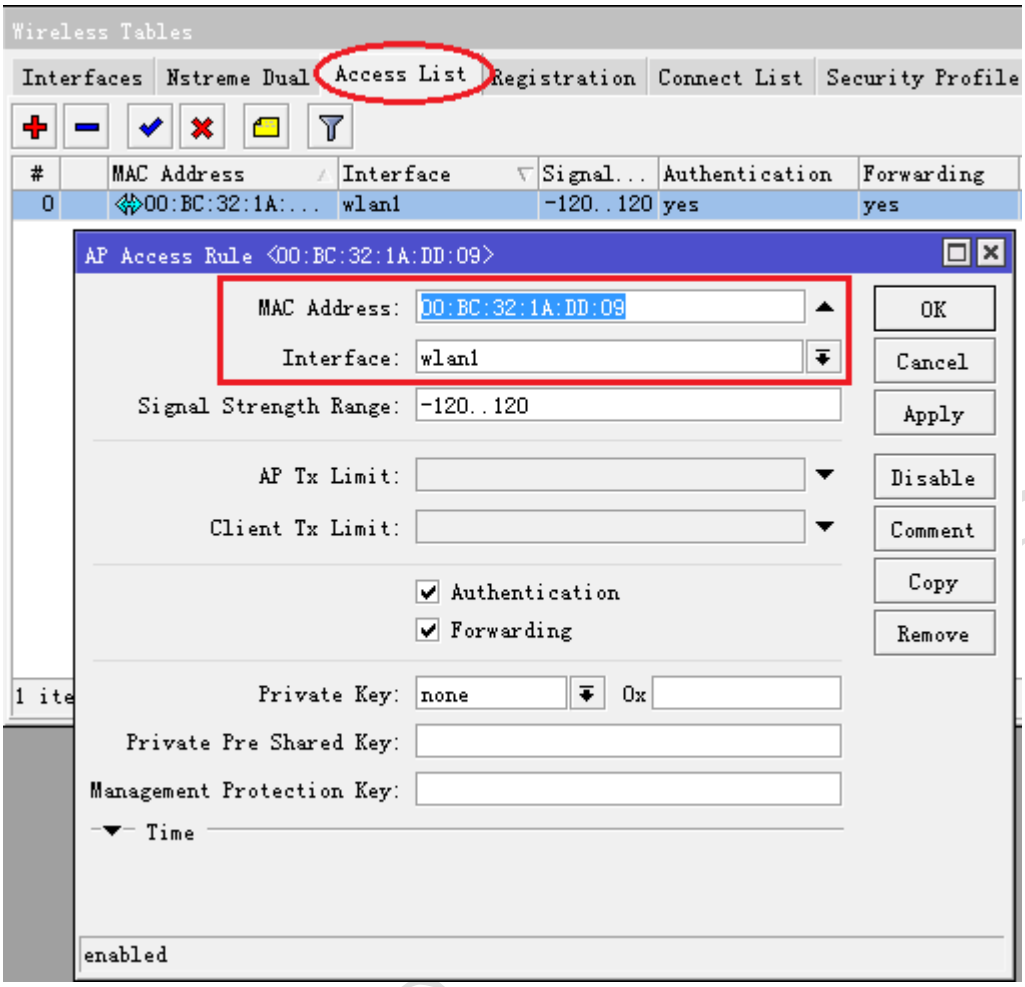
如何使用 Access-list 控制用戶端

要讓 access-list 中的規則生效，我們需要將 wireless 功能表下的 default-authenticate 參數選擇為 no，即預設連接情況下用戶端不允許自動驗證通過

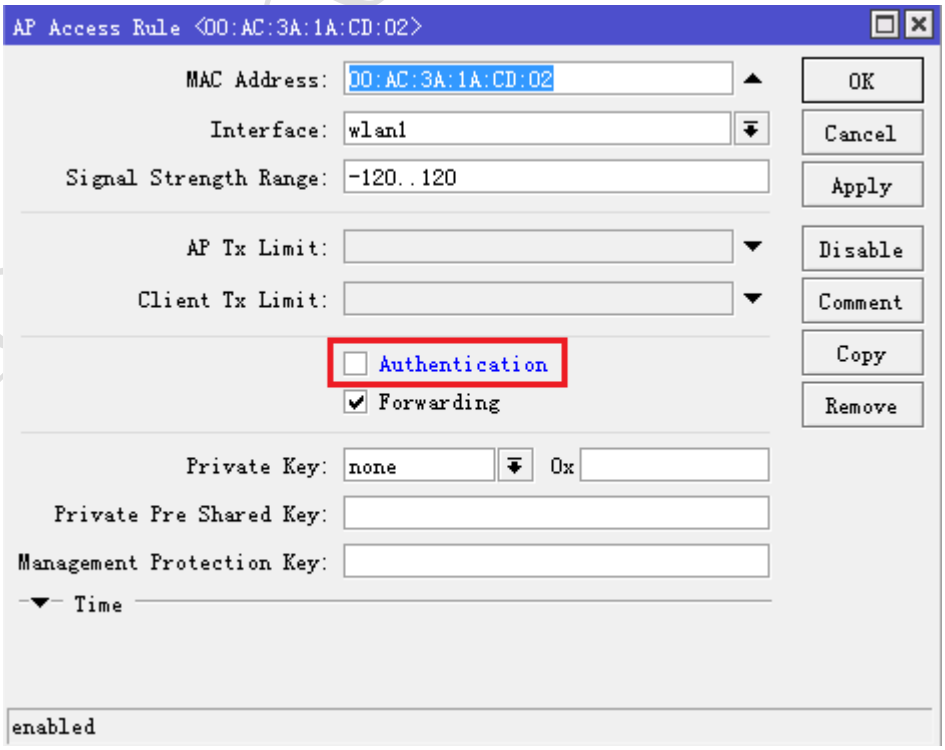


當我們關閉掉 default-authenticate，所有連接 AP 的用戶端或者 station 要會進入 access-list 進行匹配，如果沒有匹配的設備將無法連接到 AP。這樣的操作類似於我們有線網路中通過 MAC 位址綁定電腦一樣

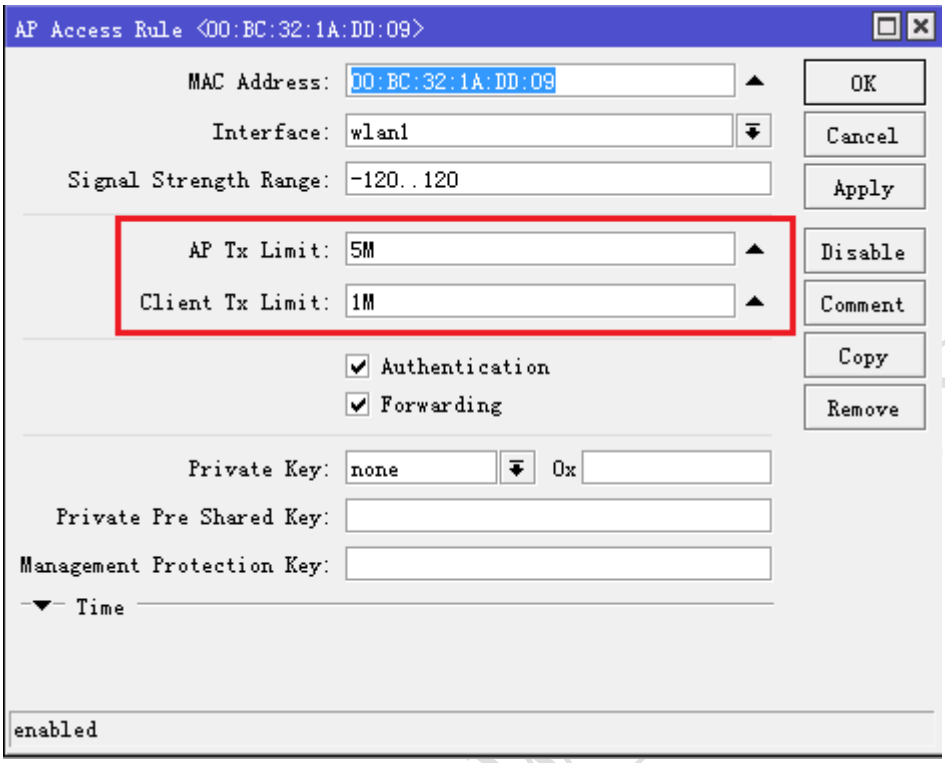
假如我們有這樣一個用戶端要對其進行連接控制，MAC 位址為：00:bc:32:1a:dd:09，連接無線網卡 wlan1



在 access-list 規則中的 authenticate 參數，當我們在規則中關閉後，表示對該用戶拒絕連接到 AP，例如我們要禁止 MAC：00:AC:3A:1A:CD:02 的連接



當遠端設備是 RouterOS 的 station，我們可以通過 ap-tx-limit 和 client-tx-limit 限制 station 的連接速率



如上圖，我們限制了 00:BC:32:1A:DD:09 的 station 的速率，AP 發向 station 的頻寬為 5Mbps，station 發向 AP 的頻寬為 1M，即對於 station 而言下載為 5Mbps，上傳為 1Mbps。

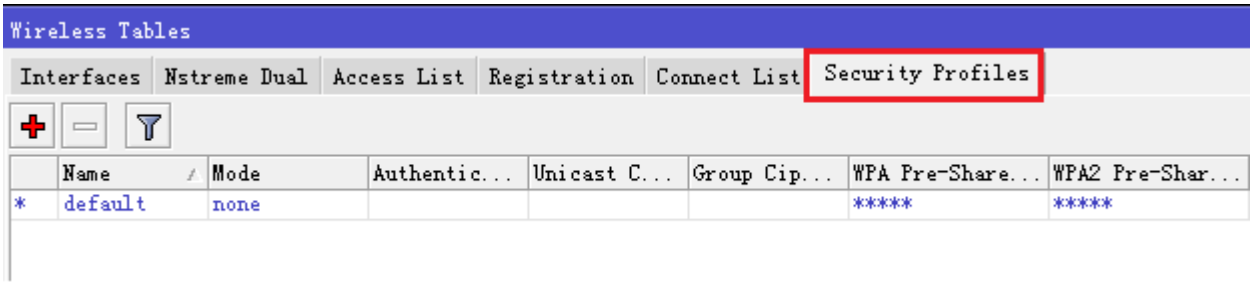
5.4 安全性原則

操作路徑: /interface wireless security-profiles

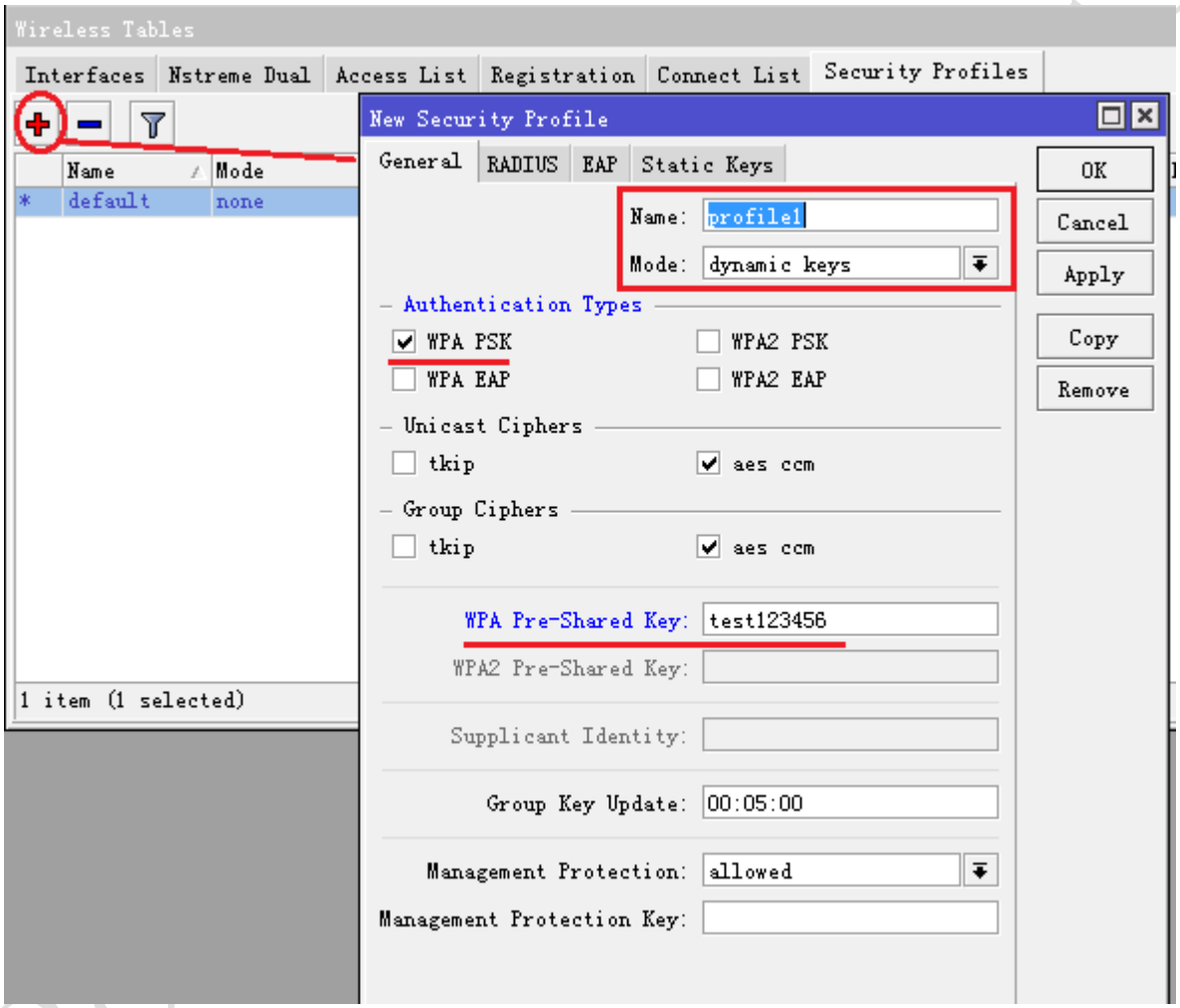
安全性原則是在**/interface wireless security-profiles** 路徑下配置，這裡我們可以配置 802.11 傳輸協定的加密方式。配置安全性原則規則定義不同的加密方式，規則被定義後，可以被應用到 Wireless 配置視窗裡的 security-profile 參數

- **mode** (none, static-keys-optional, static-keys-required 或 dynamic-keys; 默認為: none)：
- **none** - 不採用任何加密。
- **static-keys-required** - 採用靜態加密的 WEP 模式，不接收也不發送為加密的幀。如果設備採用 station 模式下選擇 static-keys-required，將無法連接到一個採用 **static-keys-optional** 的 AP 設備
- **static-keys-optional** -採用靜態加密的 WEP 模式，支援加密和解密
- **dynamic-keys** - 動態加密的 WPA 模式

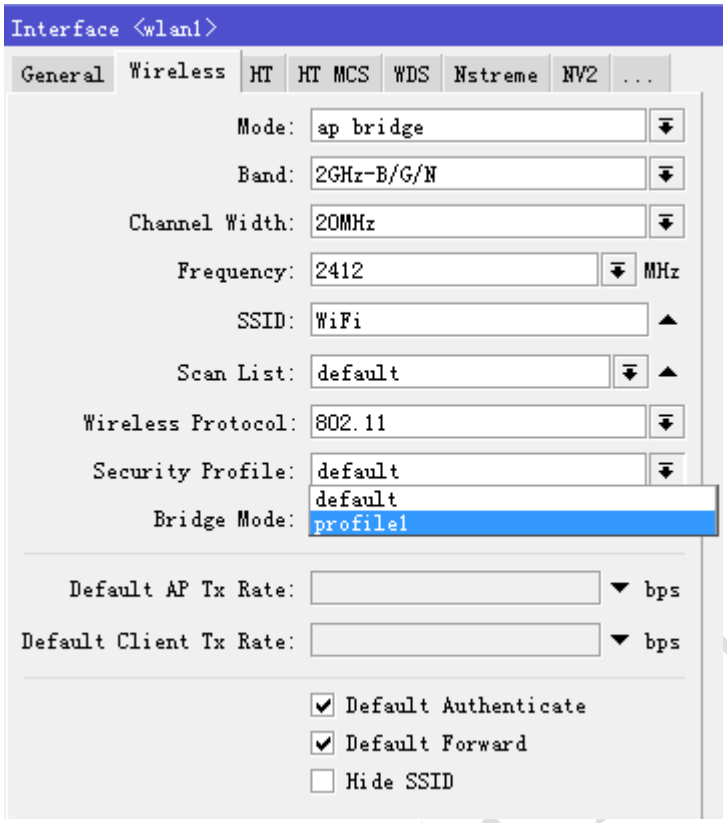
這裡我們以 dynamic-keys 為事例，演示下 WPA 模式的配置，我們首先進入 Security Profiles 功能表下



新增加一條規則 profile1，選擇 mode=dynamic keys，Authentication-types=WPA PSk，Unicast-Ciphers=aes ccm，並設置 WPA Pre-shared Key=test123456



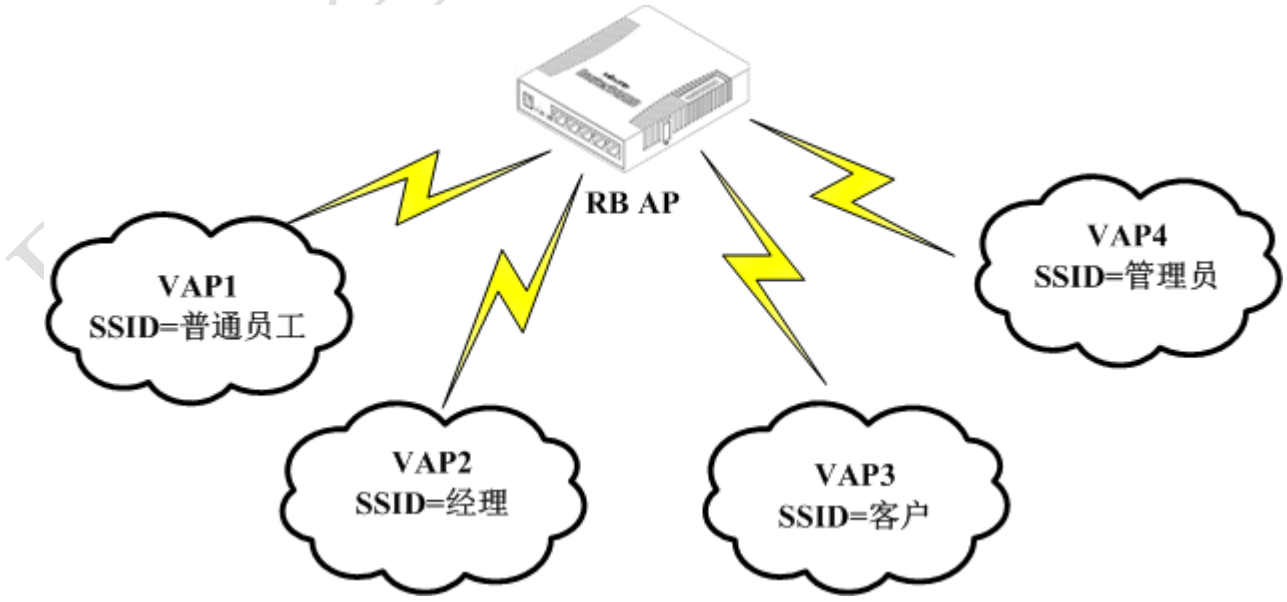
進入 wireless 菜單下，將 security-profile 設置為 profile1



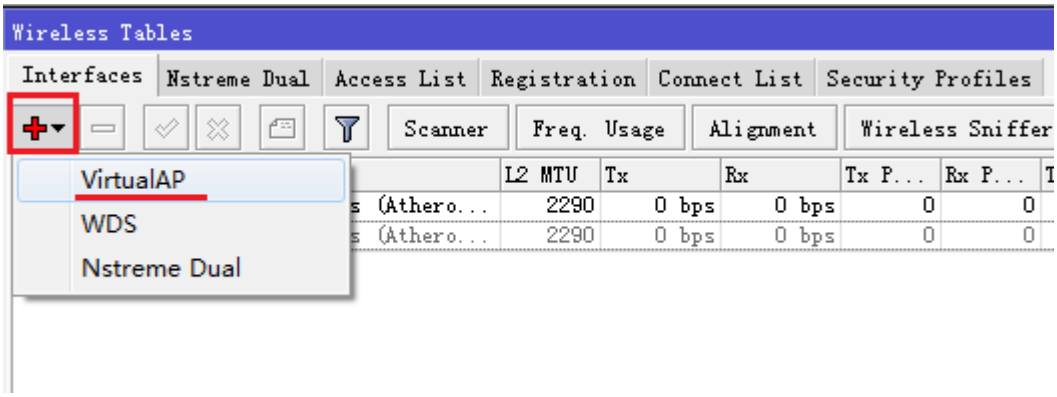
5.5 虛擬 AP(VAP)

當我們想通過一台 AP 在一個區域內建立多個 SSID，並對不同的 SSID 下的用戶進行管理，例如：辦公區域裡，可以區分為員工、經理和老闆的 WiFi 上網，在家庭中可以用來區分父母和子女的 WiFi 上網

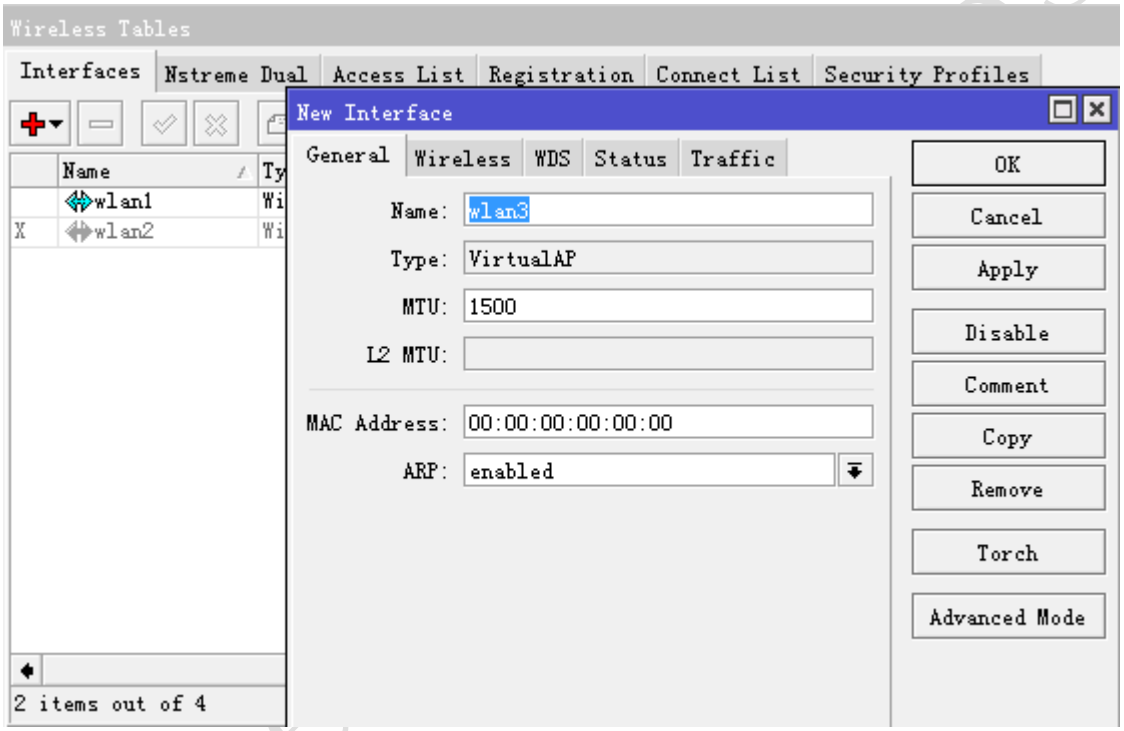
虛擬 AP 的好處在與通過一個物理網卡，類比出多個 AP 信號，在一個區域內廣播多個 SSID，讓不同的用戶選擇對應的 SSID。當然虛擬 AP 也支援加密的安全性原則，有助於你對 WiFi 網路的區域劃分和管理。



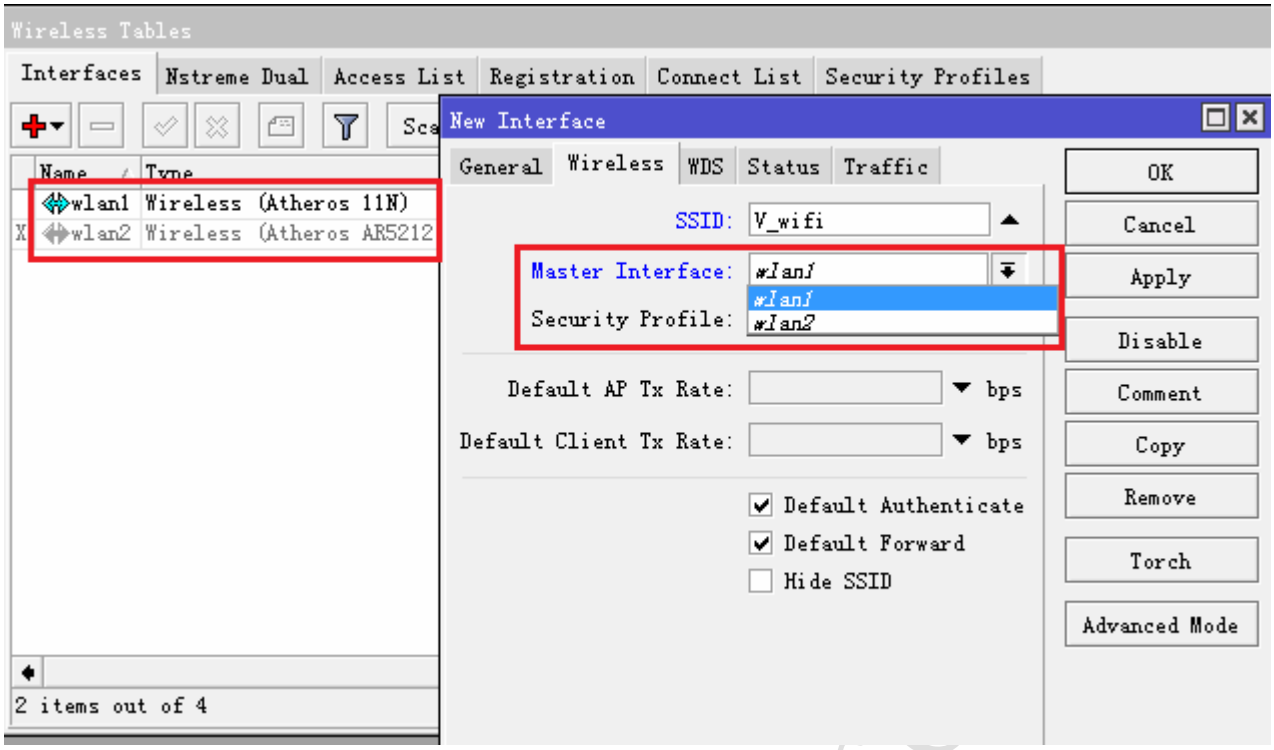
我們建立一個虛擬 AP，可以進入 wireless 目錄下，點加號可以找到 VirtualAP



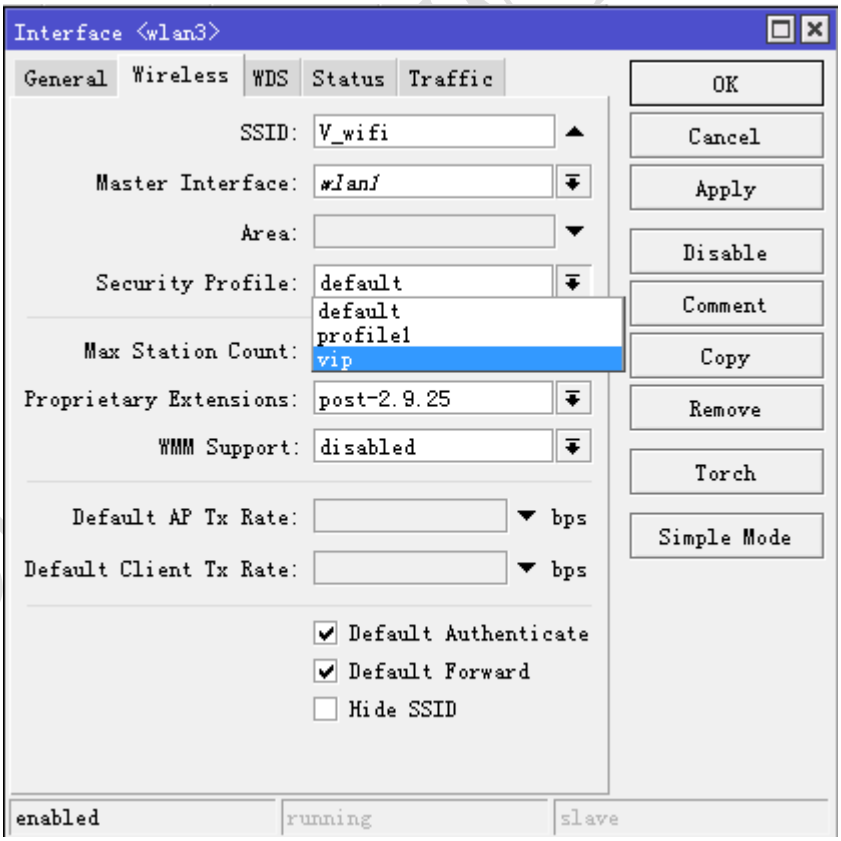
打開後，我們新建一個名為 wlan3 的虛擬 AP，可以看到 type 為 VirtualAP



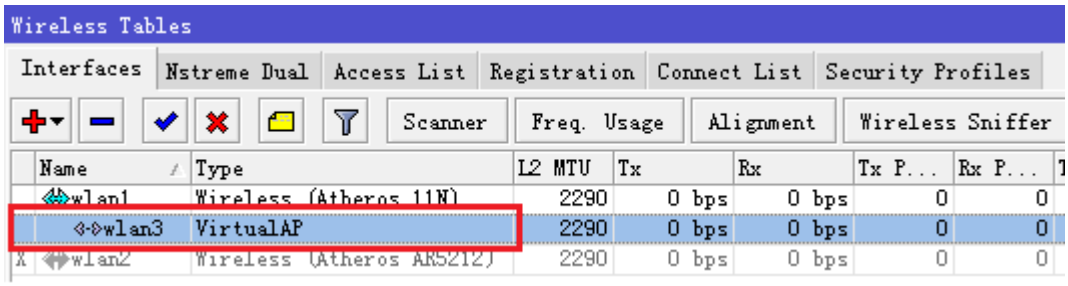
打開虛擬 AP 的 wireless 功能表，可以到基本和物理網卡相同的配置參數，主機 Master-interface 是選擇虛擬 AP 從屬於那個物理網卡



我們定義了 SSID 為 V_wifi，從屬於 wlan1 物理網卡，且我們選擇加密的 security-profile=vip



添加完成後，我們可以看到 wlan3 從屬於 wlan1 網卡下：



這時你可以在終端設備上搜索到 V_wifi 的信號。

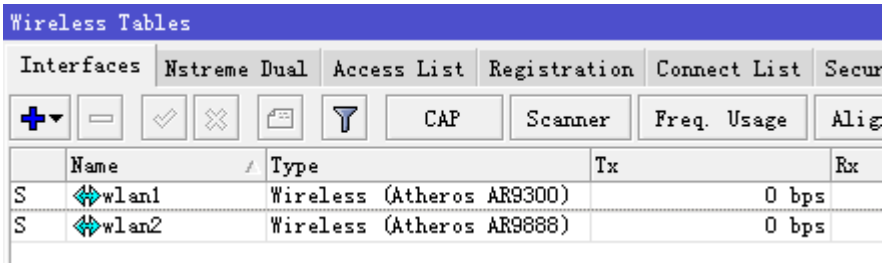
5.6 hAP ac 雙頻合一配置

hAP ac lite、hAP ac 和 hAP ac2（包括 wAP ac 和 cAP ac）三款基於 802.11ac 的家用辦公無線路由器，支援 2.4G 和 5G 雙頻，也就是 802.11bgn 和 802.11ac 兩個協議同時工作。由於 802.11ac 採用 5G 頻率覆蓋範圍有限，一般僅能在視距內傳輸，但能提供更高的頻寬，如果 3×3 80MHz 的 MIMO 可以獲得 1.3G 的頻寬，如果非視距內只能通過 802.11bgn 來彌補。下面是關於 802.11n 和 802.11ac 的技術參數，以及 hAP ac 和 hAP ac lite 在無線 MIMO 的區別

技術規格	802.11n	802.11ac
頻率	2.4G，5G	5G
調製方案	OFDM	OFDM
通道頻寬	20，40MHz	20，40，80MHz
單流額定傳輸率	150Mbps（1×1 40MHz）	433Mbps（1×1 80MHz）
多流額度傳輸率	450Mbps（3×3 40MHz）	1.3Gbps（3×3 80MHz）
hAP ac MIMO	3×3 支持 450Mbps	3×3 支持 1.3Gbps
hAP ac lite MIMO	2×2 支持 300Mbps	1×1 支持 433Mbps

不過作為用戶端同時只能連接一個頻率，要麼 2.4G 的 802.11bgn 或者 5G 的 802.11ac，一般 802.11ac 的路由器會提供 2.4G 和 5G 兩個配置，即兩個無線網路一個 2.4G 或一個 5G，配置不同兩個不同的 SSID。也可以配置所謂的雙頻合一，即使採用雙頻合一相同的 SSID，用戶端也只會連接到一個頻率上，只是會在兩個頻率上根據信號強弱做切換。很多廠商的無線路由器提供了雙頻合一的設置，這樣的設置用戶端也只是在 5G 信號好的時候連接 802.11ac，當 5G 信號變弱後，切換到 2.4G 的 802.11bgn。

在 hAP ac 路由器裡可以看到兩個無線網卡，一個 wlan1 是 802.11bgn，一個 wlan2 是 802.11ac



兩張無線網卡負責不同的協定，那該如何設置雙頻合一，AR9300 負責 802.11bgn，AR9888 負責 802.11ac，我的思路是把兩個無線網卡做 WDS 漫遊方式，即兩個網卡 SSID 相同，無線網卡 mode=ap-bridge，通過橋接創建 rstp 協議的 WDS 無線漫遊。配置如下

1、橋接配置

進入 bridge 創建 bridge1

```
/interface bridge
add name=bridge1 protocol-mode=rstp
```

進入 bridge port 將 wlan1 和 wlan2 加入 bridge1

```
/interface bridge port
add bridge=bridge1 interface=wlan1
add bridge=bridge1 interface=wlan2
```

2、網路配置

配置路由器 bridge1 的 IP 地址，即分給用戶的 IP 地址段

```
/ip address
add address=192.168.88.1/24 interface=bridge1
```

創建 DHCP 服務，配置位址集區：

```
/ip pool
add name=pool1 ranges=192.168.88.2-192.168.88.100
```

配置 DHCP 服務的介面和位址集區

```
/ip dhcp-server
add address-pool=pool1 disabled=no interface=bridge1 name=server1
```

配置 DHCP 服務分配給使用者的閘道和 DNS 伺服器

```
/ip dhcp-server network
add dns-server=192.168.88.1 gateway=192.168.88.1 netmask=24
```

配置 DNS 伺服器 IP 位址和開啟 DNS 本地解析

```
/ip dns
set servers=61.139.2.69 allow-remote-requests=yes
```

啟用 nat 轉換

```
/ip firewall nat
add action=masquerade chain=srcnat
```

3、無線網路配置

配置無線安全密碼，創建 wpa/wpa2 的無線密碼，設置為 1234567890

```
/interface wireless security-profiles
add mode=dynamic-keys authentication-types=wpa-psk,wpa2-psk group-ciphers=tkip,aes-ccm
name=yus unicast-ciphers=tkip,aes-ccm wpa-pre-shared-key=1234567890
wpa2-pre-shared-key=1234567890
```

配置無線網卡，設置兩個無線網卡 SSID 相同取名 yus，設置 wds 模式為 dynamic-mesh，wed-default-bridge 為 bridge1

```
/interface wireless
set [find default-name=wlan1] ssid=yus band=2ghz-b/g/n disabled=no frequency=2422
mode=ap-bridge security-profile=yus wds-default-bridge=bridge1 wds-mode=dynamic-mesh

set [find default-name=wlan2] ssid=yus band=5ghz-a/n/ac channel-width=20/40/80mhz-Ceee
disabled=no frequency=5745 mode=ap-bridge security-profile=yus wds-default-bridge=bridge1
wds-mode=dynamic-mesh
```

以上配置完成後，終端設備會自動連接信號最強的頻率，例如首先當連接上 5G 的 802.11ac，當你移動 5G 信號變弱後，終端設備會自動連接到 2.4G 的 802.11bgn，但如果你要從 802.11bgn 切換到 802.11ac，需要你終端設備去完成，而非路由器決定。無線漫遊的切換都是由終端設備決定的。

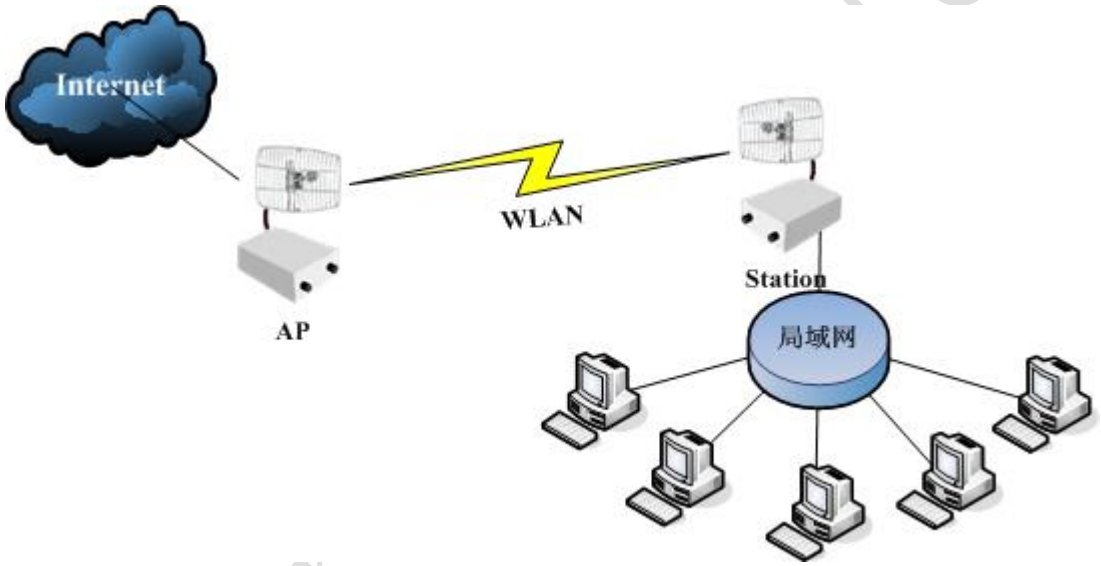
第六章 WLAN 點對點

現在隨著無線 Wlan 技術的發展，頻寬和距離已經得到成功解決，點對點的無線 wlan 傳輸已經非常成熟。通過 wlan 傳輸可以達到上百公里，最高頻寬能到 100M。點對點的無線傳輸能為偏遠的使用者上網或者需要低成本解決網路問題的使用者提供方便快捷和低成本的接入。

6.1 點對點傳輸介紹

在 WLAN 點對點傳輸配置和應用是整個 WLAN 無線網路構建的基礎，即後面的點對多點、中繼傳輸、WDS 和 Mesh 網路都是基於點對點傳輸而變化的，所以在點對點應用中我們需要特別注意。

通過無線點對點的傳輸，將 Internet 資料連接到遠端的局域網內，如圖：



無線點對點傳輸優點：

1、長距離傳輸成本低，如果一個 5 公里的網路接入，通過光纖佈線施工加材料費用成本在 5 萬以上，如果是採用 wlan 的無線接入，如一套“Groove+拋物面天線”的點對點設備和配件不會超過 3 千元。

2、安裝時間短，如果是光線接入 5 公里，施工安裝時間也要好幾天，而 wlan 接入，只需要兩點間無阻擋，安裝點確定好後，雙方架好設備，直接安裝，並調整信號也只需要 1 小時左右。

3、可持續性使用，由於無線安裝方便，當這個點無需無線連接後，我們將設備換到其他需要無線網路的地方繼續使用，而線纜在預埋好後，則很難再取出，即使取出也需要投入施工費用，幾乎沒有可持續使用的條件。

更寬的發射頻率範圍：

在發射頻率方面，通過升級 superchannel 還可以得到更寬的頻率，範圍為：802.11b/g：2.312 – 2.497 (5MHz step)，802.11a： 4.920 – 6.100 (5MHz step)，更寬的頻率能獲得更穩定的傳輸頻率。

無線設備傳輸頻寬情況：

型號	性能	點對點，最大 TCP 傳輸

RB411/A/R	Atheros 300Mhz CPU, 64M 記憶體，1 個百兆 LAN 口，1 個 MiniPCI	40Mbps-50Mbps
RB433	Atheros 300Mhz CPU, 64M 記憶體，3 個百兆 LAN 口，3 個 MiniPCI	40Mbps-50Mbps
RB411AH	Atheros 680Mhz CPU, 64M 記憶體，1 個百兆 LAN 口，1 個 MiniPCI	60Mbps-80Mbps
RB433AH/U	Atheros 680Mhz CPU, 128M 記憶體，3 個百兆 LAN 口，3 個 MiniPCI	60Mbps-80Mbps
RB711	Atheros 400Mhz CPU, 32M/64M 記憶體，1 個百兆 LAN 口，集成 1 個 5GHz 802.11a/n Atheros AR9280	80Mbps-93Mbps
RBSXT-5HnD	Atheros 400Mhz CPU, 64M 記憶體，1 個百兆 LAN 口，集成 1 個 5GHz 802.11a/n Atheros AR9280，1 個 USB	80Mbps-93Mbps
RB800	PowerPC 800MHz 處理器，256M 記憶體，3 個千兆 LAN 口，4 個 MiniPCI，1 個 MiniPCI-e	150Mbps-180Mbps

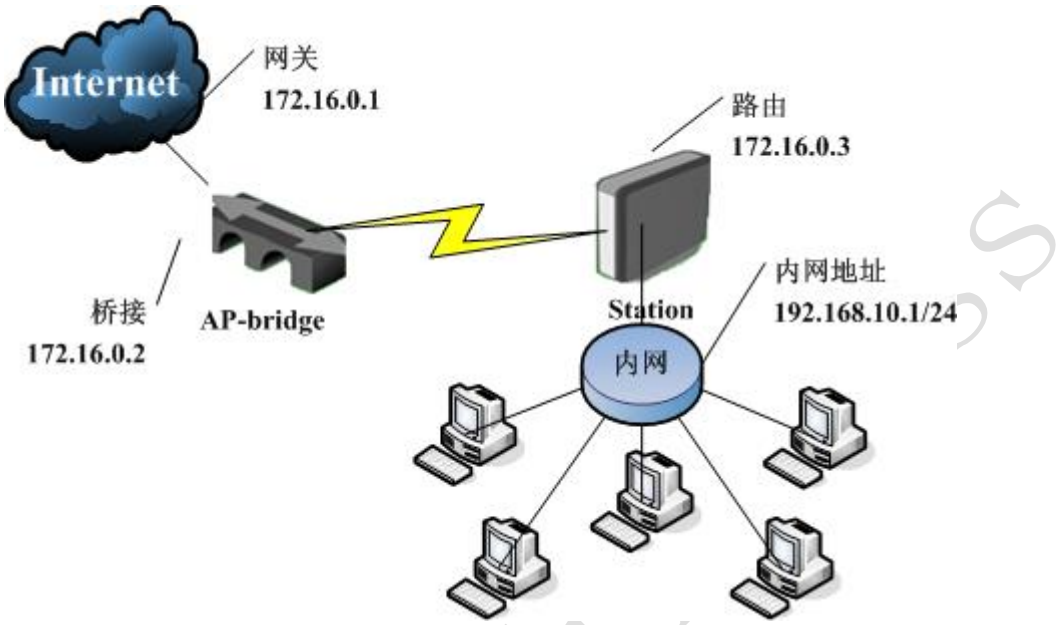
MikroTik 點對點模式的多種方式：

模式	應用	Nstreme 協議	Nv2	性能與提升
AP-Bridge to Station	路由模式，啟用橋接需要配置 EoIP	支持	支持	取決於網卡速率和選擇模式，支援 108M 模式和 11n 的 300M
AP-Bridge to Station-WDS	常用於橋接模式，自動添加到橋接設置中，也可用于路由	支持	支持	取決於網卡速率和選擇模式，支援 108M 模式和 11n 的 300M
AP-Bridge to AP-Bridge	支援橋接和路由模式，橋接模式自動添加到橋接設置中，同樣支援 MESH 和 WDS 模式	不支持	不支持	取決於網卡速率和選擇模式，支援 108M 模式
Bridge to Bridge	橋接模式	不支持	不支持	取決於網卡速率和選擇模式，支援 108M 模式
Bridge to station-bridge	為 Nv2 協議開發，替代 bridge to bridge	不支持	支持	取決於網卡速率和選擇模式，支援 108M 模式和 11n 的 300M
bonding	橋接模式	支持	不支持	取決於網卡速率和選擇模式，在 bonding 模式下的效果 1+1〉2，非常消耗 CPU 資源
Nstreme-dual	橋接和路由模式，雙向一個發送，一個接收	支持	不支持	雙向傳輸，取決於網卡速率和選擇模式，支援 108M 模式

以上清單顯示了多種無線模式的連接應用方式，Nstreme 協定是 MikroTik 長距離傳輸和獲取高頻寬下採用的，這樣的模式可以讓你的無線網路頻寬得到有效的提升，5.0 後支援 Nv2 協定，能有效提升 11n 的頻寬，最高可以達到 180Mbps。

6.2 AP-Bridge to Station 路由模式

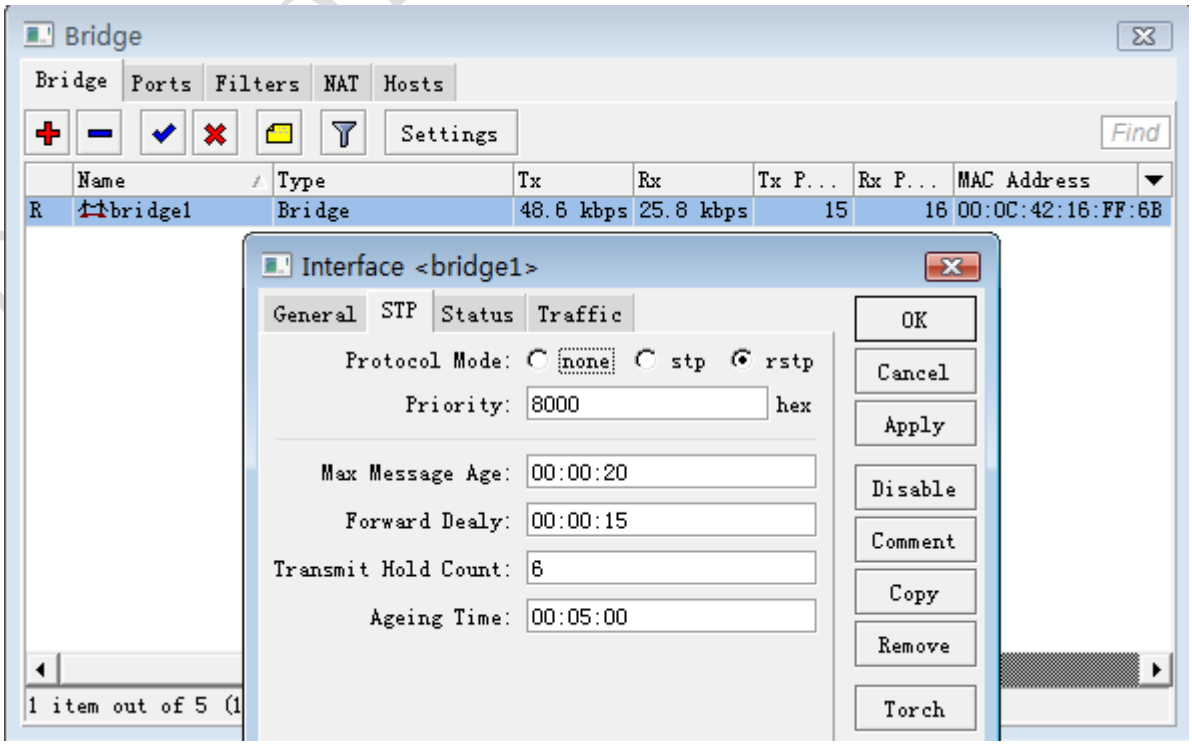
AP-bridge 和 Station 模式傳輸常用于路由模式，AP-bridge 設備仍然設置為橋模式，不啟用 WDS。Station 設備則配置為路由模式。 如下圖，外網的 Internet 通過 AP-bridge 設置的橋（IP 位址為 172.16.0.1）連接到 Station 端的路由（無線介面 IP 位址為 172.168.0.2，內網的以太網介面 IP 為 192.168.10.1，連接內網的 192.168.10.0/24 網路）

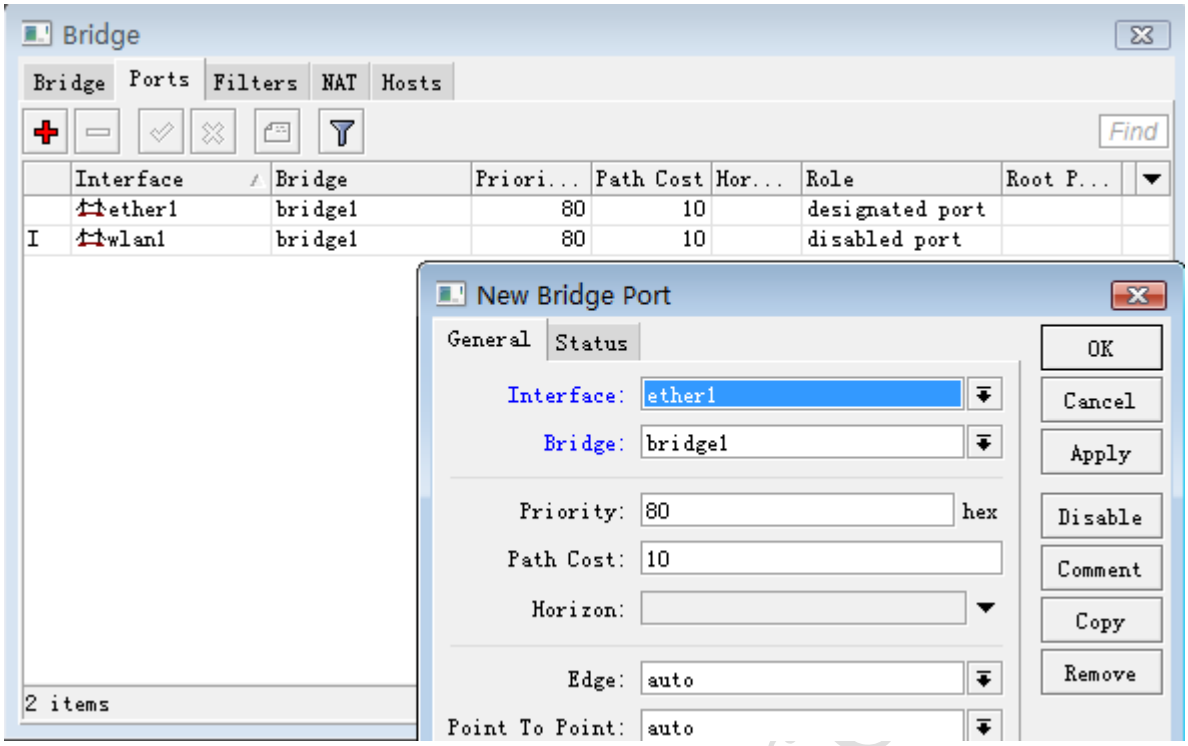


AP-bridge 與 Station 模式配置步驟：

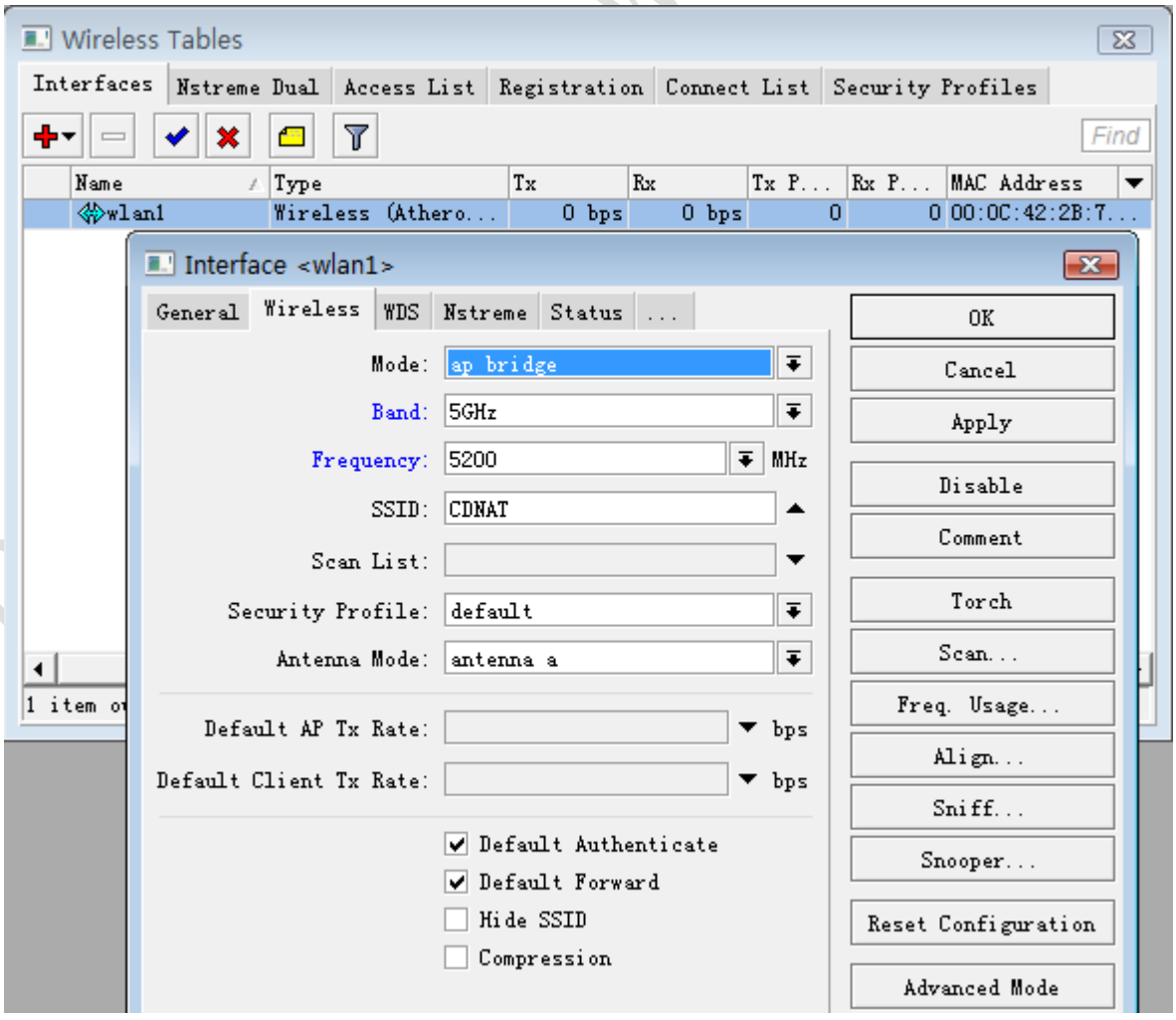
1、 在 AP-bridge 端配置橋接，將 Wlan1 和 ether1 添加如 bridge 中；
2、 配置 AP-bridge 無線參數，並在 ip address 中配置 bridge 的 IP 位址；
3、 在 Station 端配置 Wlan1 的無線 IP 位址，配置 ether1 的內網 IP 地址；
4、 配置 Station 端的無線參數，並測試無線網路與內部網路是否連接。

步驟 1：設置 AP-bridge 端的橋接，進入 birdge 添加一個橋，並設置 Port 的介面：

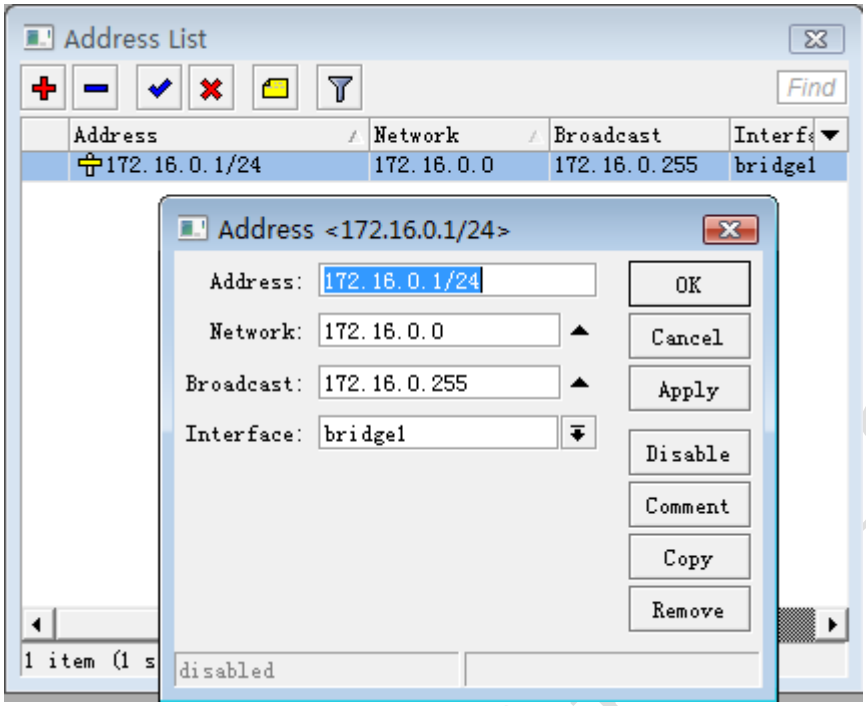




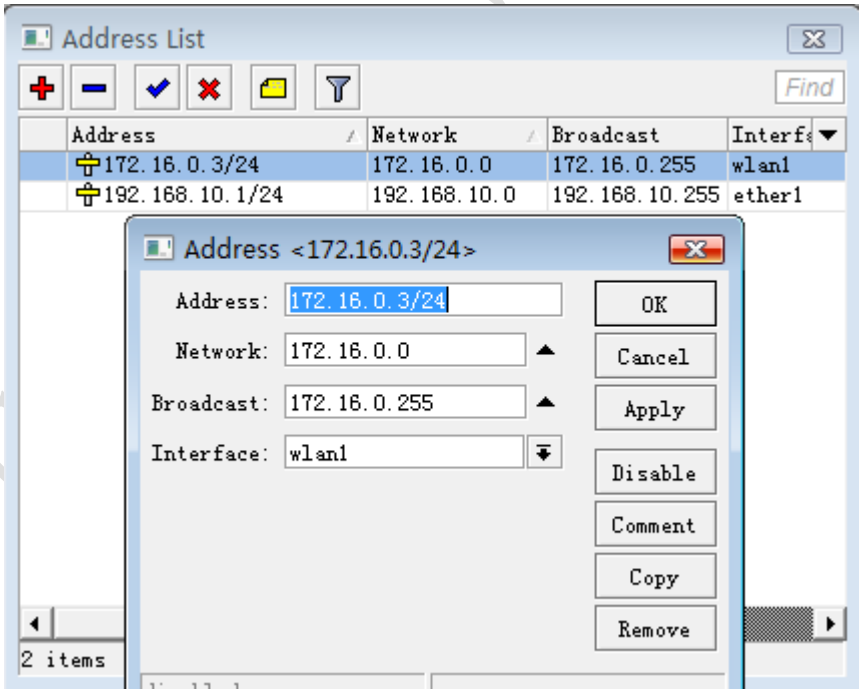
步驟 2 進入 wireless 配置 wlan1 的參數，配置 mode=ap-bridge，Band=5GHz，Frequency=5200，SSID=CDNAT，其他參數默認：



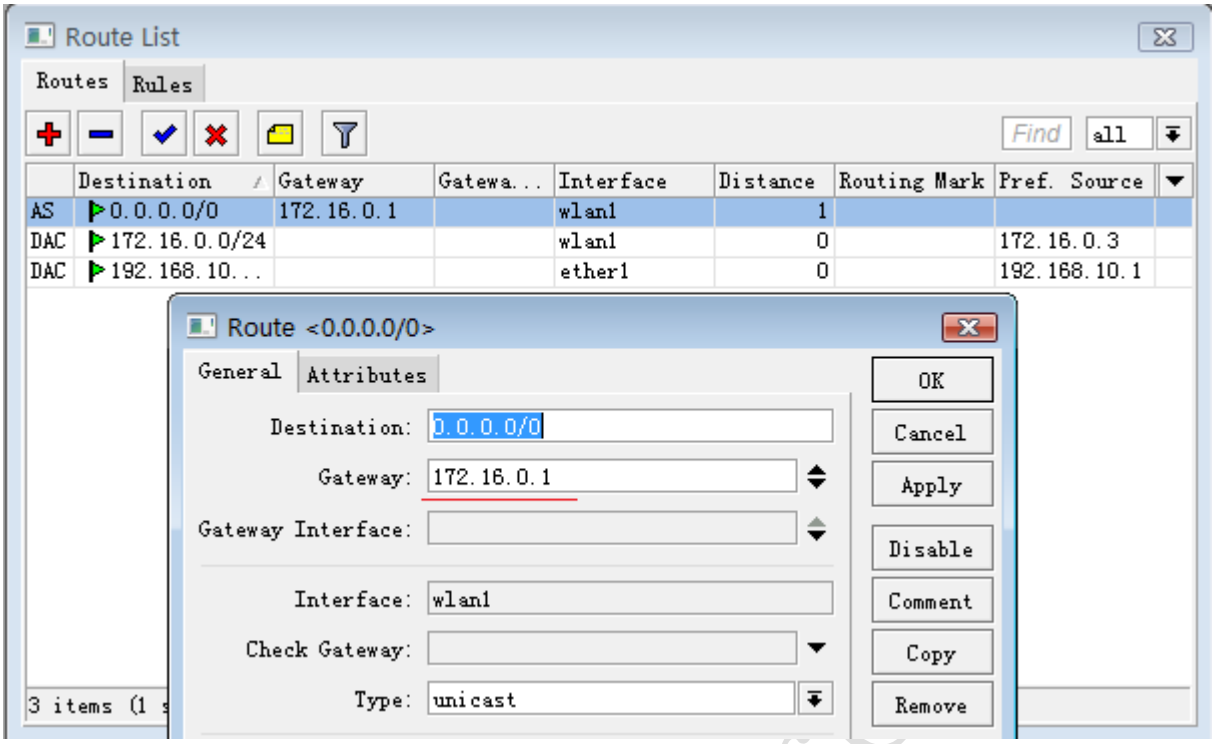
之後進入在 ip address 中添加 IP 位址，配置 IP 位址為 172.16.0.2



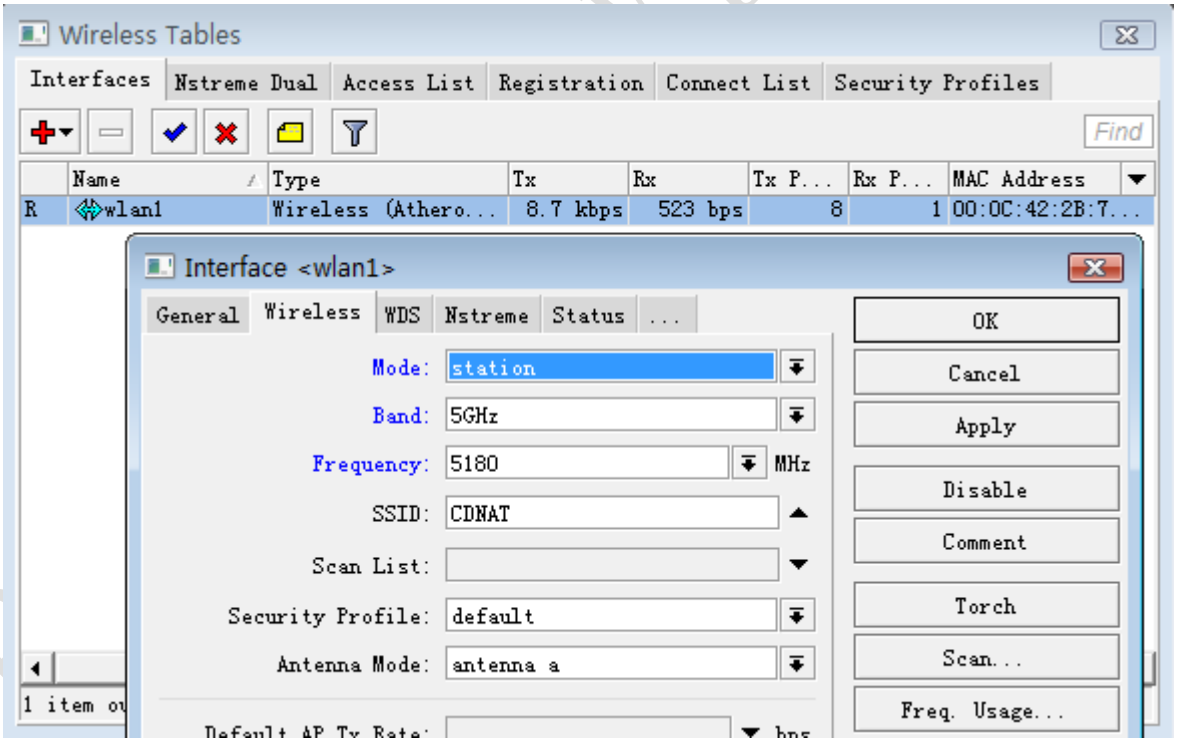
步驟 3：配置 Station 端的 wlan1 和 ether1 的 IP 地址，分別為 172.16.0.3 和 192.168.10.1，並配置閘道位址 172.16.0.1。



在 ip route 中配置閘道位址：

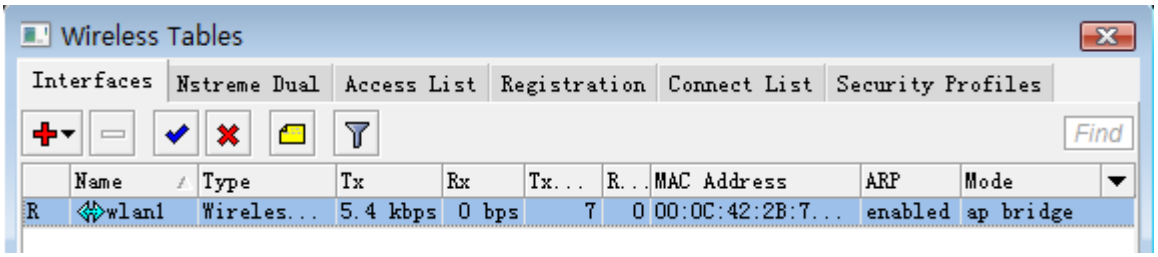


步驟 4：配置 Station 端的無線參數，設置 mode=station，Band=5G，SSID=CDNAT ：

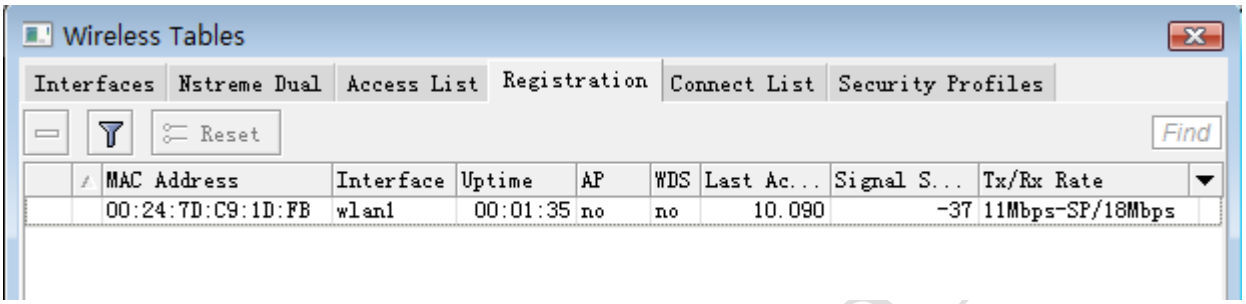


*注：*當 Mode 設置為 station 或者 station-wds 情況下，Band 和 SSID 與 AP 配置相同，Frequency 會自動適應 AP 的頻率參數。

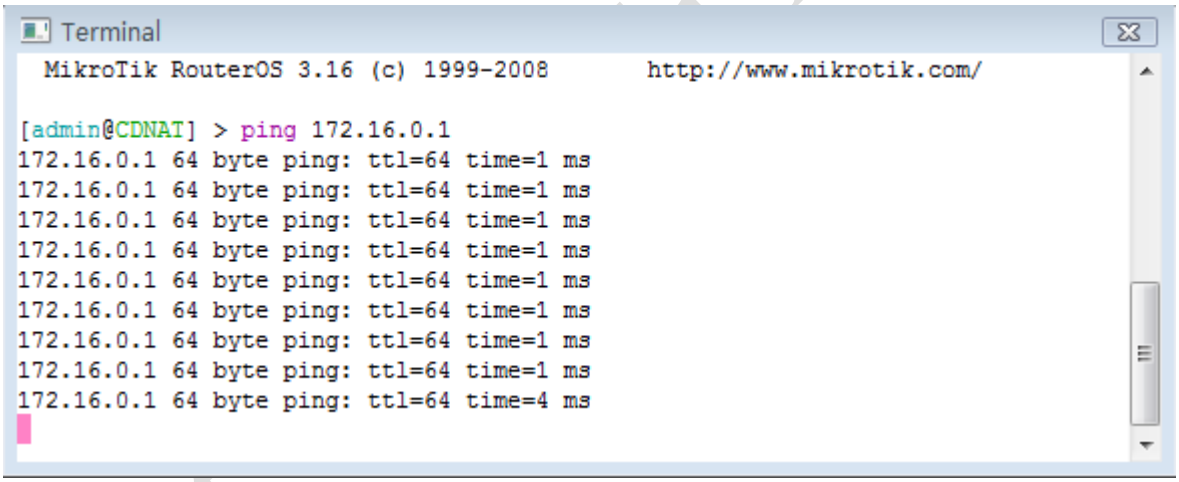
這樣 AP-Bridge to Station 配置完成，，當連接後可以查看 wlan1 的無線狀態會在專案最前面顯示“R”運行：



無線註冊信號與參數：

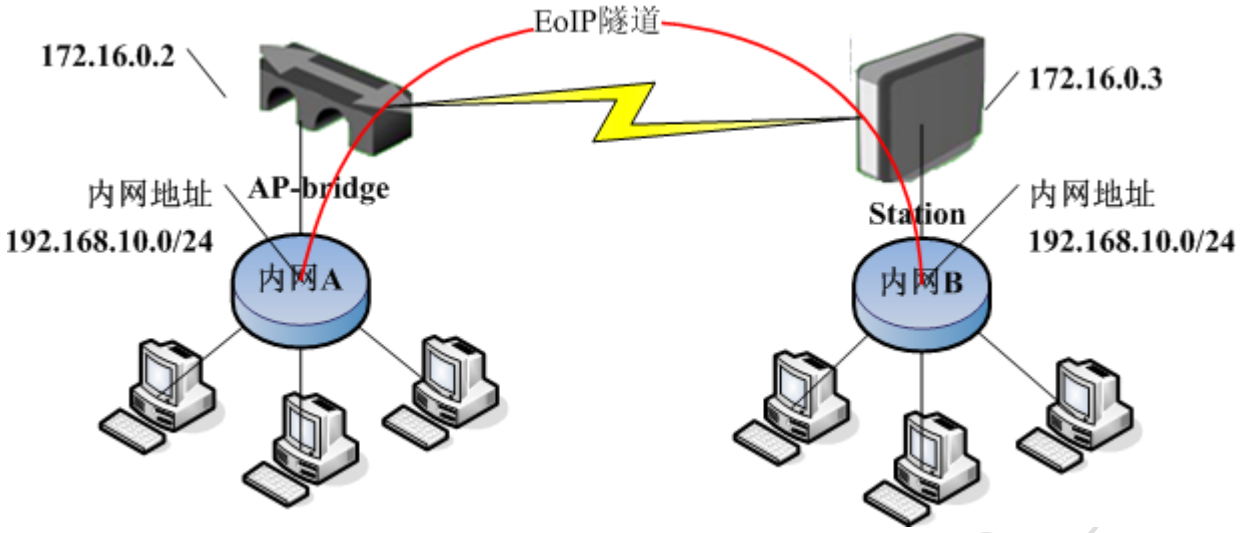


最後通過 Terminal 在終端使用 ping 命令檢測 172.16.0.1，是否連接正常：



6.3 AP-Bridge to Station 的 EoIP 橋接模式

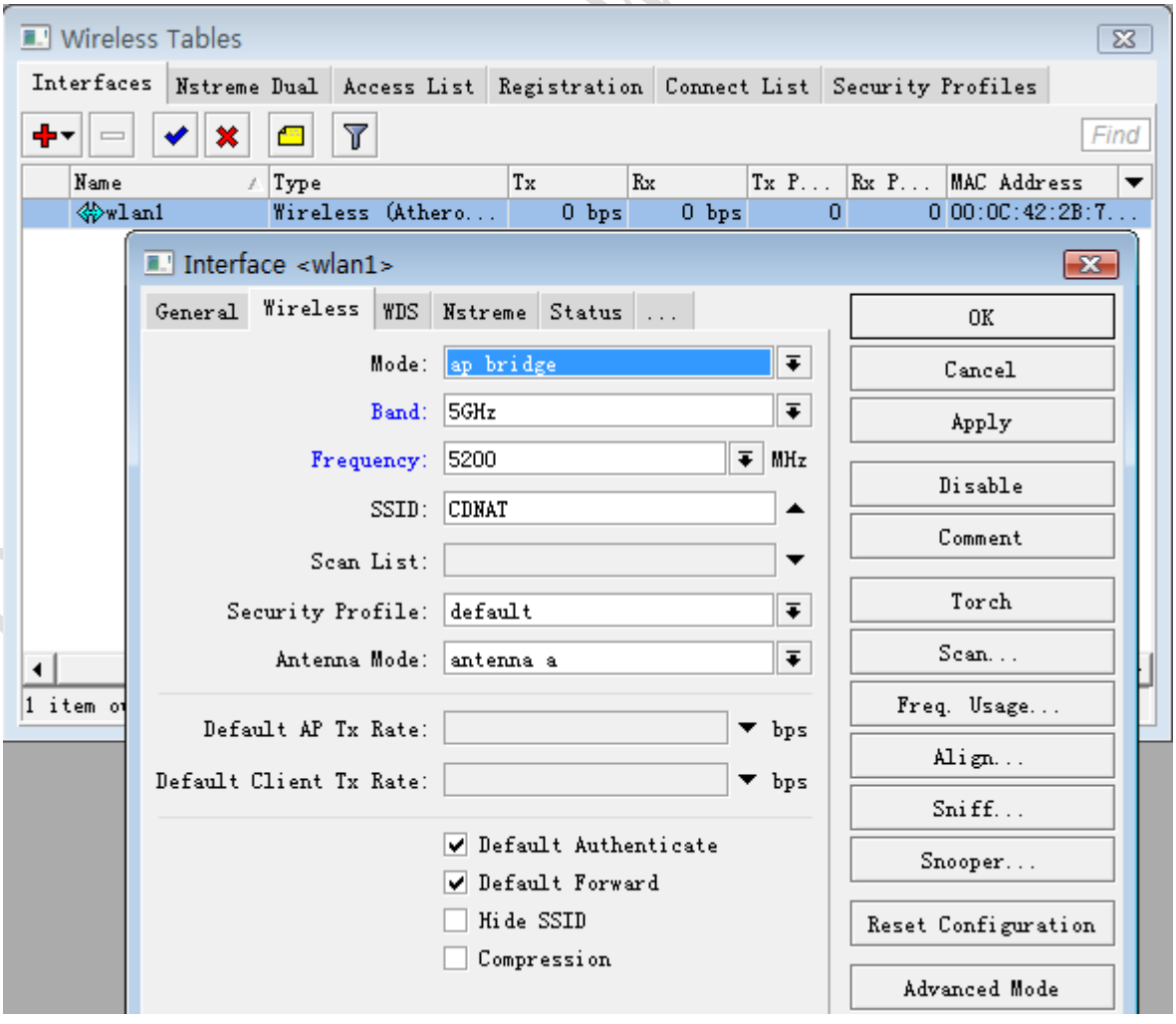
基於 AP-Bridge to Station 的 EoIP（基於 IP 傳輸的以太網協定）橋接，主要是早期為解決 RouterOS 不能實現橋接問題而設置的，通過 EoIP 我們可以將採用路由模式的 AP 的設備，在 IP 位址建立的 EoIP 隧道中透傳二層資料。如下圖：

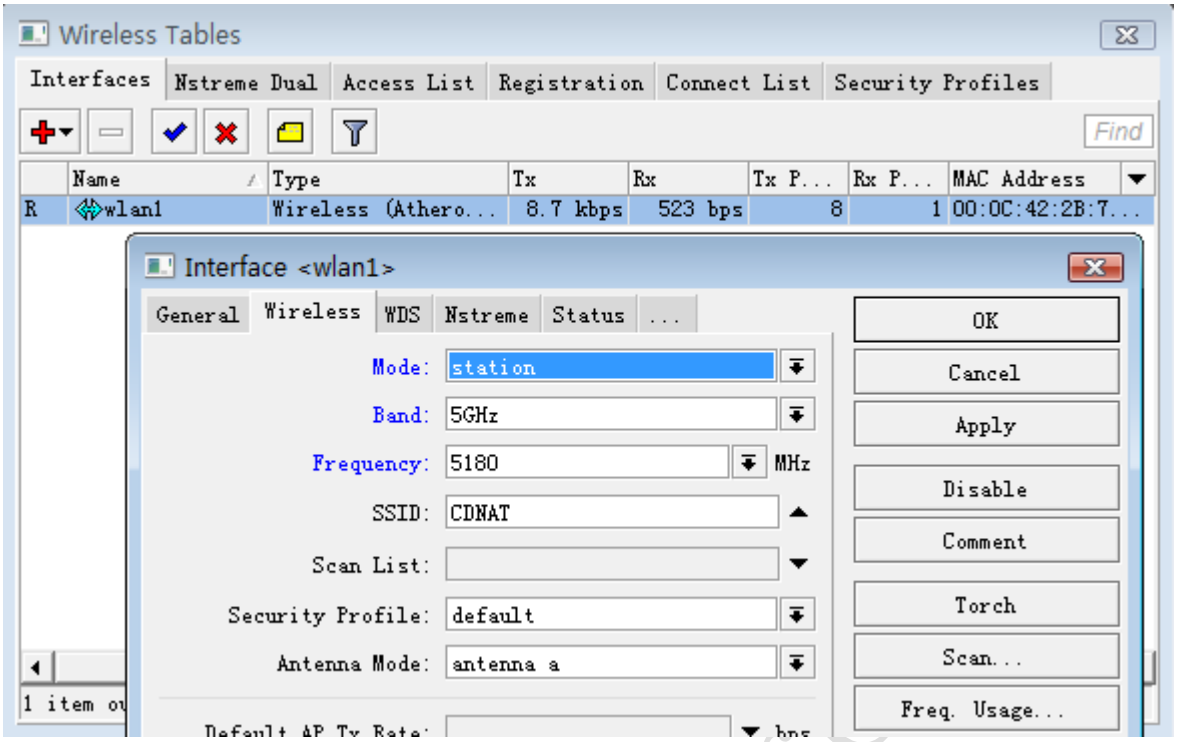


這裡我們以之前的 AP-Bridge to Station 事例為基礎，建立 EoIP 隧道：

1、 完成 AP-Bridge to Station 的無線連接和 IP 位址配置
2、 配置 AP-bridge 端的 EoIP 隧道參數，並配置 bridge 參數
3、 配置 station 端的 EoIP 隧道參數，並配置 bridge 參數

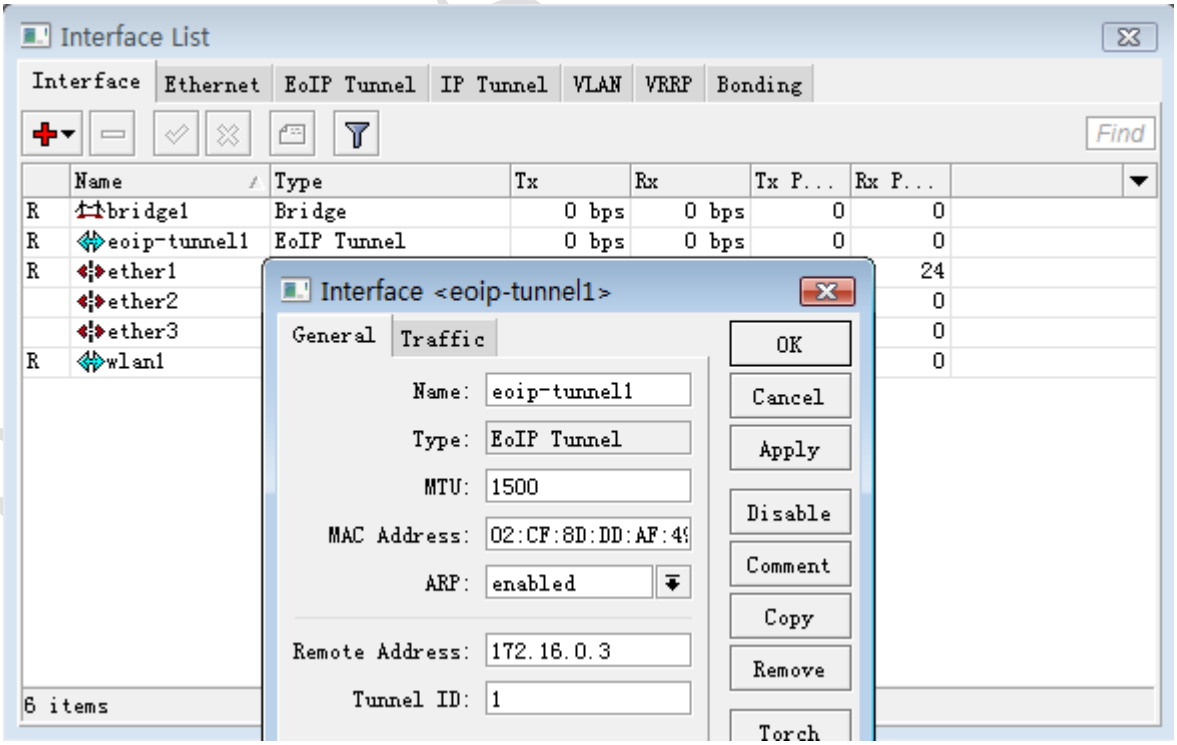
步驟 1：配置 AP-bridge 和 Station 的無線連接，這裡的 AP-bridge 和 Station 配置和之前的相同



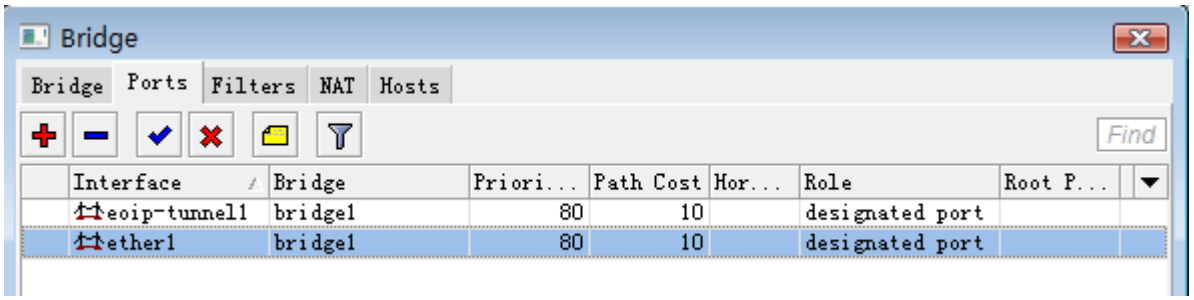


同樣在 ip address 給 AP-bridge 的 wlan1 設置 172.16.0.2 的 IP 位址，Station 的 wlan1 設置 172.16.0.3 的 IP 位址。

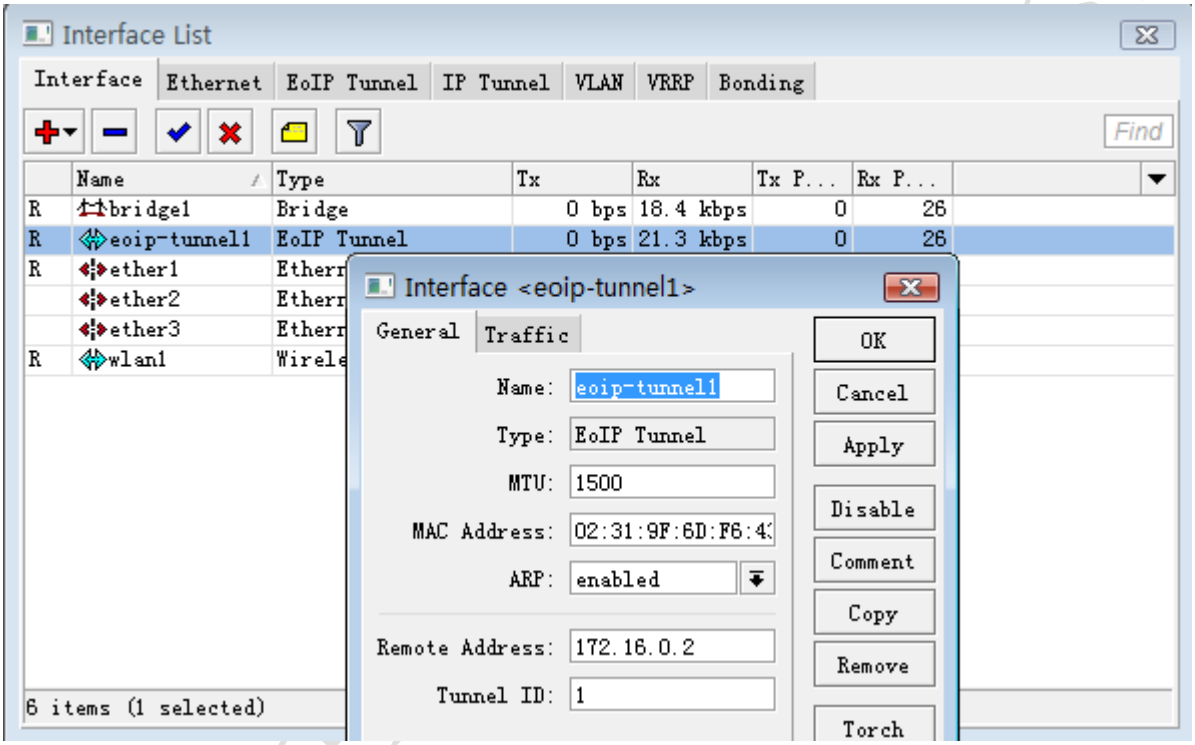
步驟 2：配置 AP-bridge 的 EoIP 隧道，填寫對方的 Station 設備的 IP 位址 172.16.0.3，並設置相同的 Tunnel ID，這裡我們設置為“1”



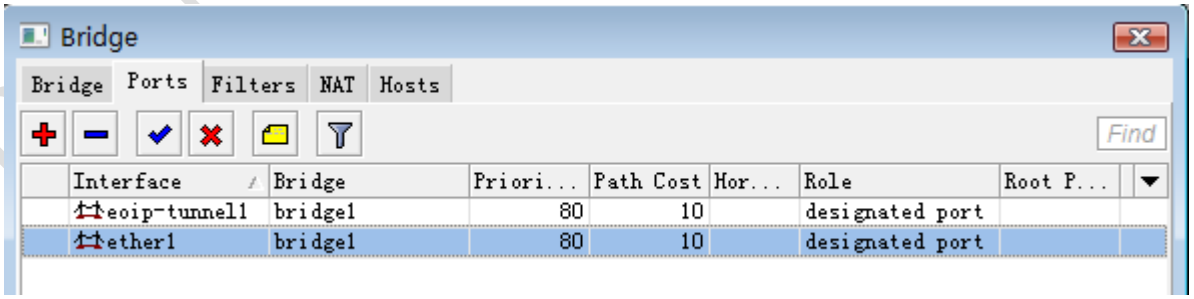
這裡 bridge 這是與之前的事例有點不同，我們只需要將 AP-bridge 端的 ether1 和設置好的 eoip-tunnel1 添加進入橋中



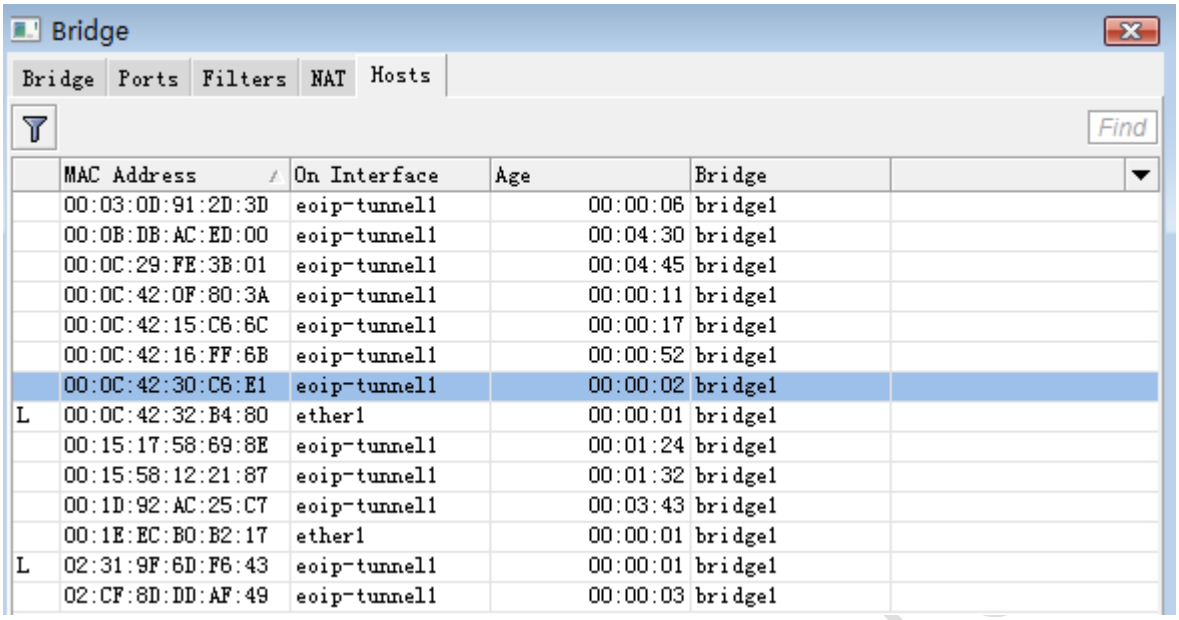
步驟 3：配置 Station 的 EoIP 隧道，填寫對方的 AP-bridge 設備的 IP 位址 172.16.0.2，並設置相同的 Tunnel ID，這裡我們設置為“1”



同樣在 bridge 中添加 ether1 和 eoip-tunnel1 到 bridge1 裡

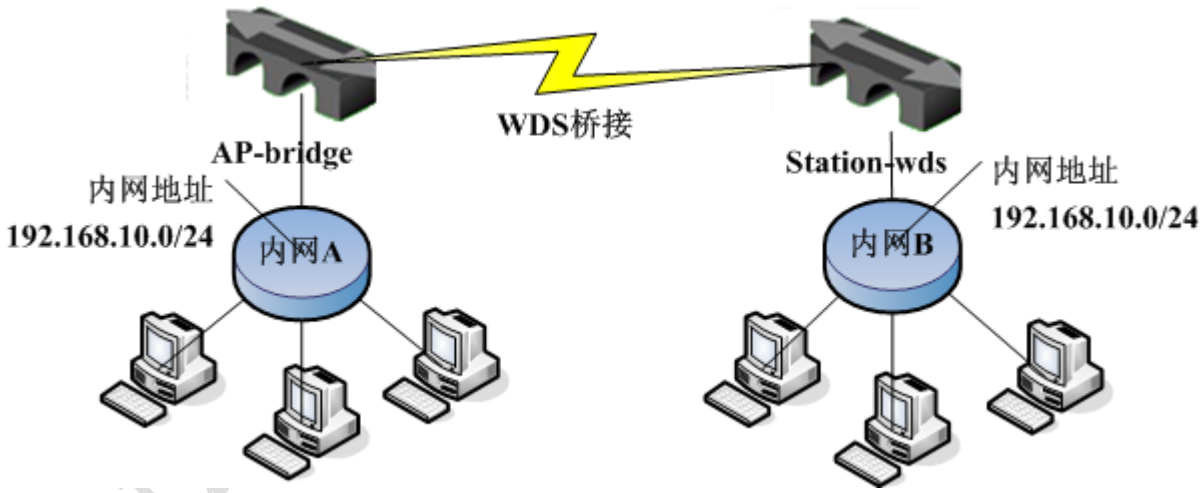


通過查看 bridge 的 Hosts 列表，可以看到 brdige 學習到在 on-interface 項目中有多個 eoip-tunnel1 的 MAC 位址，最前方 L 標示的是 Local 本地的介面。



6.4 AP-Bridge to Station-WDS 橋接模式

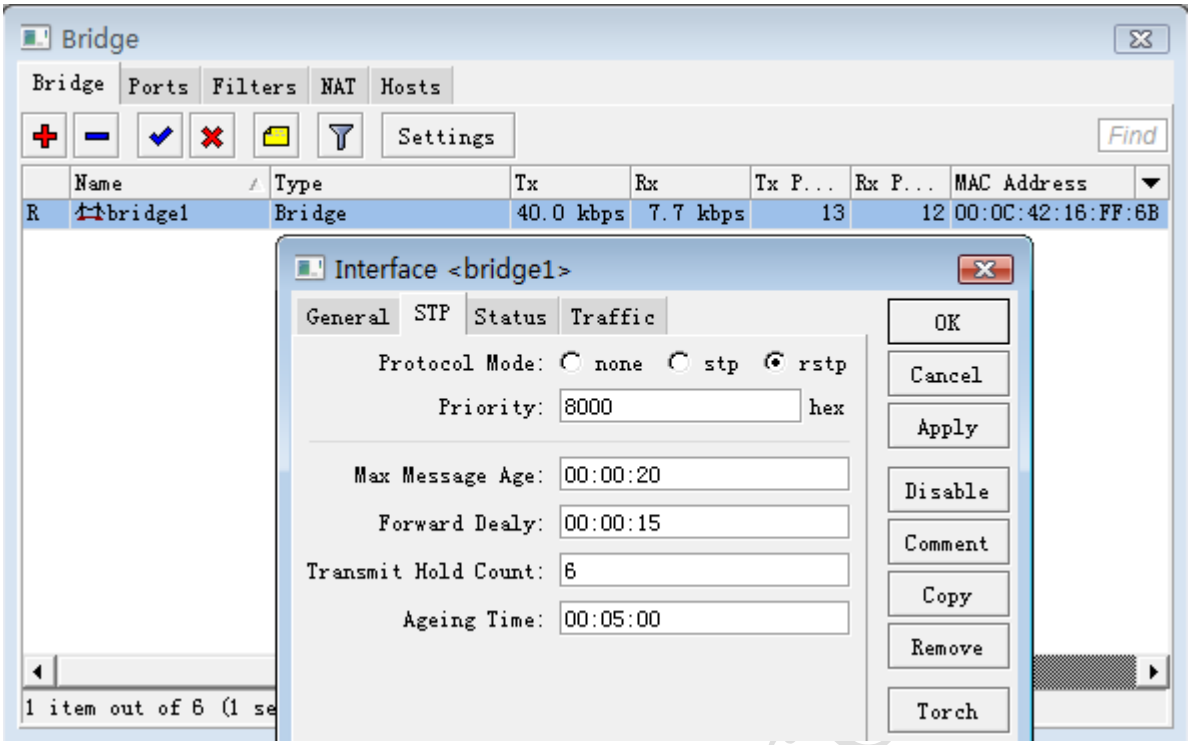
點對點方最常見的的就是 WDS 橋接模式，我們可以採用 ap-bridge 或者 birdge 方式，在這裡我們推薦使用 ap-bridge 與 station-wds 的橋接方式，如下圖：



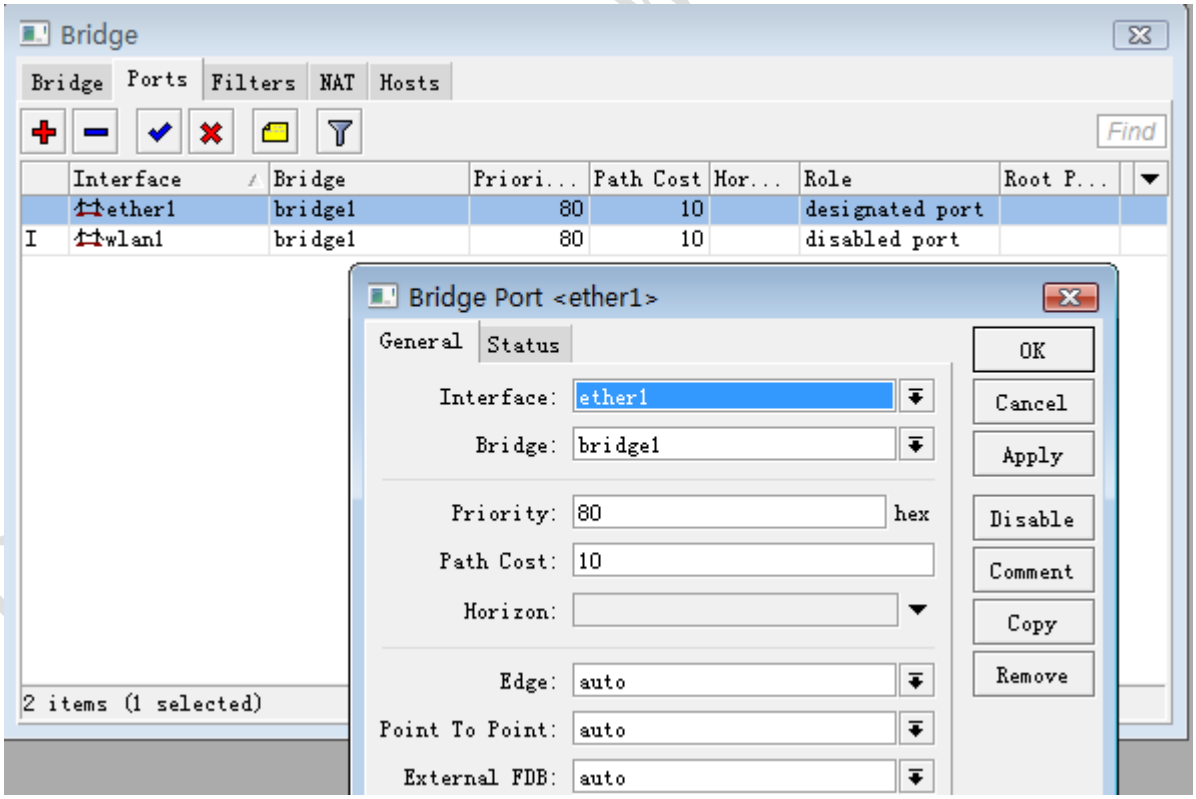
設置在 ap-bridge 和 station-wds 模式的我們分以下步驟：

1、 在 ap-bridge 和 station-wds 中添加 bridge，定義 bridge 的介面，並分配管理的 IP 位址
2、 配置 ap-bridge 和 station-wds 的無線參數
3、 檢查橋接連接情況

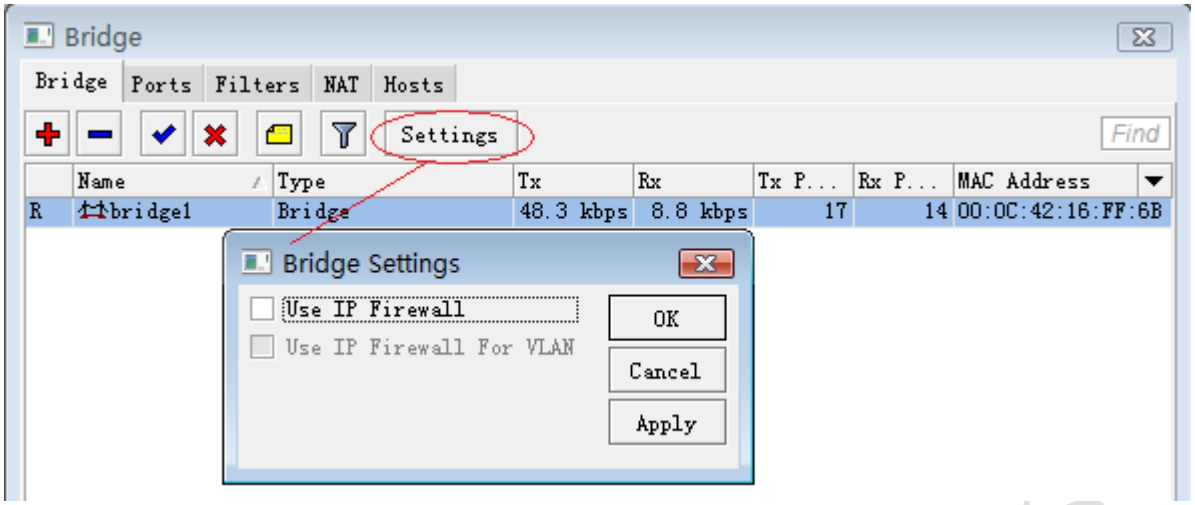
步驟 1：進入 bridge 添加 bridge1 的橋接，通常情況下我會開啟 rstp 協議，啟用生成樹協定：



添加 ether1 和 wlan1 到 brdige1 橋接中，這裡在 interface 中分別添加 ether1 和 wlan1 進入 bridge1 中。這樣 ether1 和 wlan1 就實現了橋接功能，能實現資料二層的透明傳輸：



注： 在 RouterOS3.0 的 bridge 中增加了一個設置選項，是否選擇 ip firewall 過濾，如果不使用 ip firewall 過濾路由器的橋接轉發速度將提升性能，但如果你要求對無線傳輸過程中的 IP 資料進行過濾處理，那就需要開啟 use-ip-firewall 功能：



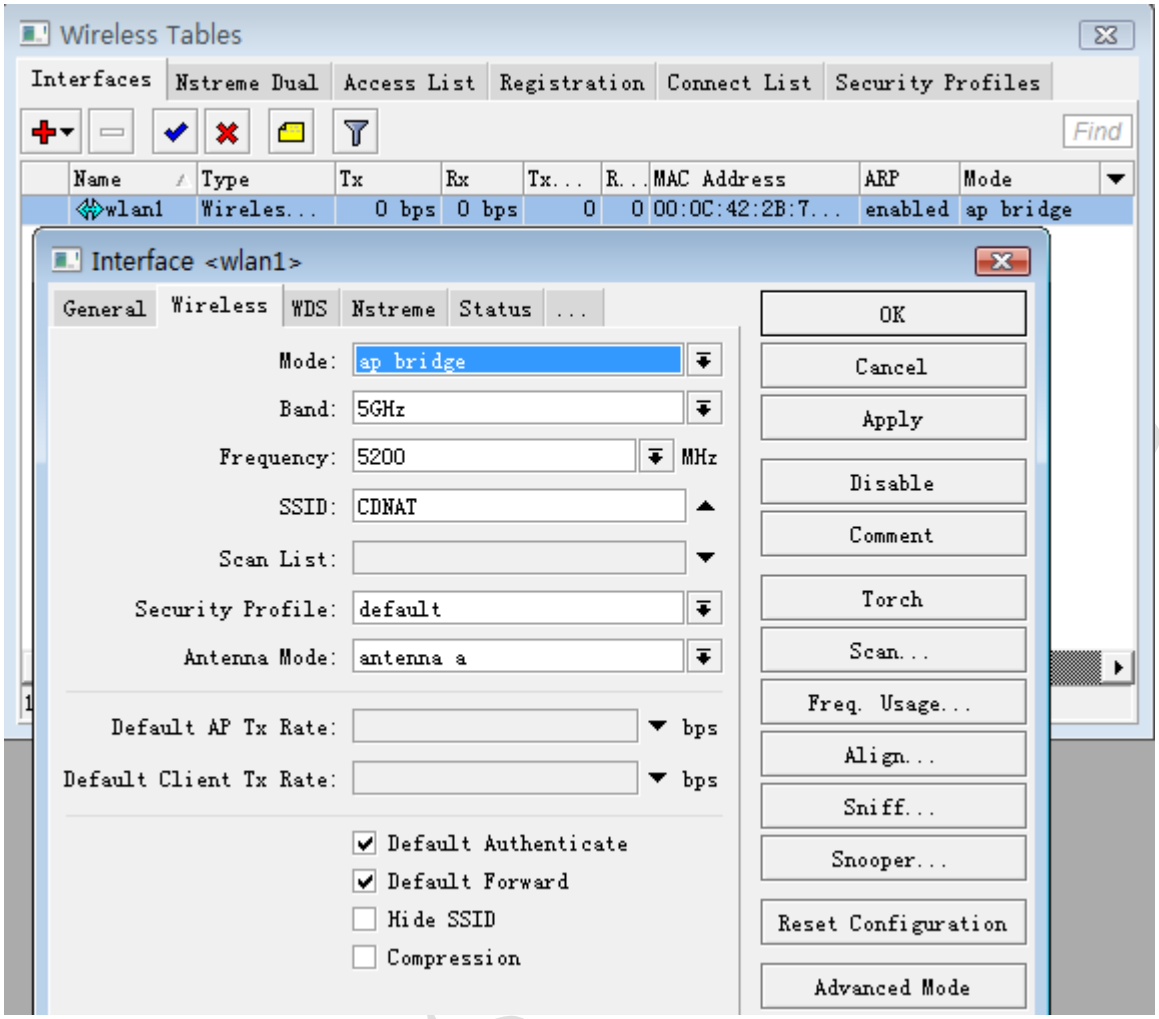
注：以上配置操作適用於 ap-bridge 和 station-wds 設備

設置完橋接後我們進入 ip address 給 **ap-bridge 和 station-wds** 的 bridge 配置一個 IP 位址 192.168.10.1/24 和 192.168.10.2/24，用於管理設備和監測用。這樣 wlan1 口和 ether1 都能分配到這個位址。命令如下：

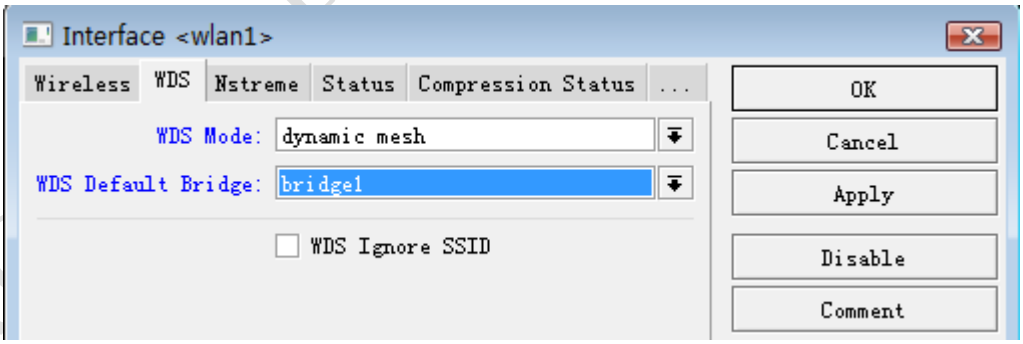
```
ap-bridge 設備
/ip address add address=192.168.10.1/24 interface=bridge1

station-wds 設備
/ip address add address=192.168.10.2/24 interface=bridge1
```

步驟 2：橋接和 IP 位址設置好後，現在配置 ap-bridge 和 station-wds 的無線參數。

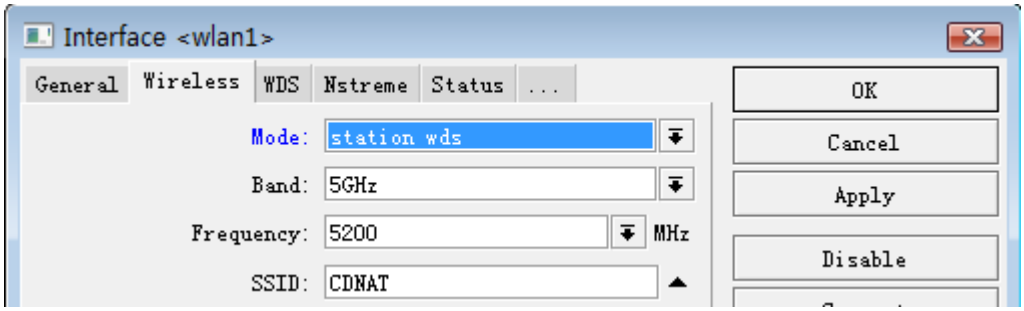
設置 ap-bridge 的無線，這裡 mode=ap-bridge，band=5G，frequency=5200，SSID=CDNAT



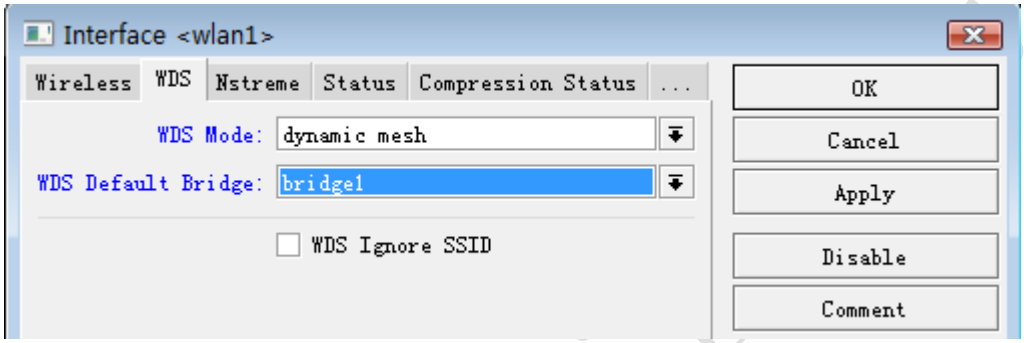
配置 ap-bridge 的 WDS 模式，配置參數 wds-mode=dynamic-mesh（動態方式），wds-default Bridge=bridge1（將連接無線添加到 bridge 中）



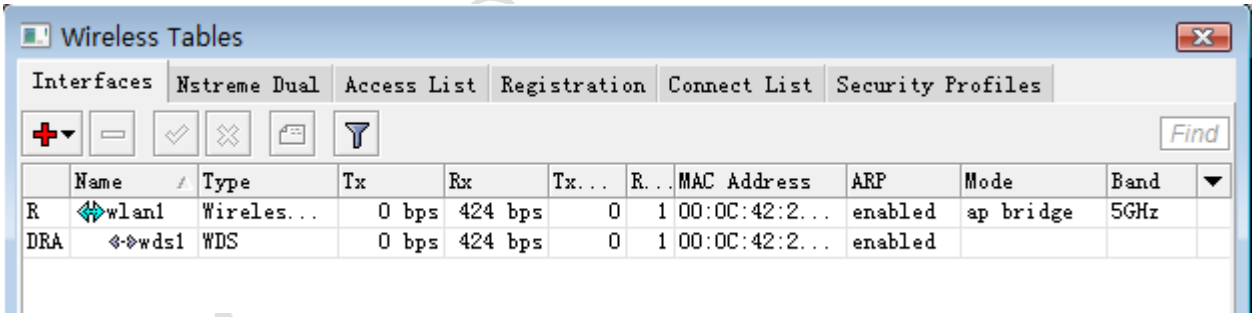
配置 station-wds 端的設置只需要將 Mode=station-wds band=5G，SSID=CDNAT，不需要設置 Frequency 參數，station-wds 在匹配 Band 和 SSID 後會自動搜索：



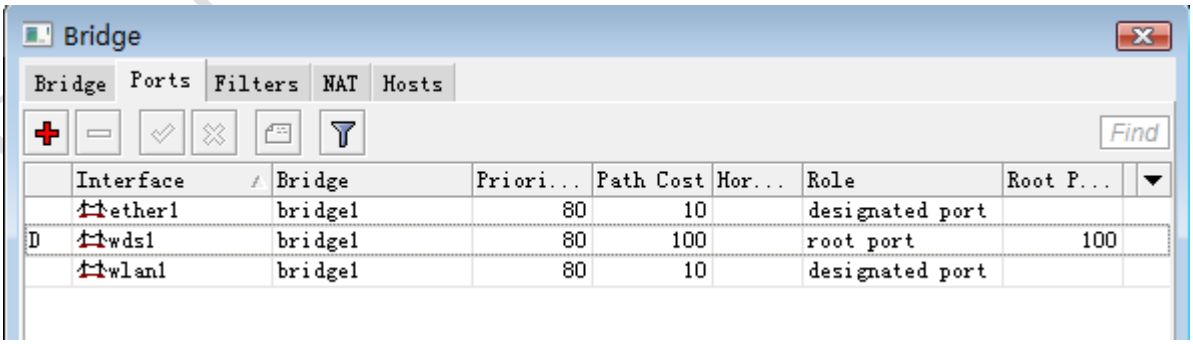
在 station-wds 模式下與 ap-bridge 的 WDS 參數配置相同



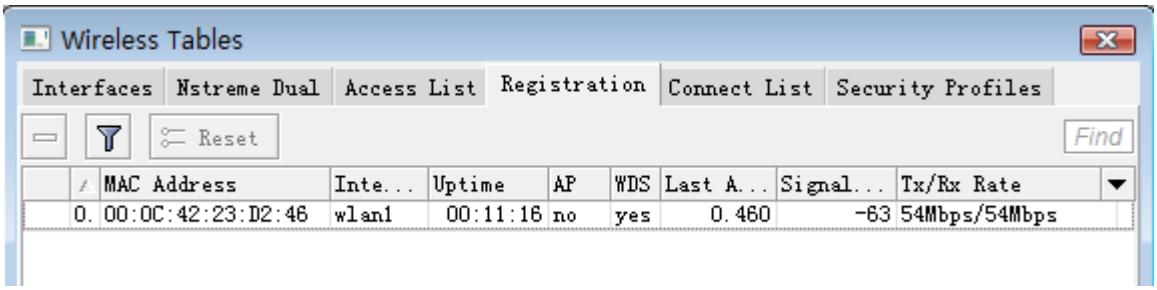
步驟 3：當配置完成後，我們可以通過在 ap-brdige 端的設備查看是否連接，如果正常連接後 ap-bridge 端的 Wireless Tables 下會在 wlan1 前現時“R”，並增加一個 wds1 的無線介面。



在 ap-bridge 下的 Bridge 中可以看到，WDS 模式自動將 wds1 介面添加到 Port 中：



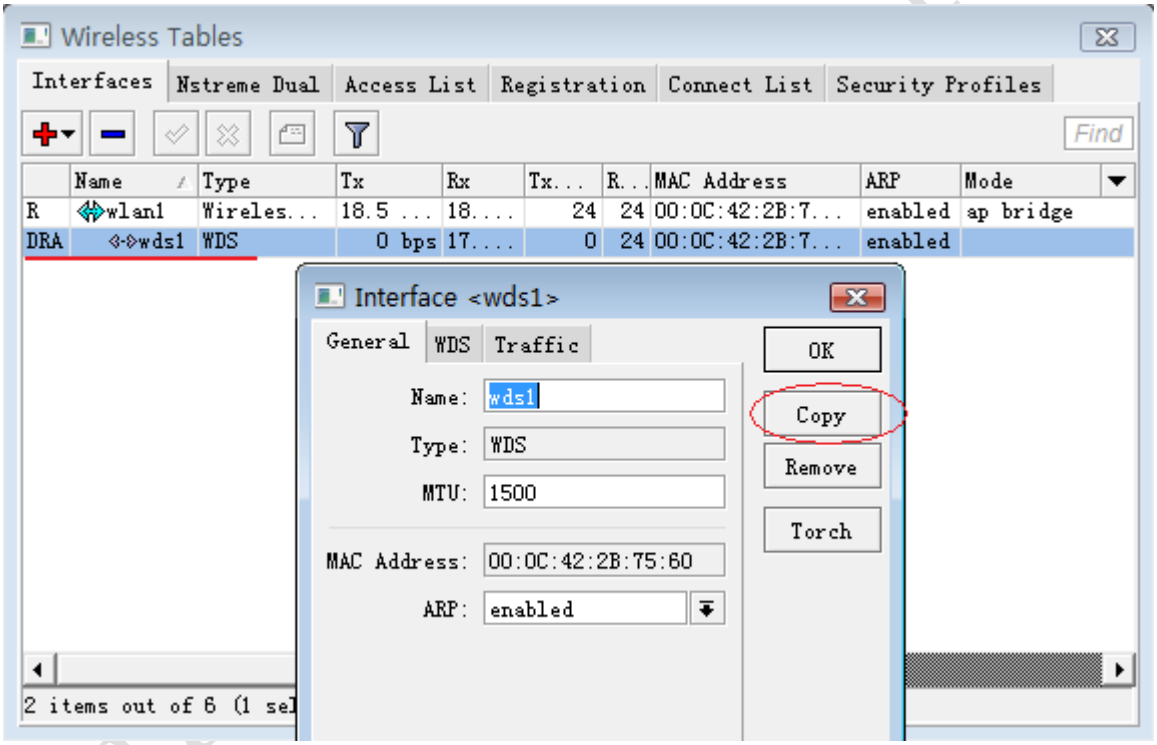
我們可以通過在無線註冊資訊清單中查看信號強度：



這裡顯示的是-63，信號能連接使用的最低值在“-88 到-90”，數位越接近正數“1”信號越強。Station-wds 端在連接後會自動適用 ap-bridge 的參數，並正常通信。

6.5 靜態的 WDS 模式連接

設置靜態的 WDS 連接可以避免其他為允許的無線連接進入我們的網路，保證網路不受到入侵，配置靜態



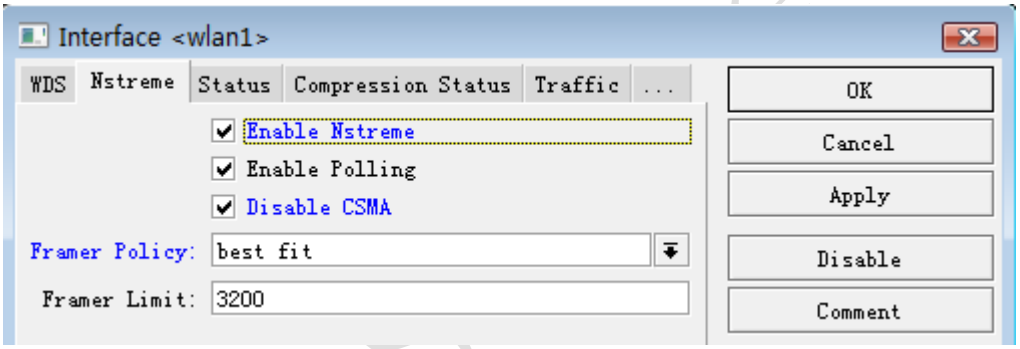
第七章 MikroTik 特有協定與應用

7.1 Nstreme 協議

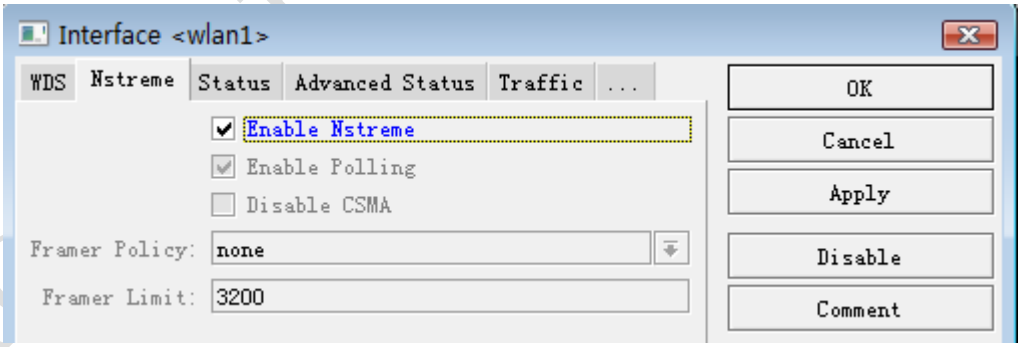
MikroTik 獨有的 Nstreme 協議用於長距離的無線傳輸、增加頻寬和提高網路品質的作用。 Nstreme 支持 **ap-bridge/bridge** to **station/station-wds**，不支持 ap-bridge to ap-bridge，所以通常 Nstreme 被用於點對點傳輸和點對多點傳輸，不能用於漫遊組網。啟用 Nstreme 的操作只需要進入無線網卡配置的 Nstreme 選項。

在我們做點對點傳輸的時候，可以啟用 Nsteme 協定，提高網路安全和傳輸品質。這裡我們基於點對點的 ap-bridge 和 station/station-wds 模式是配置為例，啟用 Nsteme 協定。

ap-bridge 的配置，啟用 Nsteme 協定，禁止 CSMA 協定，啟用 Polling 輪詢方式，並選擇幀策略為 best fit 的幀傳輸，將幀封裝為 3200，最大可以到 4000：



Station 和 station-wds 的配置，只需要啟用即可，其他參數會自我調整 AP 端：



Nstreme 協議必須雙方同時啟用，才能正常連接，所以配置 Nstreme 時一定要注意。

7.2 Nstreme Version 2 協議（NV2）

Nv2 協定由 MikroTik 獨立基於 Atheros 802.11 無線晶片開發無線通訊技術，Nv2 是基於 TDMA (Time Division Multiple Access)介質訪問技術替換 CSMA (Carrier Sense Multiple Access) 介質訪問技術，用於普通的 802.11 設備。

TDMA 介質訪問技術解決了隱藏節點問題，提高了媒體利用率，從而提高輸送量和降低延遲，特別是在點對多點網路中。Nv2 支援 Atheros 802.11n 晶片，而老的 802.11a/b/g 晶片，從 AR5212 開始支持，不支持 AR5211 和 AR5210 晶片。

介質訪問在 Nv2 網路中，是由 Nv2 AP 控制，Nv2 AP 將時間劃分為固定大小的“週期”，這些“週期”根據 AP 和用戶端的佇列情況，動態劃分為下行(從 AP 發送到用戶端的資料)和上行(從用戶端發送到 AP 的資料)部分。上行時間根據連接的客戶機對頻寬的需求進一步劃分。在每個週期開始時，AP 廣播計畫告訴用戶端他們應該在什麼時候傳輸以及他們可以使用的時間。

為了允許新用戶端連接，Nv2 AP 定期為“未指定的”用戶端分配上行時間——然後新用戶端使用這個時間間隔啟動對 AP 的註冊。然後 AP 估計與用戶端之間的傳輸延遲，並開始定期為該客戶機調度上行時間，以完成註冊並從用戶端接收資料。

Nv2 實現了基於每個用戶端的動態速率選擇和資料傳輸的 ARQ。這支援跨 Nv2 鏈路的可靠通信。

對於 QoS, Nv2 使用內置的缺省 QoS 調度程式實現了定義變數的優先順序佇列，該調度程式可以與基於防火牆 mangle 或使用 VLAN 優先順序，以及 MPLS EXP 參數在網路上傳輸的優先順序資訊

Nv2 vs 802.11 區別

- **介質訪問由 AP 預先分配** - 這樣消除了隱藏的節點問題，並允許實現集中的媒體訪問策略，AP 規劃每個用戶端使用的時間，並可以根據某些策略為用戶端分配時間，而不是每個設備相互競爭媒體訪問
- **減少傳輸延遲開銷**- 在 Nv2 中沒有每幀 ACK 請求，這樣能有效的提示輸送量，特別是在長距離鏈路上，資料幀和跟隨 ACK 幀傳輸延遲會顯著降低在介質中的使用率。
- 減少每幀開銷- Nv2 實現幀聚合和分段發送，以最大限度地分配介質使用率，並減少每幀的開銷。

Nv2 vs Nstreme 區別

- **減少輪詢開銷** - Nv2 協定不再是輪詢每個客戶的方式， Nv2 AP 廣播上行調度，將時間分配給多個用戶端，這樣被稱為“組輪詢”- 不會浪費輪詢每個用戶端的時間，為實際的資料傳輸留下更多的時間。這樣提高了輸送量，特別是在點對多點配置下。
- **減少傳輸延遲開銷** - Nv2 不會輪詢每個用戶端，這允許根據到用戶端的估計距離(傳輸延遲)創建上行調度，從而使介質的使用率最大。這樣提高了輸送量，特別是在點對多點配置下。
- **更好地控制延遲** - 減少開銷、調接週期長度和 QoS 策略等多種方式控制無線網路的延遲。

從 RouterOS v5.0beta5 開始，可以在 wireless 功能表下配置 Nv2，Nv2 協議限制了 511 個用戶端

Nv2 參數

- nv2-qos 設置資料包的優先順序機制，首先資料將從優先順序高的開始發送，這時低佇列資料，要等到 0 佇列優先到達目的地為止才能發送。當高優先順序佇列資料連接滿載，低優先級數據不能被發送，使用這個功能在 AP 上非常有效。
- frame-priority - 能在 mangle 裡手動設置
- default - 默認將小包接收設置為最低延遲優先順序
- nv2-cell-radius (默認 值: *30*); 這個設置會影響連線時間間隔大小，AP 分配給開始連接用

戶端估算用戶端距離的時間週期值。當時間太小，遠端的客戶的可能會出現連接問題，並且或由於“ranging timeout” 斷開連結錯誤，為了保持最高性能，沒有必要的情況下，不要增加這個值。

- on AP: 最遠的用戶端距離，單位 km
- on station: 無作用

- **tdma-period-size** (默認 值: 2) 指定 TDMA 週期為毫秒。有助於較長距離的連結，能略微增加頻寬，當然延遲也同樣會增加

Nv2 可能問題

在長距離通過 tdma-period-size 可以增加輸送量，每個“period”，離開 AP 後不會使用到的傳輸時間(等於一個往返時間 - 時間在幀被發送和從用戶端接收到)，它是用於確認用戶端能接收到從 AP 發出的一個幀，即較長的距離，較長的時間週期不會被使用。

例如，AP 與用戶端距離 30km，幀直接發送到需要 100us，接收一個往返需要大約 200us，tdma-period-size 默認是 2ms，即 10%的時間沒有被使用（1000us=1ms，即 0.2ms/2ms=10%），當 tdma-period-size 增加到 4ms，僅有 5%的時間沒有被使用，如果增加到 60km，往返時間為 400us，未使用時間 20%，這時 tdma-period-size 為 2ms，4ms 時為 10%，更大的 tdma-period-size 值增加連接延遲。

Nv2 相容性

僅 RouterOS 設備支援 Nv2 技術，在搜索時，僅有 RouterOS 設備能發現支援 Nv2 技術的 AP。Nv2 網路將干擾其他網路相同頻道的 AP，也同樣包括附近的 Nv2，當 RouterOS 啟用 Nv2 後將不能連接其他任何基於 TDMA 網路。

Wireless-protocol 參數

從 5.0rc1 開始加入了新的 wireless 設置參數 wireless-protocol，根據無線網路環境配置不同協定，以及需要相容的模式，如下表：

值	AP	client
unspecified	建立基於老版本的 nstreme 或者 802.11	連接到老版本的 nstreme 或者 802.11
any	如同 unspecified	搜索所有匹配的網路，不論協定。
802.11	建立 802.11	只能連接到標準的 802.11 網路
nstreme	建立 Nstreme	只能連接到 Nstreme
nv2	建立 NV2	只能連接到 NV2
nv2-nstreme-802.11	建立 NV2	搜索 Nv2 網路，如果找到有適當的網路，並連接。 否則搜索 Nstreme 網路，如果找到有適當的網路，並連接。 否則搜索 802.11 網路，如果找到有適當的網路，並連接。
nv2-nstreme	建立 NV2	搜索 Nv2 網路，如果找到有適當的網路，並連接。 否則搜索 Nstreme 網路，如果找到有適當的網路，並連接。

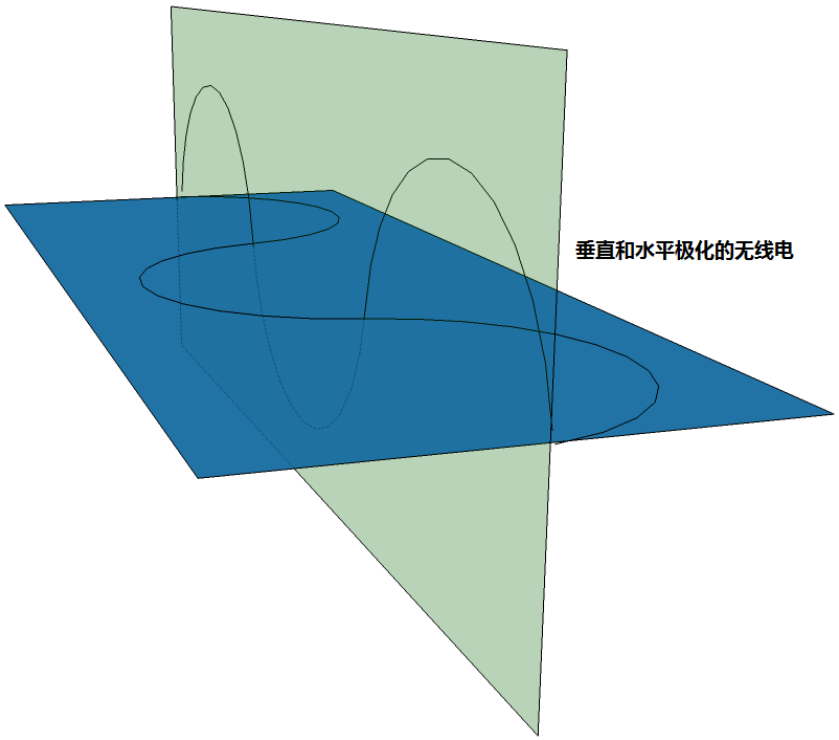
注意：wireless-protocol 值設置 Nv2-nstreme-802.11 指定某種混雜模式或某種類型協定，這樣當客戶機連接的網路通訊協定發生更改時，這些值可以簡化客戶機配置。使用這些參數值可以說明將網路遷移到 Nv2 協定連接。

7.3 配置 802.11n 的 Nv2 協定

要發揮 802.11n 首先要使用 MIMO 技術，即在前面介紹的 MIMO 技術，RouterOS 通過 Nv2 協定優化了 802.11n 協定的傳輸性能選擇設備。

首先我們需要選擇 v5.0 以上的版本，最好選擇 RB400、RB700 或者 RB800 的設備，無線網卡使用 AtherosAR9000 系列，支且持 2x2 的無線網卡，並採用雙極化天線，雙極化天線。

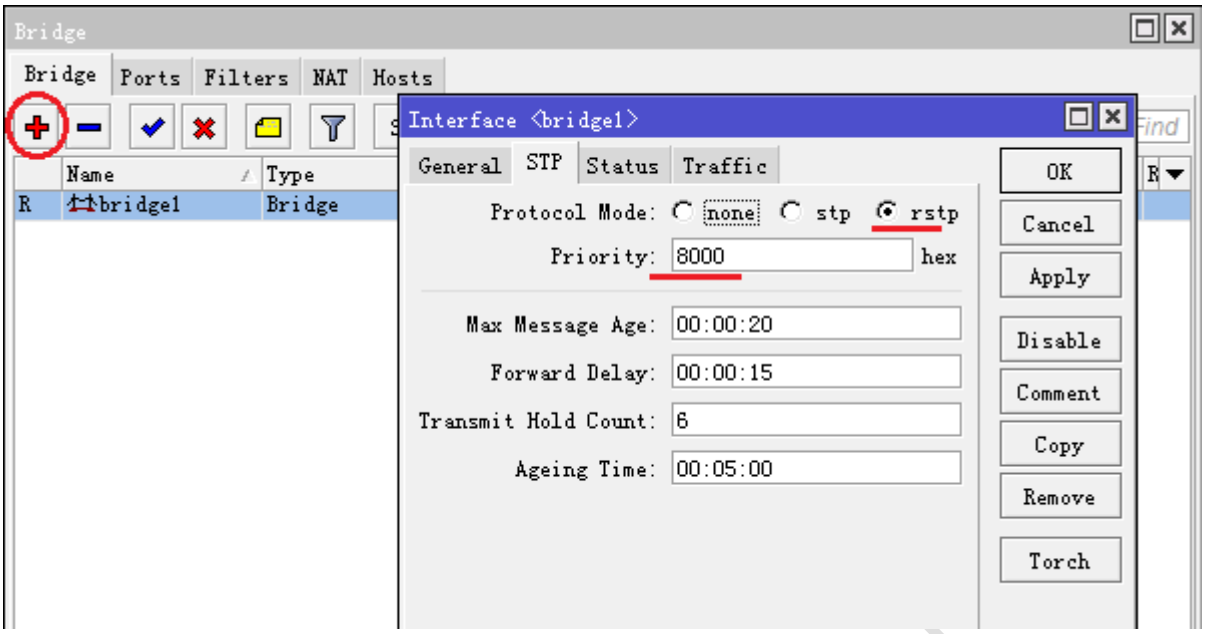
雙極化天線採用垂直和水準極化方式傳輸無線信號，如下圖：



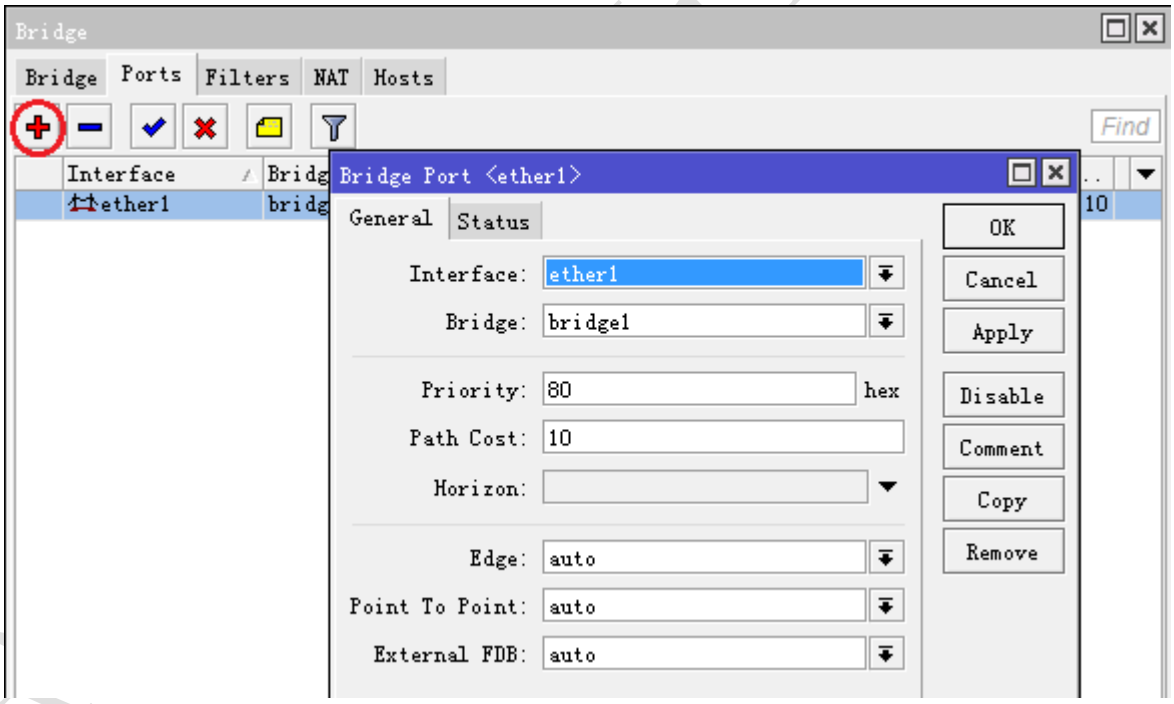
AP-bridge 配置：

首先設置兩個 RouterBOARD 的 bridge 橋接，兩個設備的 bridge 配置基本相同，只是 STP 裡的 priority 參數不同，以下的 bridge 配置 2 台通用。

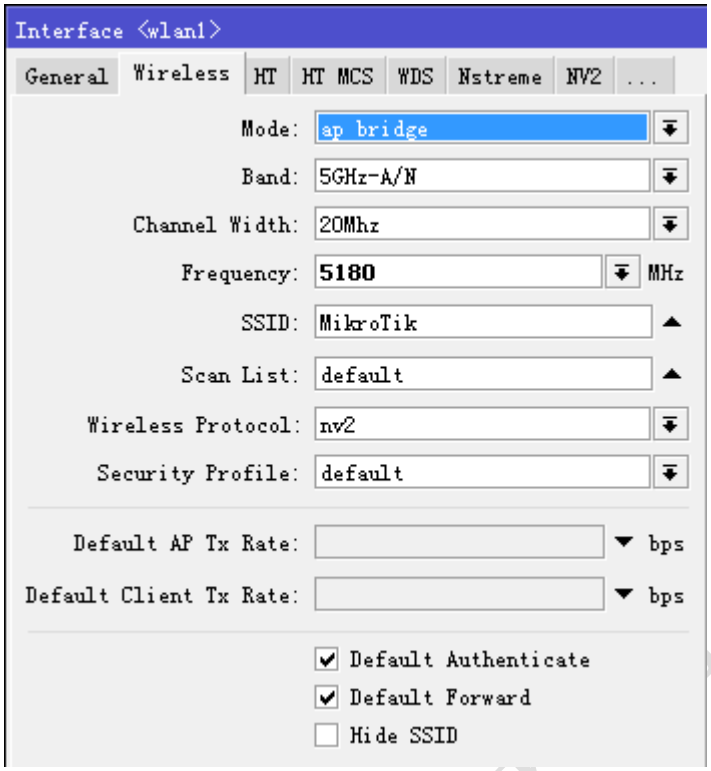
進入 bridge 功能表，添加一個 bridge1，並設置 STP 參數，選擇模式為 rstp，兩台設備的 priority 為 8000：



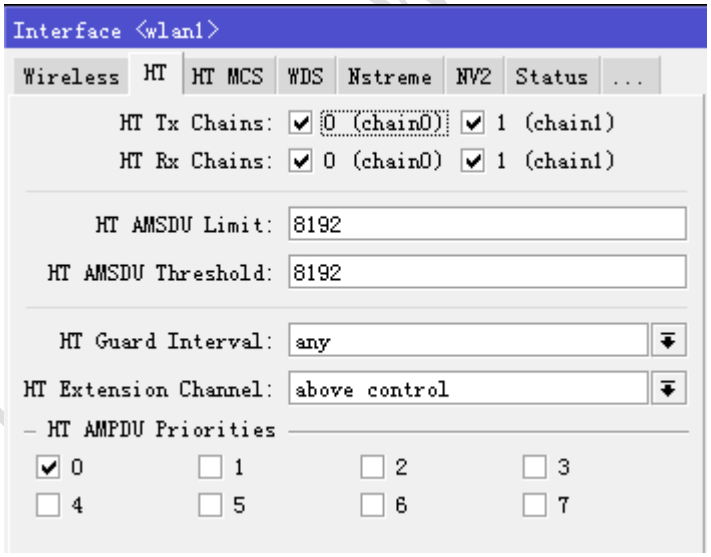
進入 port 標籤，將 ether1 添加到 bridge1 裡，這裡我們只添加 ether1 的網卡，wlan 介面可以不用添加，在後面的無線 wds 配置，由 WDS 配置動態添加



無線網卡為 R52n，採用 5GHz-A/N，SSID 為 MikroTik，wireless-protocol 使用 nv2

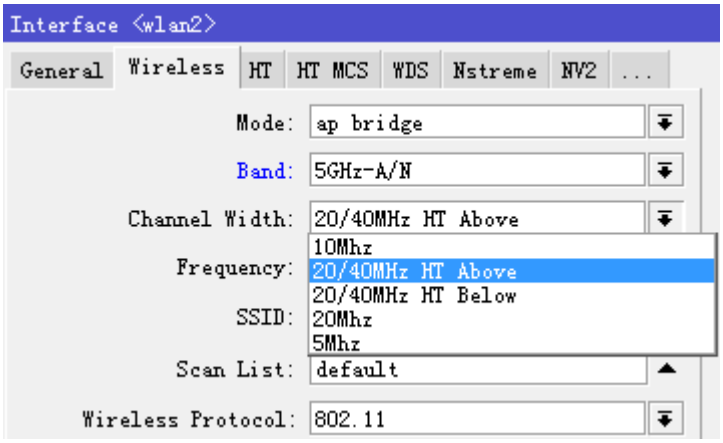


HT 設置，鏈路通道全部開啟，選擇 HT-Extension-Channel=above-control，在 v5.3 版本前的設置

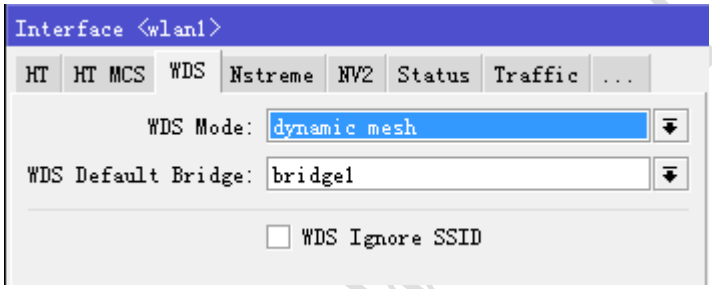


注：在 5.3 版本後 11n 的 HT Extension-Channel 選項變動

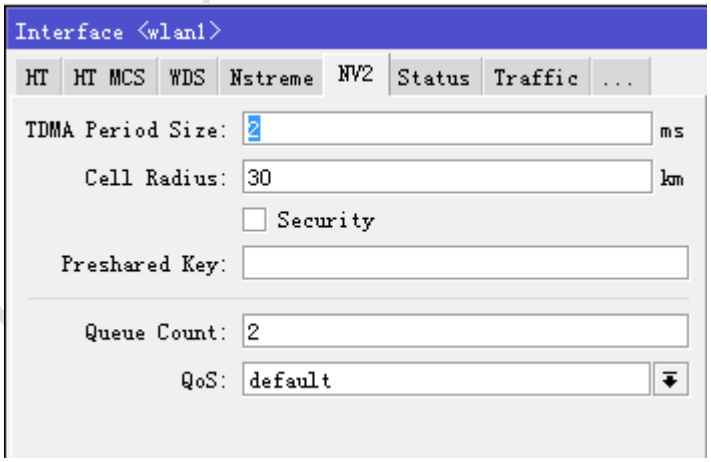
為了更簡單的配置參數，避免選項在不同配置功能表下，從 v5.3 版本開始將 **ht-extension-channels** 融合到 **channel-width** 選項，以前被稱為 40MHz，現在被稱為 40MHz-turbo（非 11n 的卡），並在 802.11n 卡的 HT extension channel 被稱為“20/40MHz Above”和“20/40MHz Below”，而且他們現在僅能在 channel-width 的選項中獲取，如下圖：



WDS 配置，選擇 WDS 模式為動態 mesh，並設置預設橋接為 bridge1，這裡會自動將 WDS 連接添加到剛才的 bridge1 port 中：

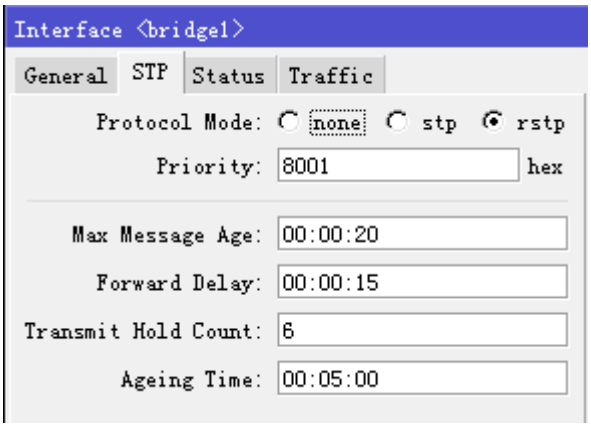


NV2 標籤下的配置預設即可！



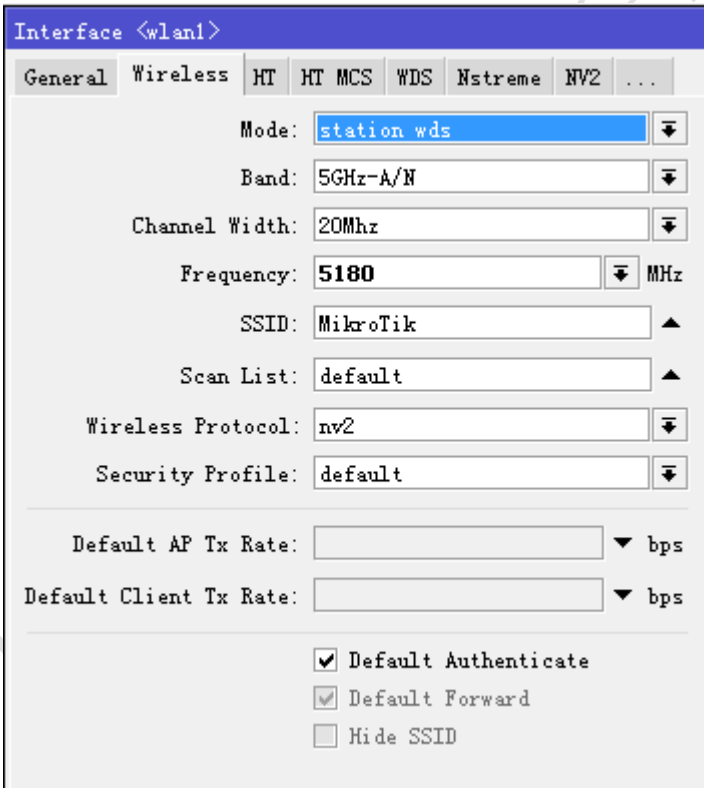
Station 配置

之前的 bridge 配置同上，不在多講解，只是 rstp 的 priority=8001，以區別 ap-bridge 的 STP 優先順序參數

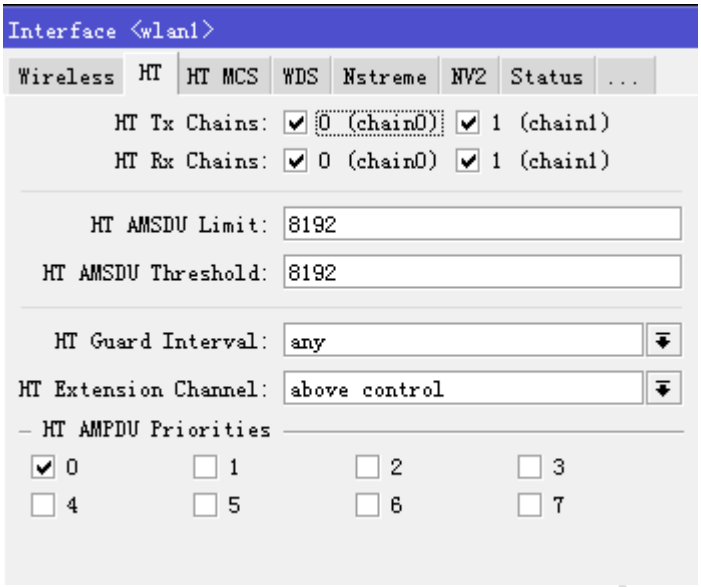


配置 station-wds

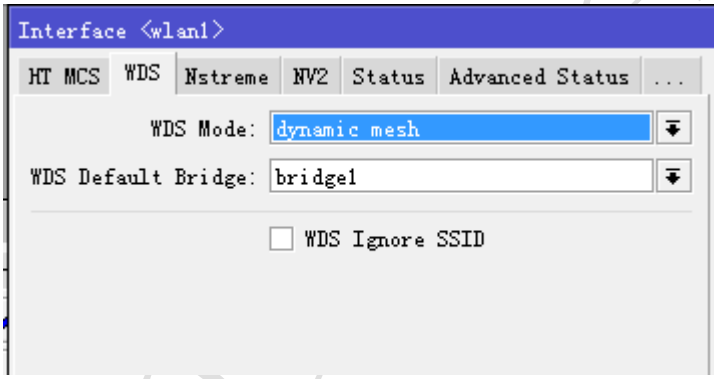
設置 mode 為 station-wds，同樣選擇 5GHz-A/N，SSID 為 MikroTik，並選擇 wireless-protocol 為 nv2



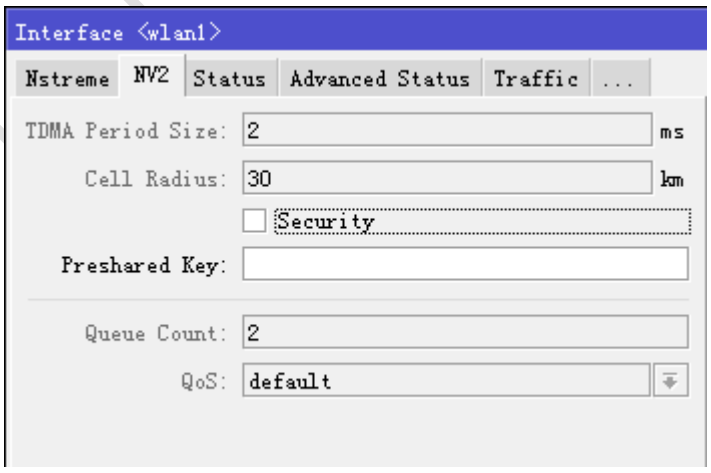
對應的 11n 的 HT 參數，將對應的 tx 和 rx 連結全部打開：



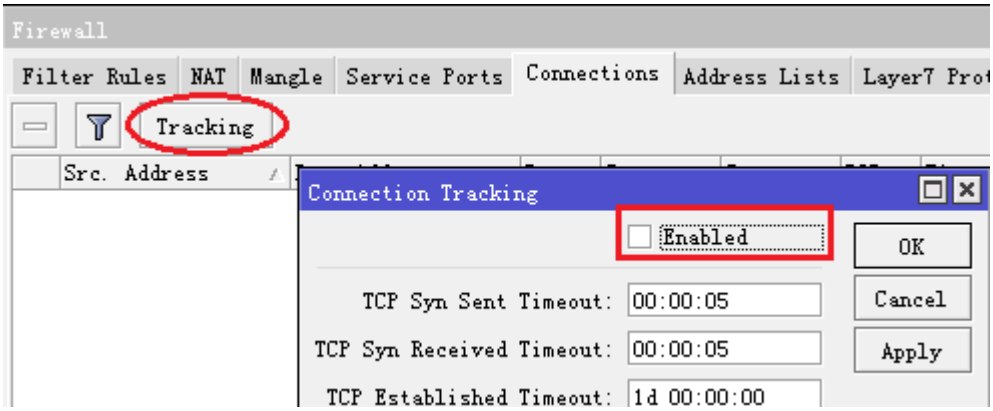
設置 WDS 模式：



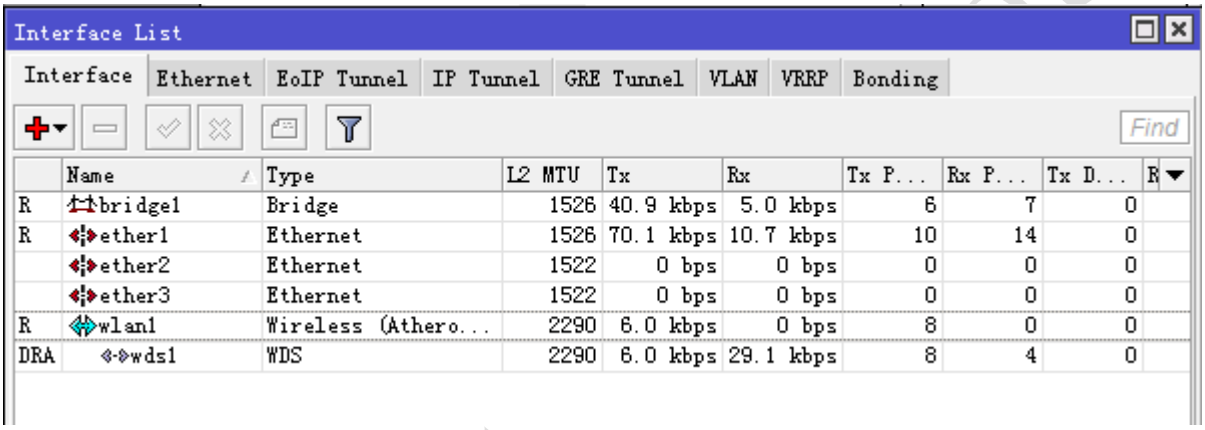
NV2 參數預設設置，也可以根據 AP 的 security 參數選擇加密方式



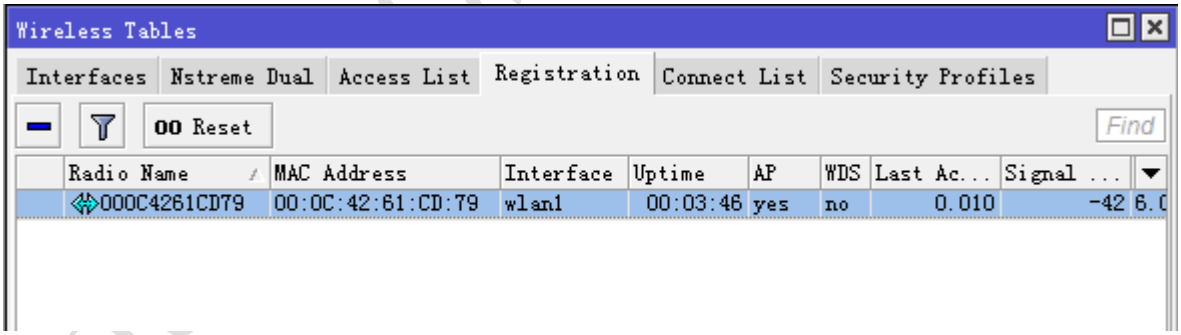
最後優化兩台設備性能，我們將 nat 連接跟蹤關閉，因為不需要使用到 nat 功能，減小 CPU 開銷



連接完成：



信號強度



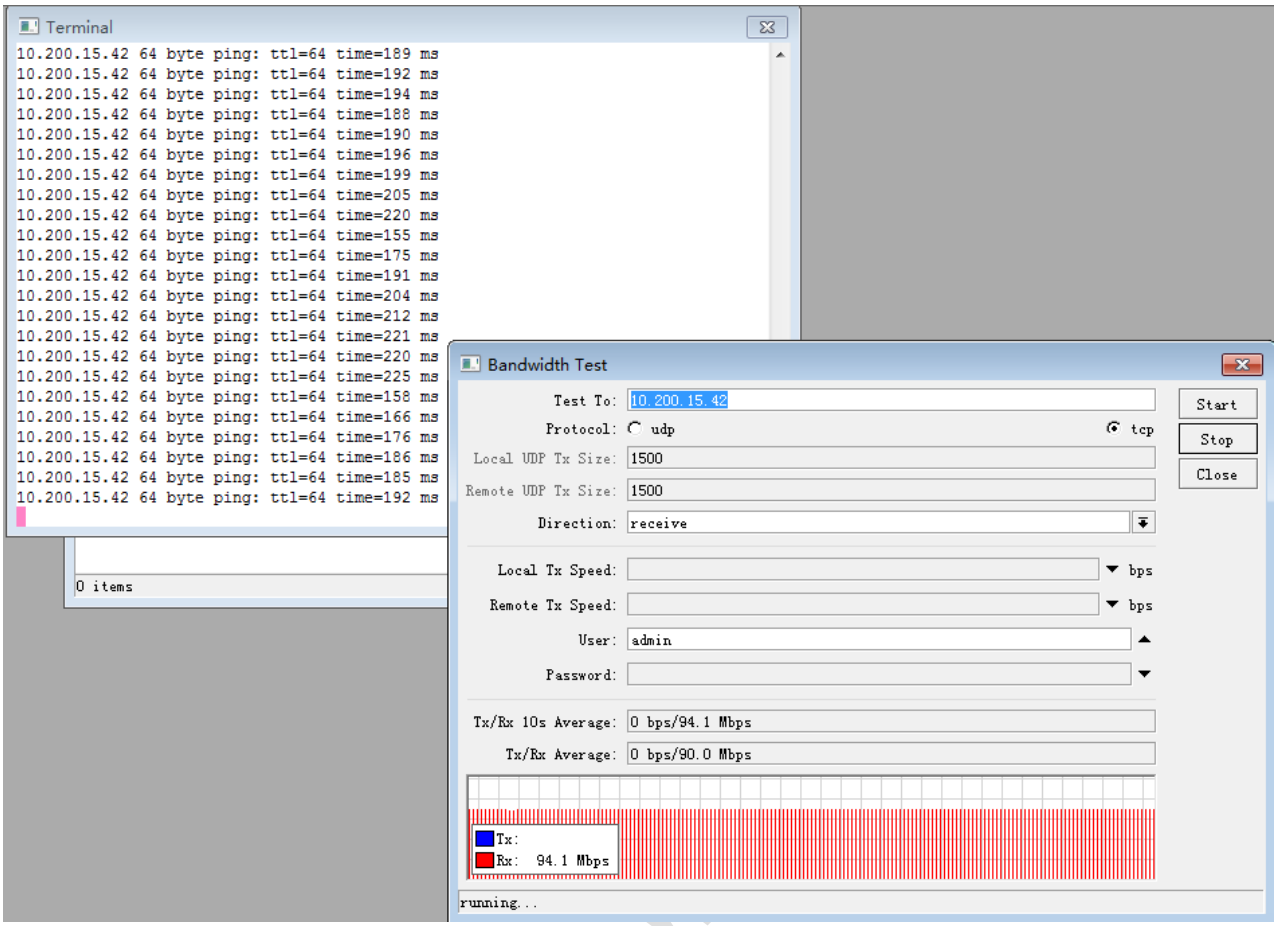
頻寬測試環境：

服務端 RouterOS 3.30，硬體平臺主機板 5000VSA，CPU 至強 5405，2G 記憶體

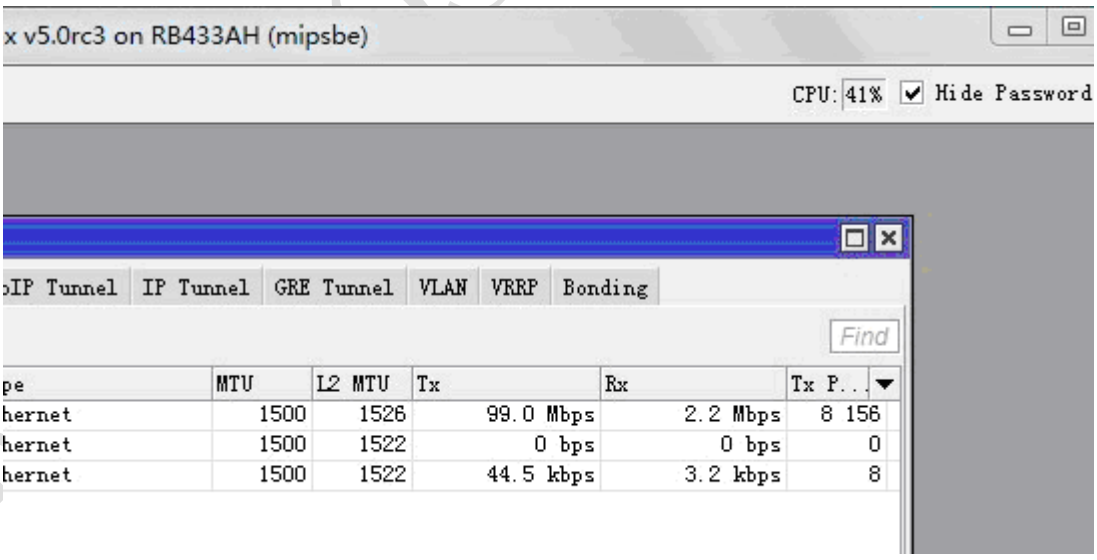
接收端：RouterOS 4.11，硬體平臺：主機板 5000VSA，CPU 至強 5420，2G 記憶體

網路結構：服務端 PC-----RB433AH+R52n ------ RB433AH+R52n --------接收端 PC

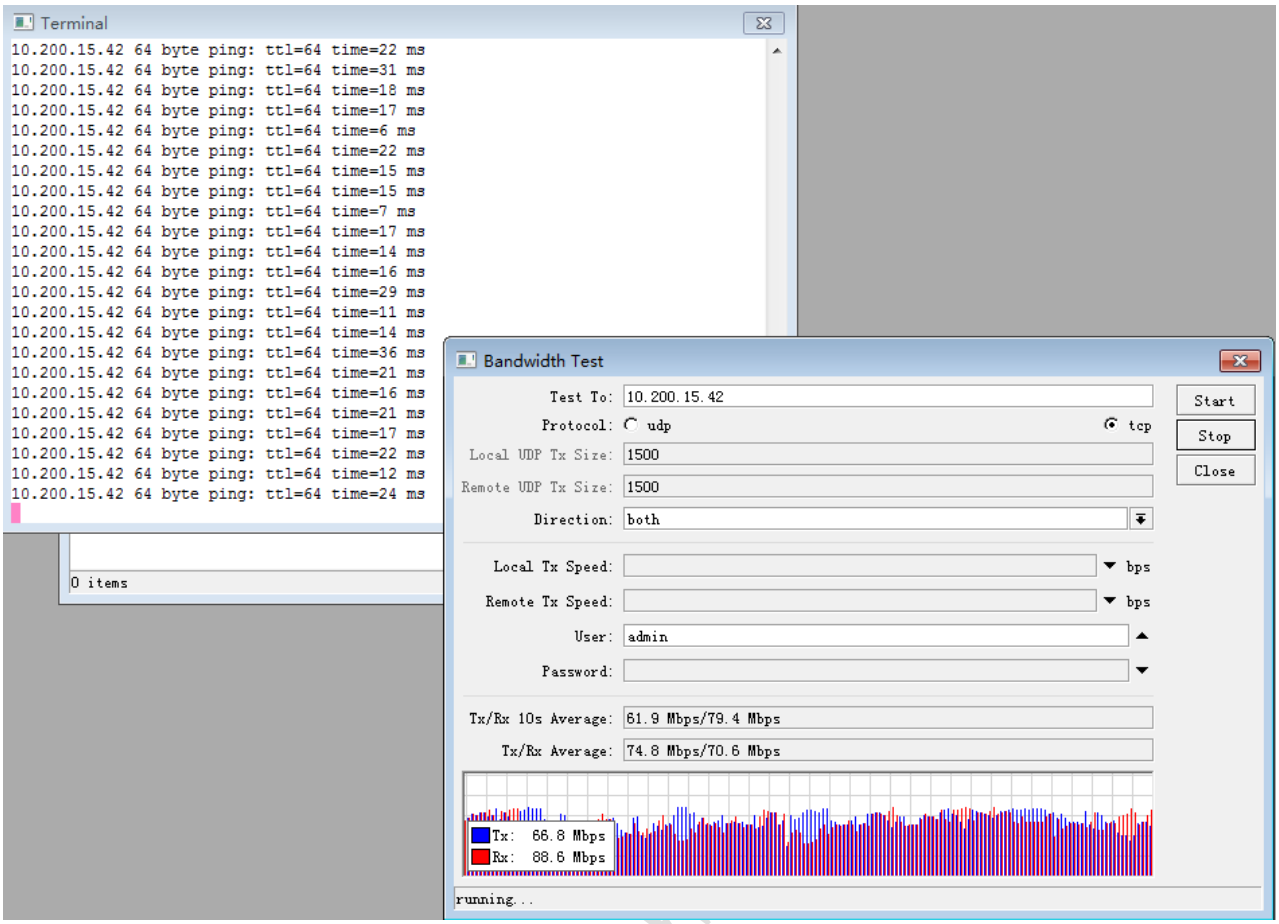
TCP 測試：單向 94.1Mbps



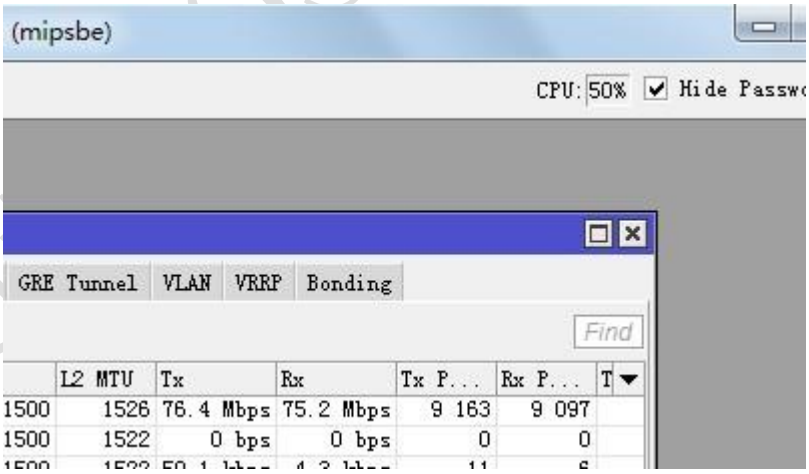
RB433AH 的 CPU 情況： 29-42%



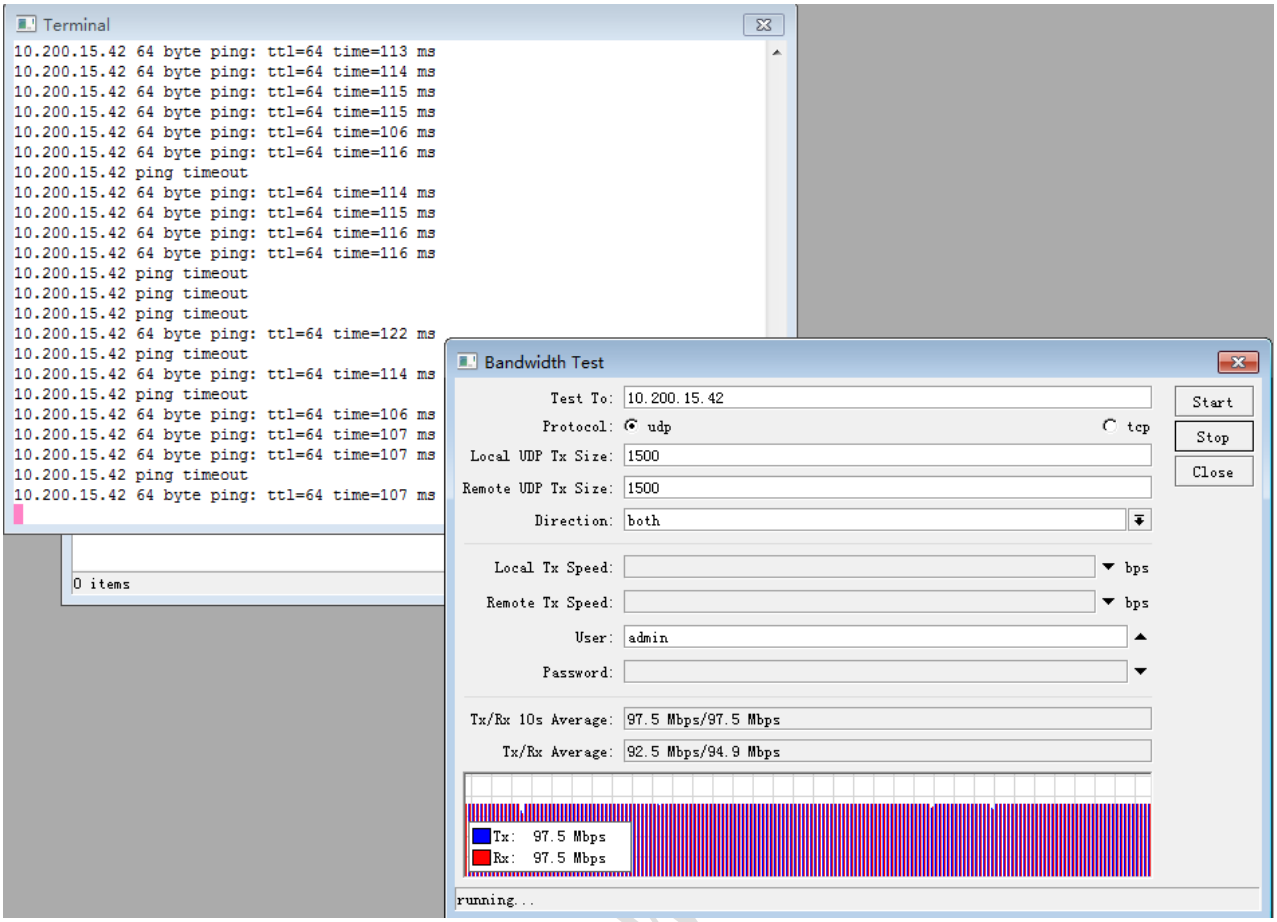
雙向 74 / 70Mbps



RB433AH 的 CPU 情況： 45%-55%

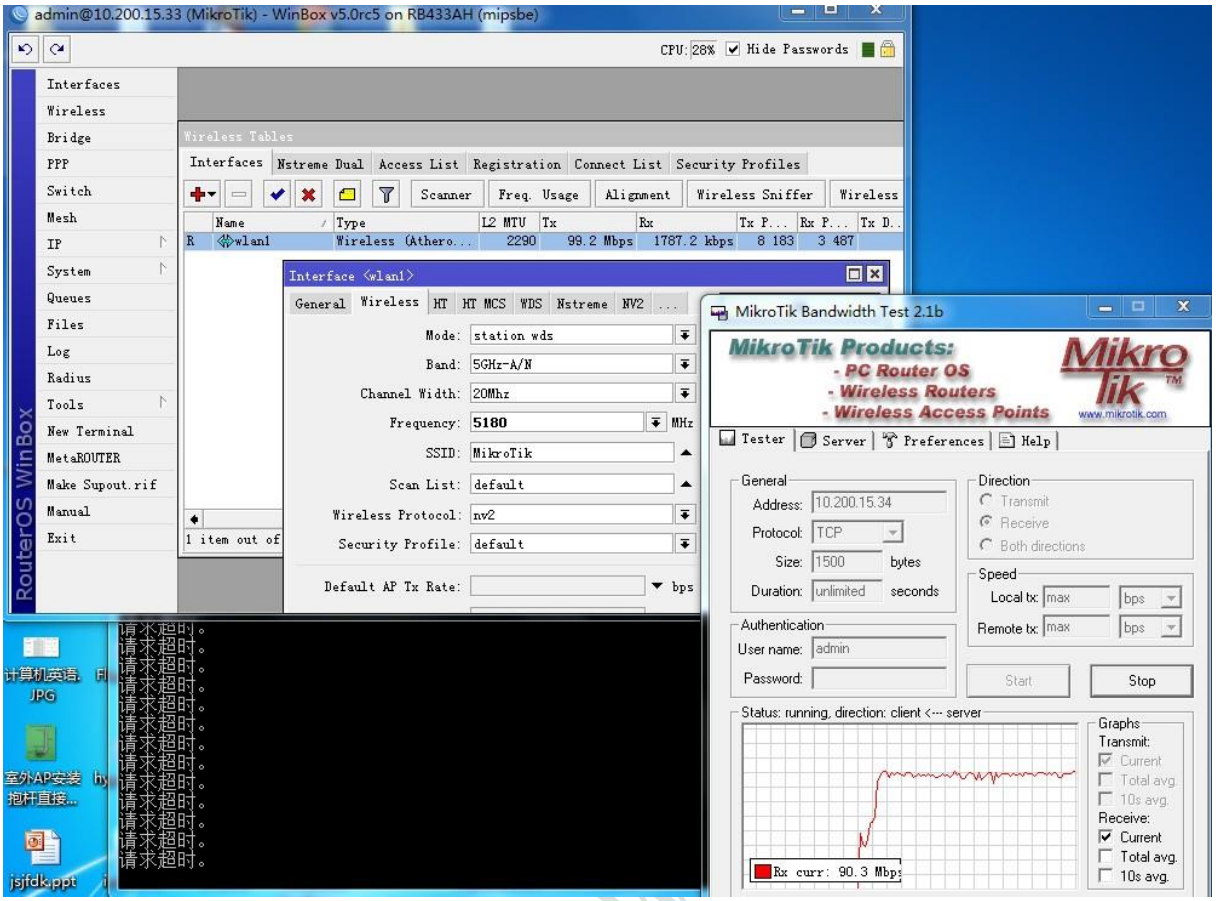


UDP：雙向頻寬都為 97.5Mbps



RB433AH 的 CPU 情況：47%-55%

以下是基於 Windows 的 bandwidth 軟體測試頻寬，可以看到在頻寬滿負荷的情況下，ping 請求已經超時，但連接並未斷開，達到 90.3Mbps，



結論是：RB433AH 由於受到 100M 以太網卡的限制，無法發揮最大的性能，TCP 只能在 94.1Mbps 的頻寬，從雙線頻寬測試看達到了 70Mbps 的全雙工，如果是 RB435G 使用 1000M 以太網卡，單向 TCP 頻寬可以達到 140-150Mbps，UDP 可以在 200-220Mbps，採用 RB800 會更好。

最近一朋友測試了一對 RB711GA-5HnD 的 11n 在 Nv2 下的傳輸，RB711GA-5HnD 是千兆以太網卡，TCP 頻寬達到了 190Mbps。

7.4 Nv2 QoS

802.11 協議基於點對點雙向通信其實是半雙工模式，即通過時分間隔的雙向傳輸。在很多情況下傳輸會出現流量過高而堵塞的情況，我們需要使用 QoS 來保證某些協議能優先通過。

Nv2（Nstreme Version 2）無線協定是 RouterOS v5.0 加入的，Nv2 基於 TDMA 時分多址協議，並只能工作在 RouterOS 設備上。Nv2 的 QoS 採用變數佇列數，能自由配置 2，4 或 8 個佇列，這個佇列基於 IEEE802.1D。

Nv2 的 QoS 是通過數量可變的佇列實現優先順序，佇列傳輸標準是基於 802.1D-2004，實際運行中 QoS 會對所有幀進行分析，佇列優先順序高的優先發送。Nv2 網路中的 QoS 策略是被 AP 控制的，用戶端策略來至於 AP 端，僅需要 AP 的 QoS 策略配置即可

Nv2-qos=default

在這個模式發出的幀首先會被內建的 QoS 策略檢查，佇列的選擇會針對資料包和長度決定。如果內建策略沒有匹配會選擇幀優先順序欄位，如 ICMP 默認是被 Nv2 QoS 優先處理。

Nv2-qos=frame-priority

這個模式 QoS 佇列選擇基於幀優先順序欄位，即我們開啟了自訂的優先順序策略配置

Frame-Priority 欄位 RouterOS 必須知道，可以用的優先順序。預先定義每一個標識為 2、4 或 8 的 Nv2 佇列，分配到實際的資料包中。這個操作可以通過防火牆的 action=set-priority 預先設置。這個

設置可以從以下獲得：

二層在 Bridge Filter 設置

三層在 ip firewall mangle 設置

注意：Frame-Priority 欄位不會被保存在任何資料包頭部，僅僅是 RouterOS 系統內部使用。

佇列的數量選擇基於 802.1D 的‘Frame-Priority’ 欄位代碼， 下面是優先順序與傳輸類型的對比

優先順序	傳輸類型
1	Background
2	Spare
0 (Default)	Best Effort
3	Excellent Effort
4	Controlled Load
5	Video
6	Voice
7	Network Control

映射 ‘Frame-Priority’ 欄位到佇列，依賴 Nv2-queue-count 參數（2、4 或 8 佇列），映射代碼如下：

nv2-queue=2	nv2-queue=4	nv2-queue=8
priority 0,1,2,3 -> queue 0	priority 0,1 -> queue 0	priority 0 -> queue 2
priority 4,5,6,7 -> queue 1	priority 2,3 -> queue 1	priority 1 -> queue 0
	priority 4,5 -> queue 2	priority 2 -> queue 1
	priority 6,7 -> queue 3	priority 3 -> queue 3
		priority 4 -> queue 4
		priority 5 -> queue 5
		priority 6 -> queue 6
		priority 7 -> queue 7

當選擇 nv2-queue=2 時，欄位 0-3 為佇列 0，4-7 為佇列 1，也就是佇列數越多，對資料優先分類越細。

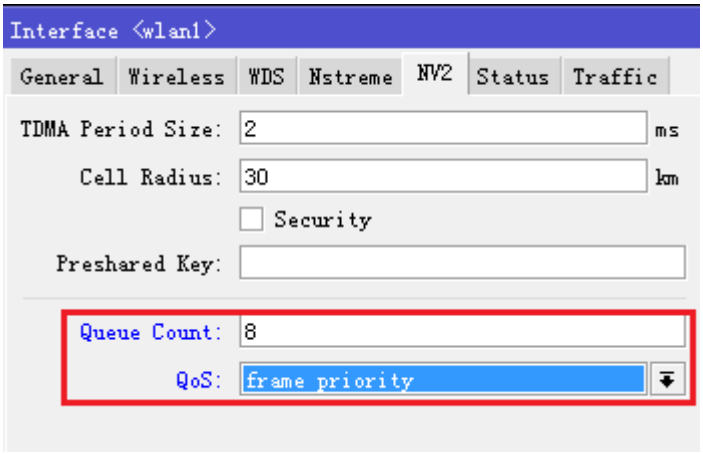
Nv2 QoS 實例

下面通過實例來講解，我們通過 ICMP 來演示如何使用 Nv2 的 QoS，為了方便演示這裡考慮 ICMP 協定優先通過，即不管 Nv2 無線鏈路傳輸頻寬是否瓶頸都讓 ICMP 協定優先通過，保持低延遲。

首先我們需要瞭解等級的排序，我們可以通過下面的表來匹配，但注意 0 為默認級優先順序高於 1 和 2：

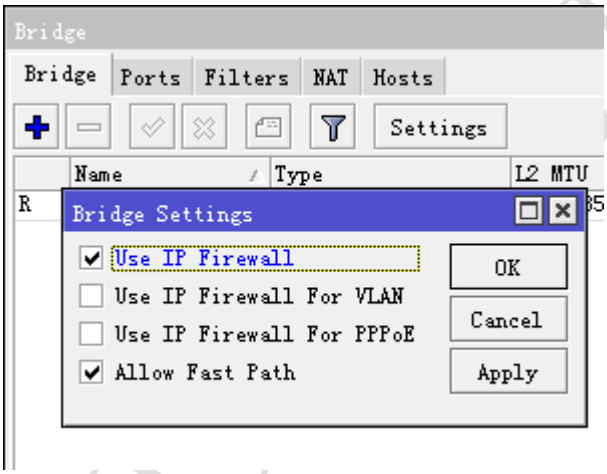
Set Priority	優先順序
1	低
2	
0 (默認)	
3	
4	
5	
6	
7	高

我們將 Nv2 的配置修改為，開啟 8 個佇列，使用 Qos 為 frame priotity



三層 ip mangle 標記

我們需要預先定義 ICMP 協定，這裡需要通過三層的 ip firewall mangle 定義 priority-field 欄位，所以我們需要設置 bridge setting 中的 use-ip-firewall=yes



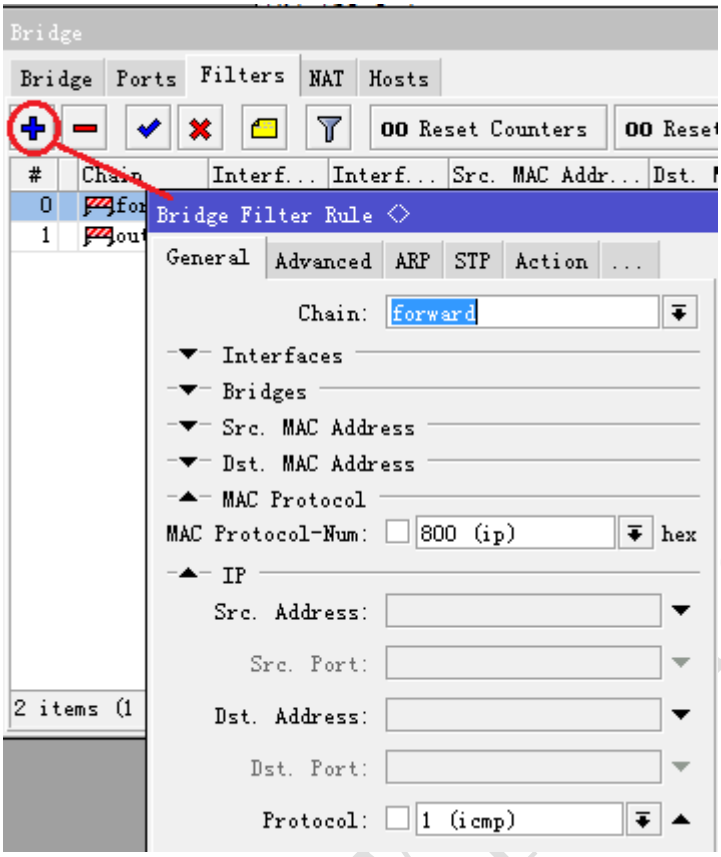

```
/ip firewall mangle
add chain=forward action=set-priority new-priority=7 protocol=icmp
add chain=output action=set-priority new-priority=7 protocol=icmp
```

這裡設置兩個鏈表，因為 forward 負責終端與終端的 icmp 資料，output 負責 AP 設備到對端 station 設備的 icmp 資料，設置 priority=7 是最高優先順序。

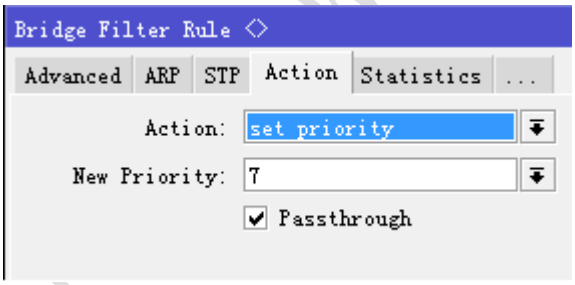
這裡是使用了 mangle 來完成 nv2 qos 配置，我們也可以選擇 bridge filter 完成，這樣可以不需要開啟 use ip firewall，減少複雜的策略調用，降低系統開銷。當如果你有較為複雜的三層和四層協定需要控制，可以調度 mangle 策略

二層 bridge filter 標記

在 bridge filter 中添加策略，如下添加一條 forward 的 icmp 優先策略



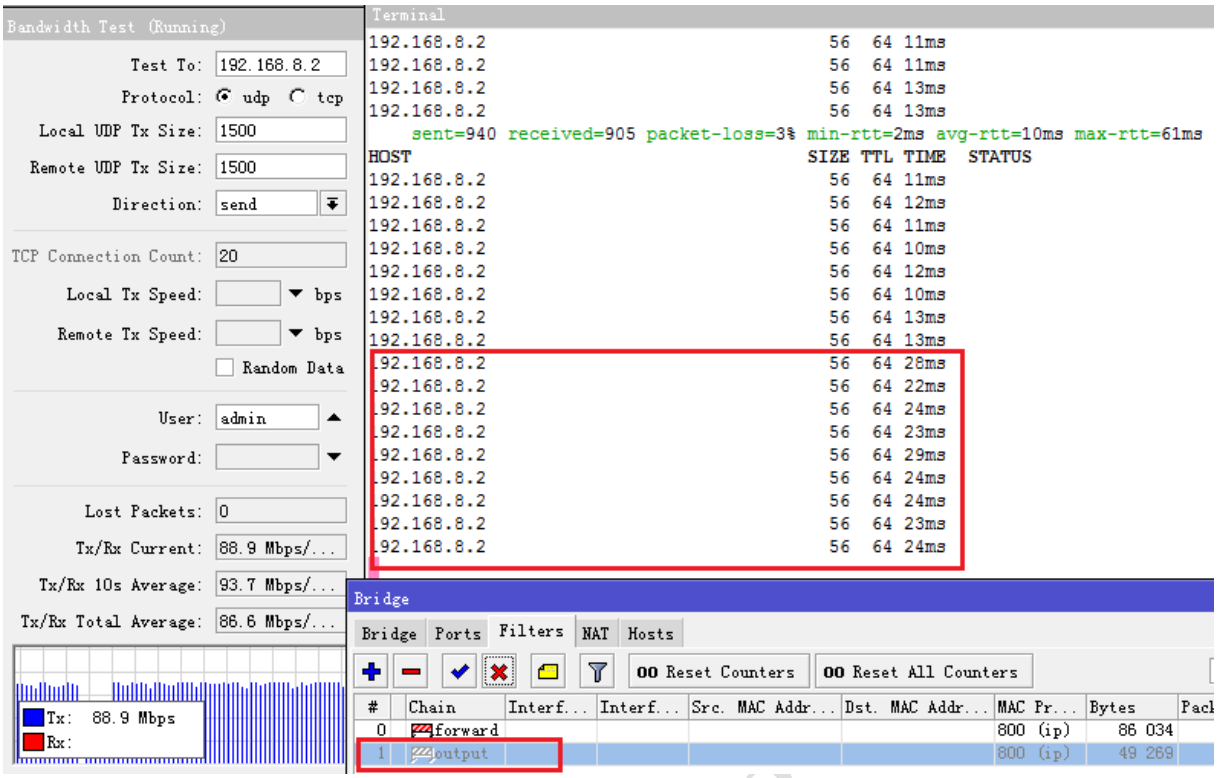
設置 priority 為 7



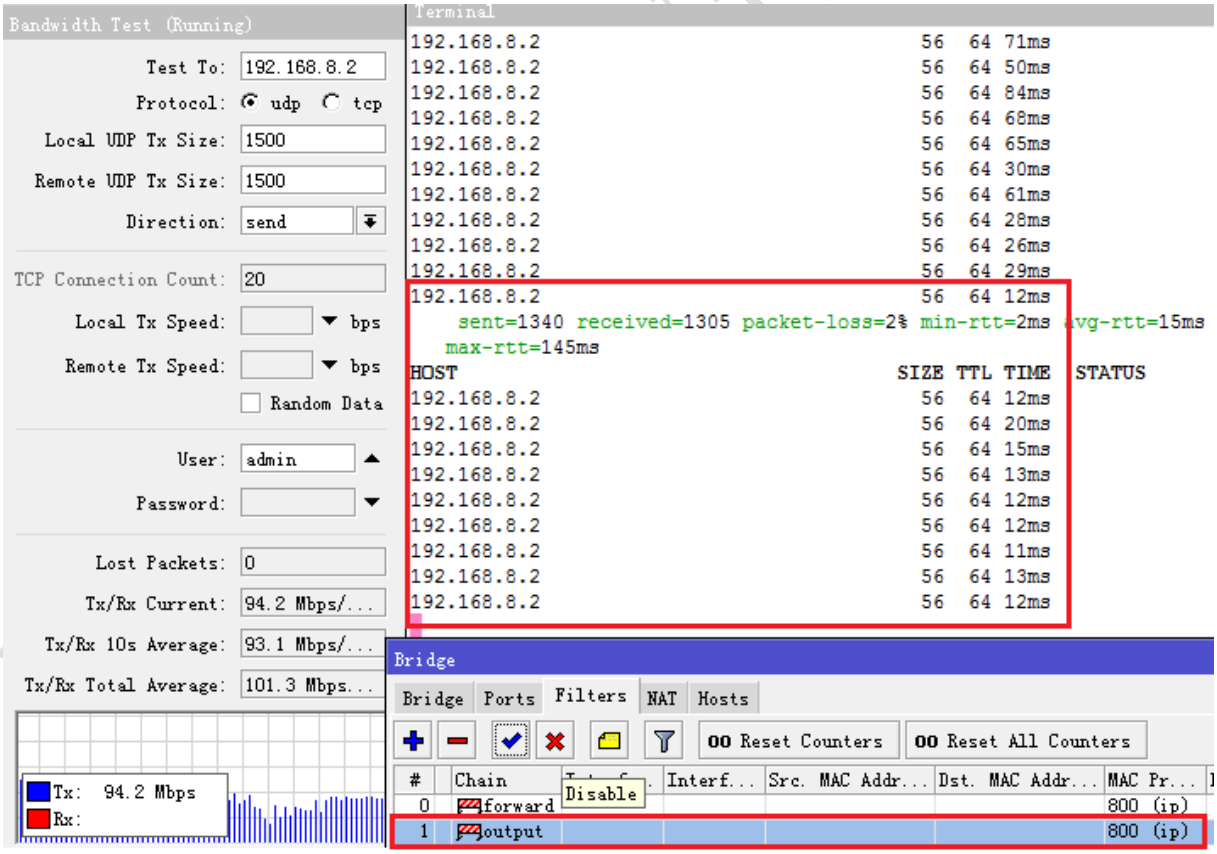
然後我們用相同的配置添加 output 規則：

```
/interface bridge filter
add action=set-priority chain=forward ip-protocol=icmp mac-protocol=ip new-priority=7
add action=set-priority chain=output ip-protocol=icmp mac-protocol=ip new-priority=7
```

當標記完成後，我們就可以測試 Nv2 的 QoS 應用是否生效，我們將頻寬測試開啟，延遲增加到 20～28m，這是 output 的 icmp 規則被禁用



我們開啟 output 規則，明顯延遲下降到 11～13ms



這裡通過 ICMP 介紹了如何讓指定的資料優先通過基於 Nv2 的 AP，根據實際生產環境，你可以制定自己需要的 QoS 規則，例如協定、端口或 L7 應用層等

7.5 Nv2 AP Synchronization

該功能是讓多個 MikroTik Nv2 AP 在同一區域以更好的方式並存，減少彼此之間的干擾，該功能將同步多個 AP 之間在相同頻段的傳輸/接收劃分時間窗口。這允許同一區域多 AP 重複使用相同的無線頻率，為多個 AP 提供更靈活的頻率規劃。

Nv2 同步過程：

- 首先要選擇並設置 Nv2 AP 同步主機，即"nv2-mode=sync-master"，為 Nv2 AP 同步設置通訊密碼"nv2-sync-secret"
- Nv2 從 AP 需要設置"nv2-mode=sync-slave"，並與主 Nv2 AP 設置相同無線頻率，以及相同的"nv2-sync-secret"值
- 當主 AP 啟用，從 AP 將嘗試通過指定的"nv2-sync-secret"搜索主 AP
- 當主 AP 找到從 AP 後，從 AP 將計算到主 AP 的距離，因為可能與主 AP 不在相同的位置。
- 然後從 AP 開始工作在 AP 狀態，並同步主 AP 匹配週期大小和下行比率
- 另外，如果有其他從 AP，可以通過該從 AP 進行同步。
- 從 AP 定期監聽主 AP，並檢查“nv2-sync-secret”是否仍然匹配並再次調整參數。如果主 AP 介面被禁用/啟用，那麼所有從介面也將被禁用，並將重新開始同步過程。
- 如果主 AP 停止工作，從 AP 同樣將停止，他們之間將不在同步資訊

配置事例

主 AP:

```
/interface wireless set wlan1 mode=ap-bridge ssid=Sector1 frequency=5220
nv2-mode=sync-master nv2-preshared-key=clients1 nv2-sync-secret=Tower1
```

從 AP:

```
/interface wireless set wlan1 mode=ap-bridge ssid=Sector2 frequency=5220
nv2-mode=sync-slave nv2-preshared-key=clients2 nv2-sync-secret=Tower1
```

在從 AP 上監控介面狀態 Monitor interface on the Slave AP:

```
[admin@SlaveAP] /interface wireless> monitor wlan1
                  status: running-ap
                 channel: 5220/20/an
       wireless-protocol: nv2
             noise-floor: -110dBm
      registered-clients: 1
    authenticated-clients: 1
          nv2-sync-state: synced
         nv2-sync-master: 4C:5E:0C:57:84:38
       nv2-sync-distance: 1
     nv2-sync-period-size: 2
  nv2-sync-downlink-ratio: 50
```

主 AP Debug 日誌

```
09:22:08 wireless,debug wlan1: 4C:5E:0C:57:85:BE attempts to sync
```

從 AP Debug 日誌:

```
09:22:08 wireless,debug wlan1: attempting to sync to 4C:5E:0C:57:84:38
09:22:09 wireless,debug wlan1: synced to 4C:5E:0C:57:84:38
```

7.6 Nstreme Dual

Nstreme Dual 是早期的 Nsteme 協議的改進，即進一步提高頻寬，類似採用 2x2 傳輸方式，只不過是單向的 2x2。

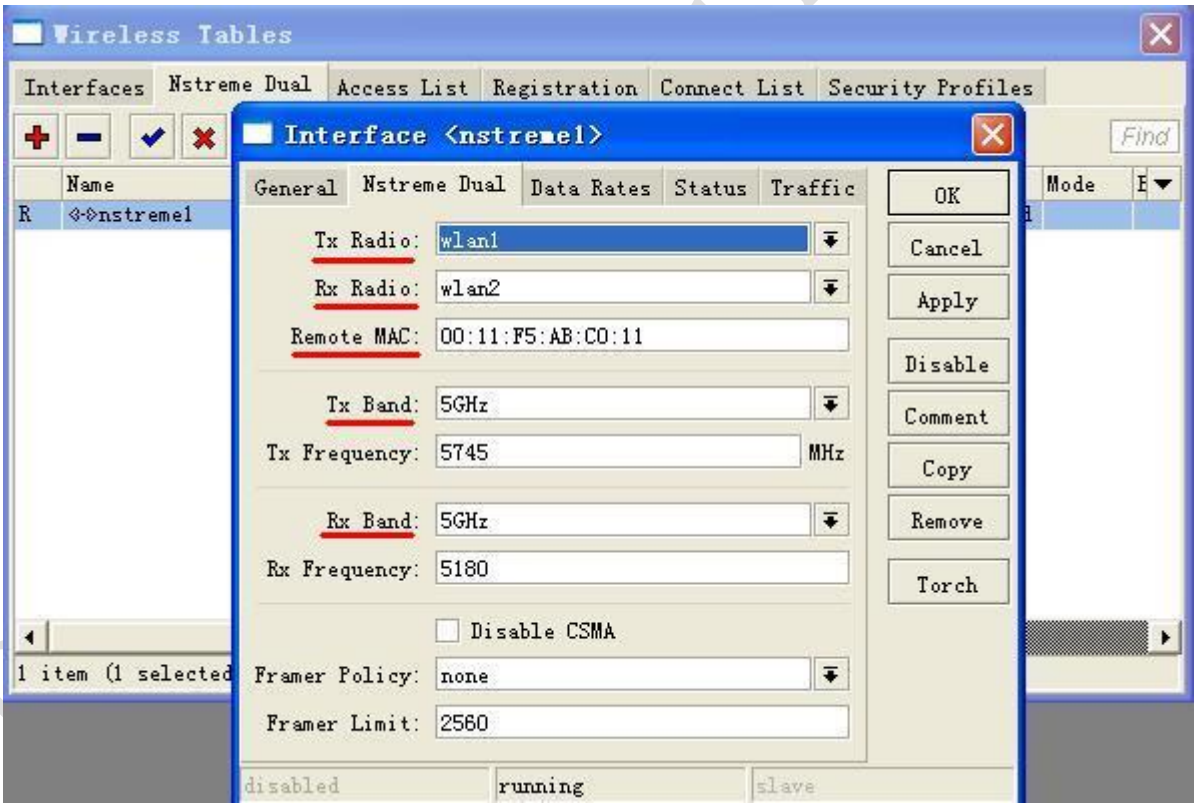
wireless 的 Nstreme Dual 目錄項中，添加 nstreme 介面，需要注意以下參數：

- **Tx Radio：傳輸網卡**
- **Rx Radio：接收網卡**
- **Remote MAC：遠端 nstreme 介面 MAC 位址（非物理網卡 MAC 位址）**
- **Tx band：傳輸頻段**
- **Tx frequency：傳輸頻率**
- **Rx band：接收頻段**
- **Rx frequency：接收頻率**

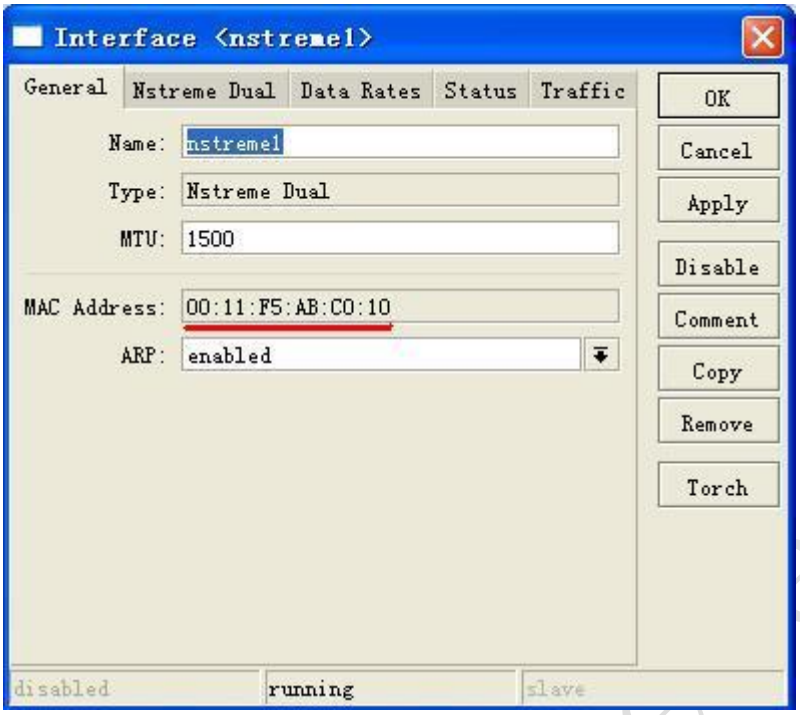
注意：兩邊設備的傳輸和接收頻段和頻率必須相同，雙方的傳輸和接收頻率應當對應。

假設我們有兩個網站：網站 1 和網站 2，都採用 Dual radio Point-to-Point mode (nstreme2)的方式連接

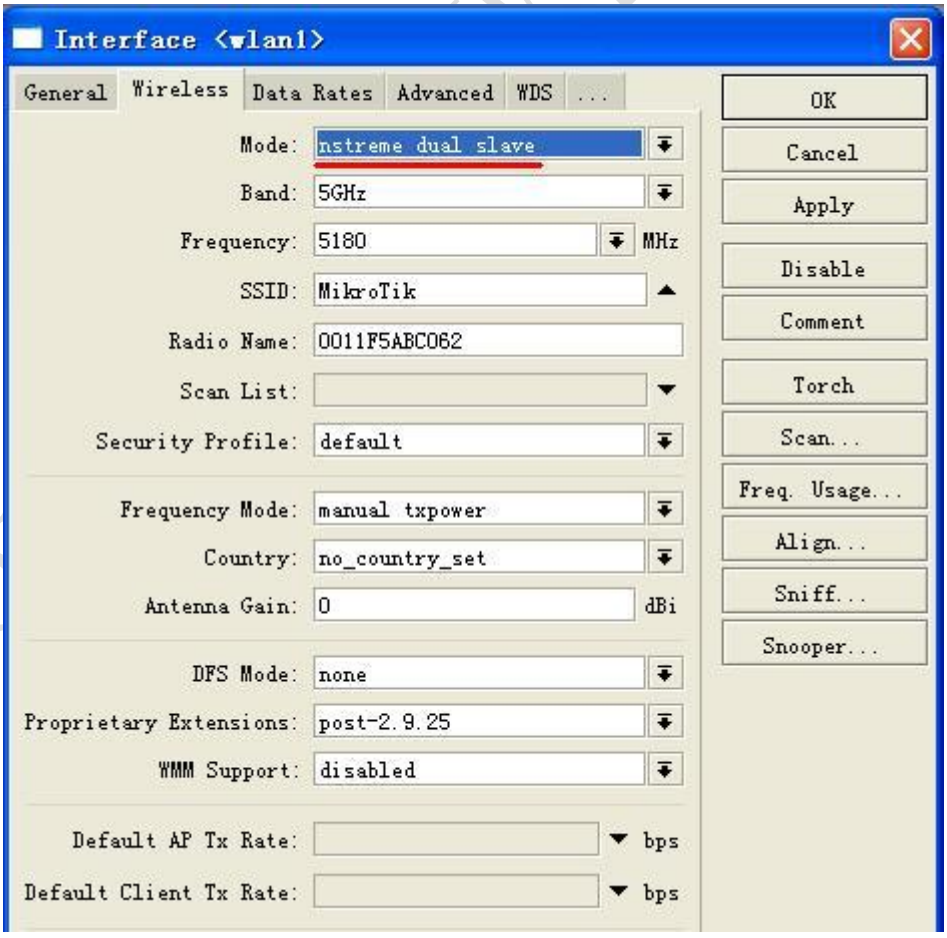
網站 1 的 Nstreme1 配置如下：



這裡我們需要記錄下網站 1 的 nstreme 的 MAC 位址，注意不是無線網卡的 MAC 位址：

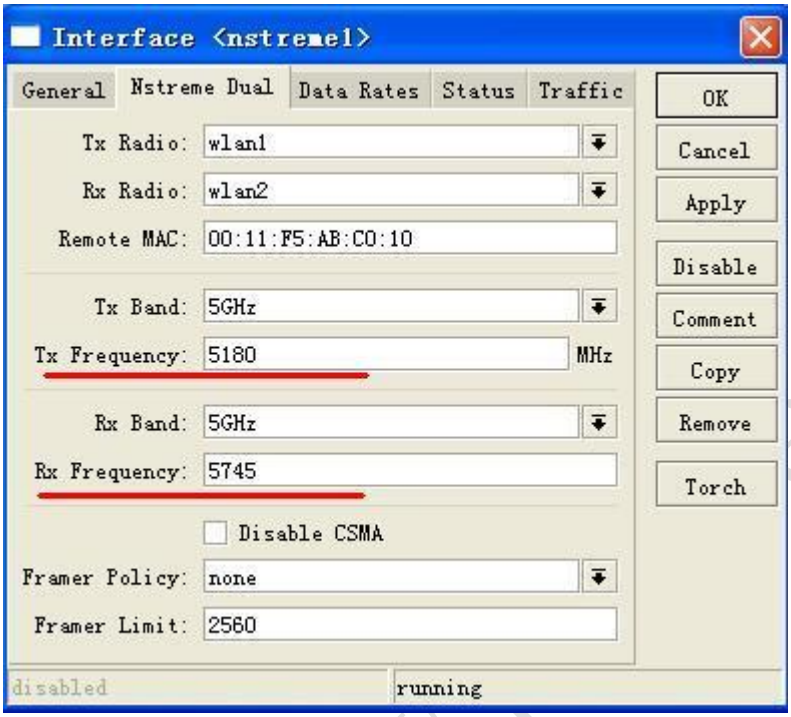


接下來配置網站 1 的兩張無線網卡參數，兩張無線網卡的 Mode 設置為 nstreme dual slave,其他參數默認即可。



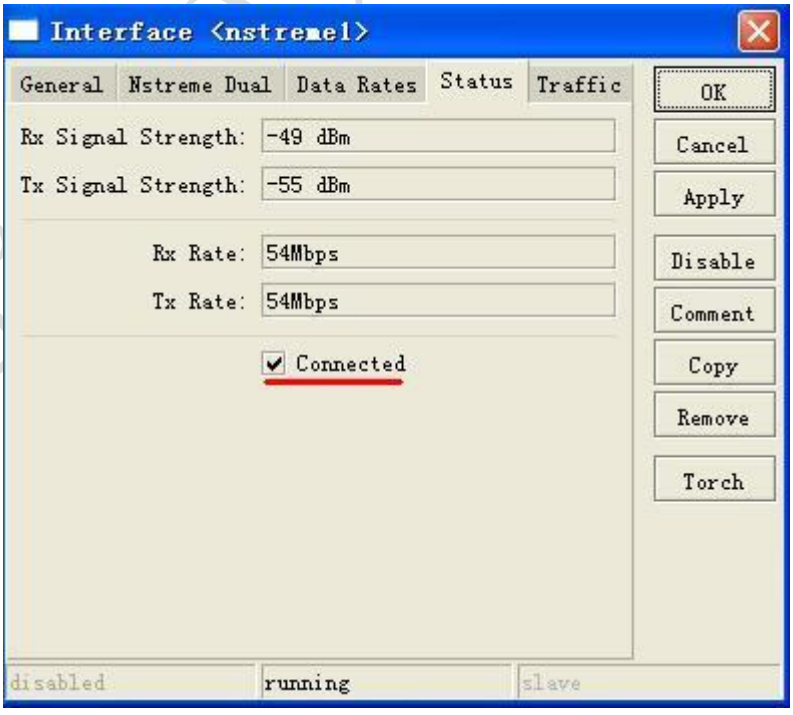
這裡當無線網卡模式選擇為 nstreme dual slave,網卡的頻率和其他參數將從屬於之前添加好的 nstreme1 網卡介面。

網站 2 的配置與網站 1 的配置相同，但有一個地方需要注意，Tx Radio 和 Rx Radio 需要設置對應。如下圖：

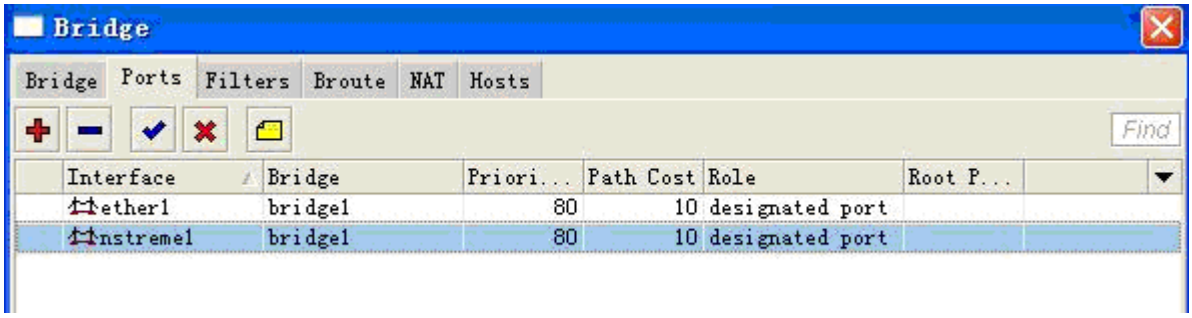


網站 2 的 Tx 頻率要和對面網站 2 的 Rx 頻率相同，同樣網站 2 的 Rx 頻率需要和網站 1 的 Tx 頻率對應

當 Nstreme 連接上後，在 status 中顯示 connected，並能看到其信號強度和速率:



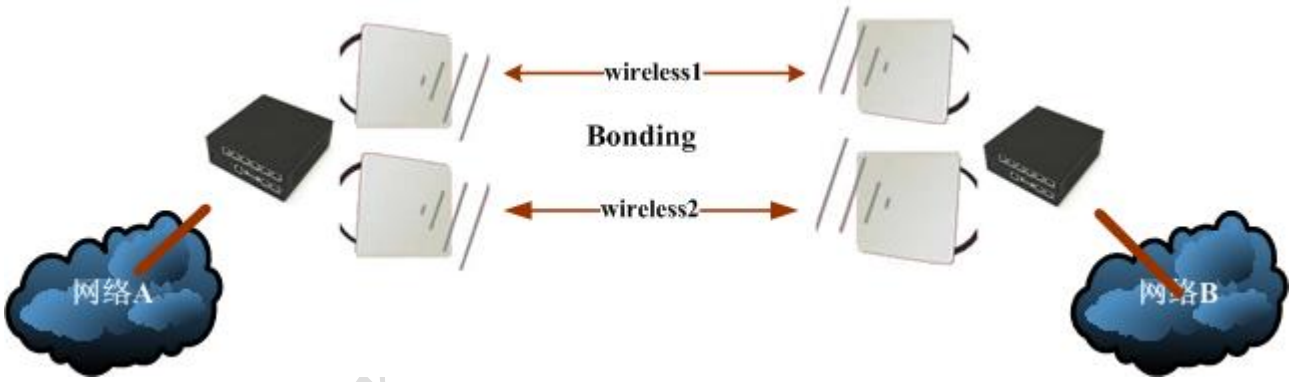
連接完成後，同樣將無線建立橋接，進入 bridge 中，將 ether1 與 nstreme 歸入 bridge1 中



7.7 無線網路 Bonding

Bonding 是基於二層的網卡綁定協定，可以將多個以太網卡加入到一個虛擬連接中，提供速率傳輸和容錯功能。RouterOS 不僅支援普通的以太網卡的綁定，也可以支援基於 802.11 傳輸協定的無線網卡綁定

Bonding 操作是將 2 組或多組無線模組綁定在一起，起到頻寬合併的作用，提高無線頻寬的輸送量。Bonding 功能是採用的是二層鏈路聚合頻寬，所以對三層的 IP 網路沒有直接的影響，和普通的三層負載均衡不同的是，不會受路由策略的影響。



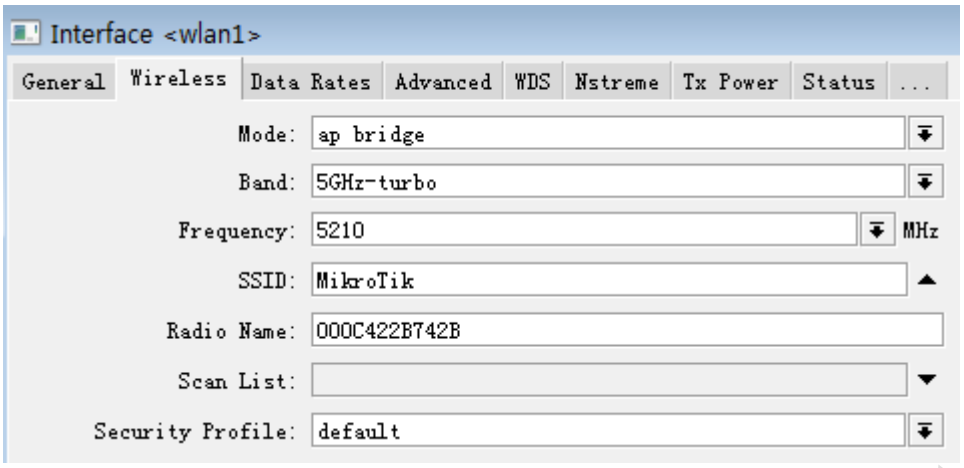
這裡我們通過兩台 RB600A 做測試，首先我們啟用 2 個無線模組，配置 5G-Turbo 模式，提高無線頻寬的輸送量。

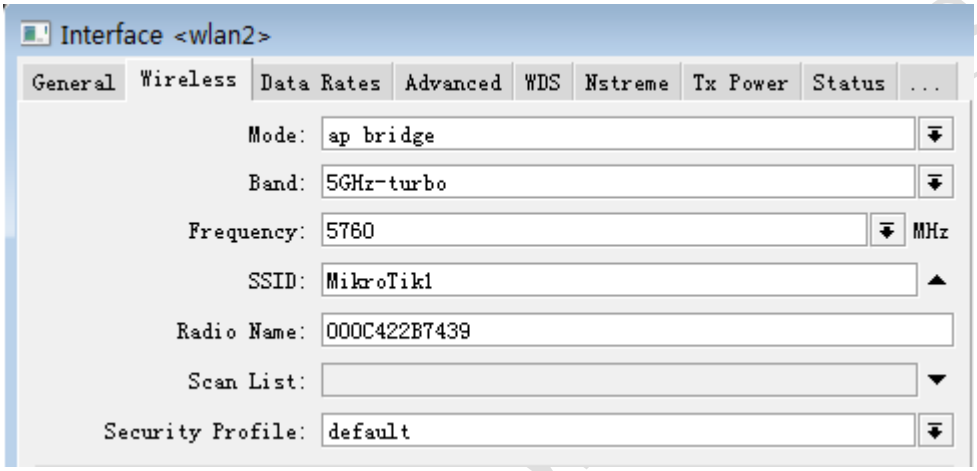
AP 配置

我們首先配置 AP 端的 RB600A，分別對 Wlan1 和 Wlan2 兩個無線模組做配置：

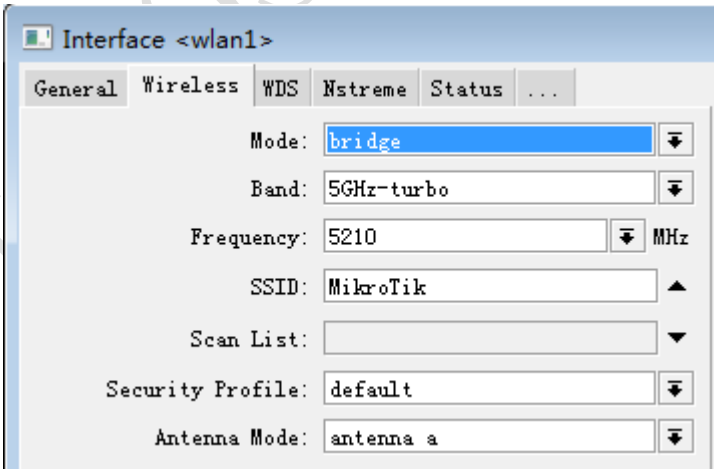
Wlan1	Wlan2
Mode：AP-bridge（bridge） Band：5GHz-turbo Frequency：5210MHz SSID：MikroTik	Mode：AP-bridge（bridge） Band：5GHz-turbo Frequency：5760MHz SSID：MikroTik1

下面分別是 wlan1 和 wlan2 的 AP 部分配置：

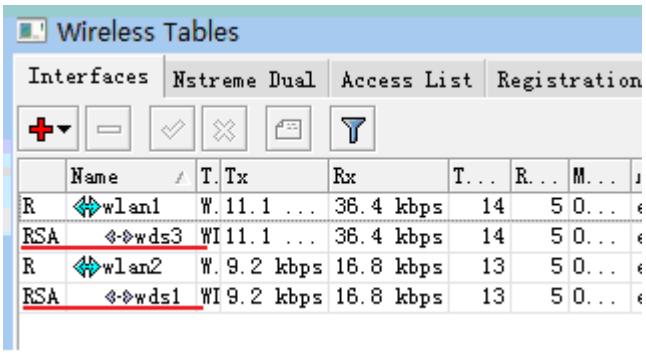




在 AP 下如果 ap-bridge 連接有問題，可以改用 bridge 模式，有時需要禁用啟用網卡，才能正常連接：

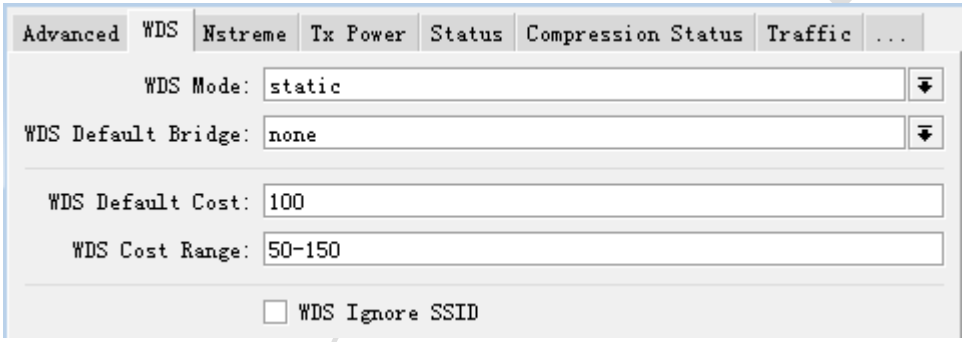


注：這裡我們需要配置 WDS 的模式，因為在為連接前我們需要將 WDS-Mode 設置為 dynamic，當連接後在將 WDS 介面修改為 static 的靜態模式。

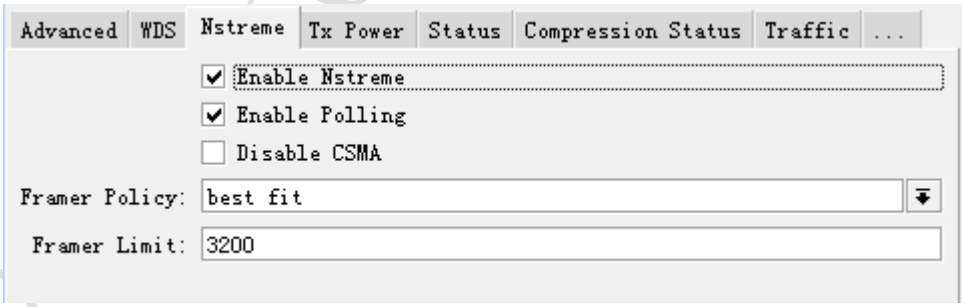


如上圖，將 wds3 和 wds1 修改為靜態的，首碼為 RSA 的標記，如果不是靜態的模式，會顯示 DRA。

連接後，我們將 wlan1 和 wlan2 的 WDS-Mode 修改為 Static，因為 bonding 需要指定網路介面，而如果採用動態 dynamic 模式，會在每次連接斷開後，原來的 WDS 介面會失效，所以需要將 WDS 模式設置為靜態的，在重複連接的情況下 bonding 功能才不會失效。



為了保證更好的頻寬和傳輸品質，啟用 Nstreme 協定，配置如下：

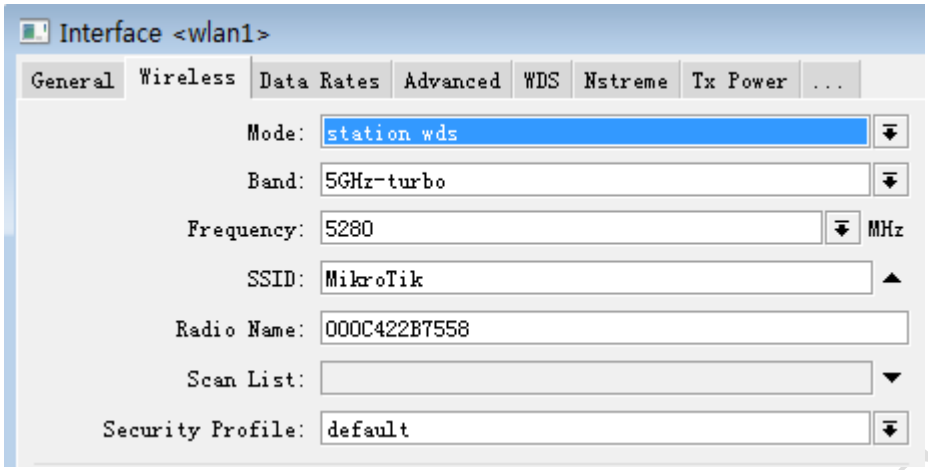


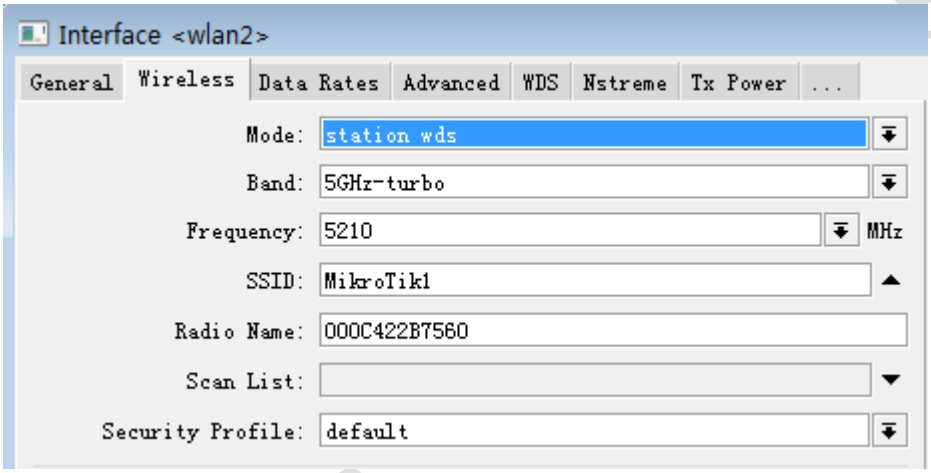
Station 配置

將 Station 設備的 wlan1 和 wlan2 配置為 station-wds 模式

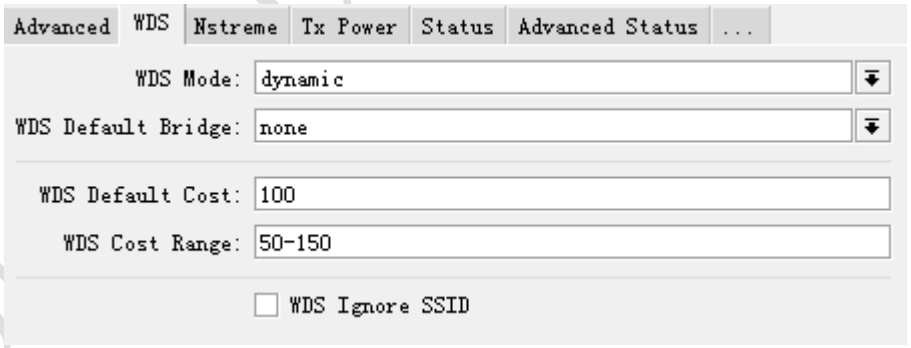
Wlan1	Wlan2
Mode: station-wds	Mode: station-wds
Band: 5GHz-turbo	Band: 5GHz-turbo
SSID: MikroTik	SSID: MikroTik1

下面分別是 wlan1 和 wlan2 的 station 部分配置

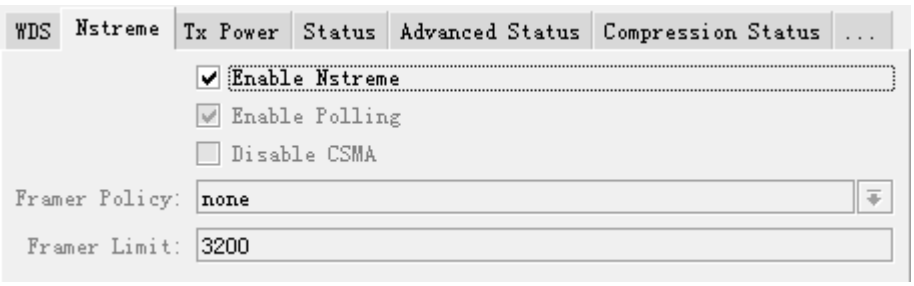




這裡將 wds 參數配置為 dynamic

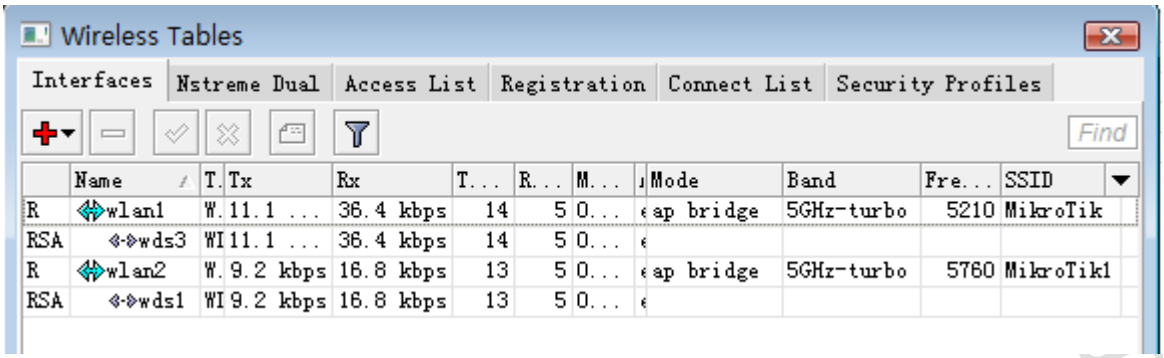


同樣，啟用 Nstreme 協議參數：

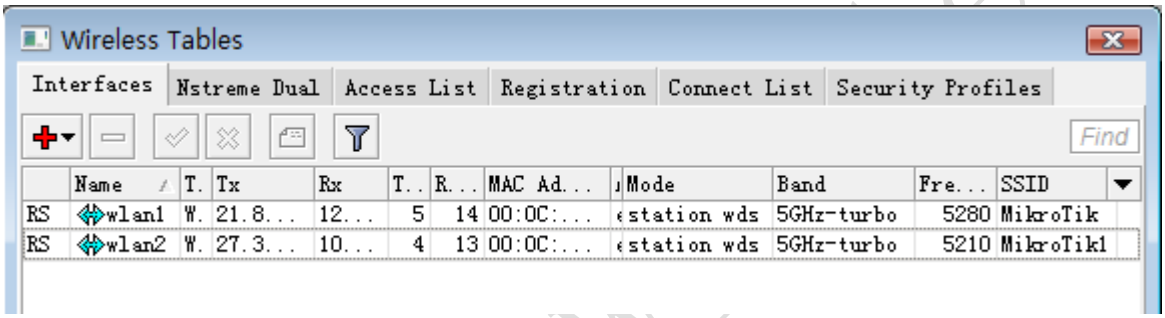


配置完成無線參數後，AP 和 Station 相互連接成功：

AP 連接狀態：



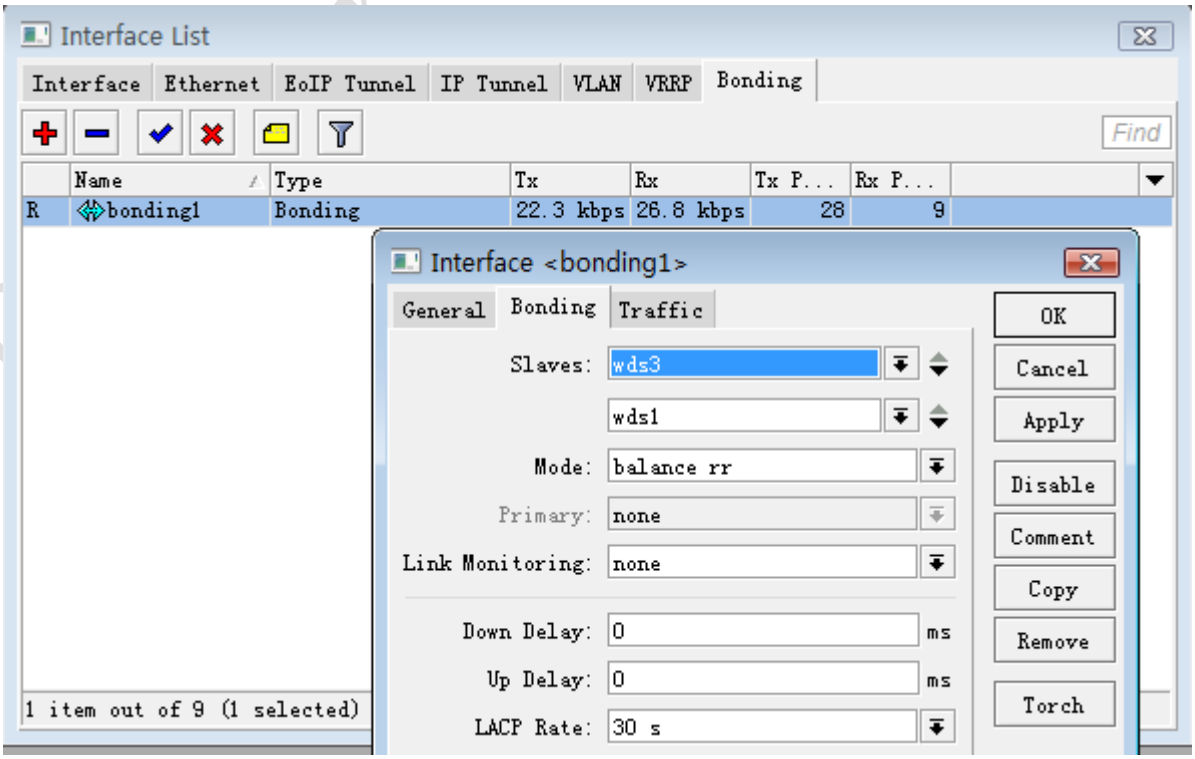
Station 連接狀態：



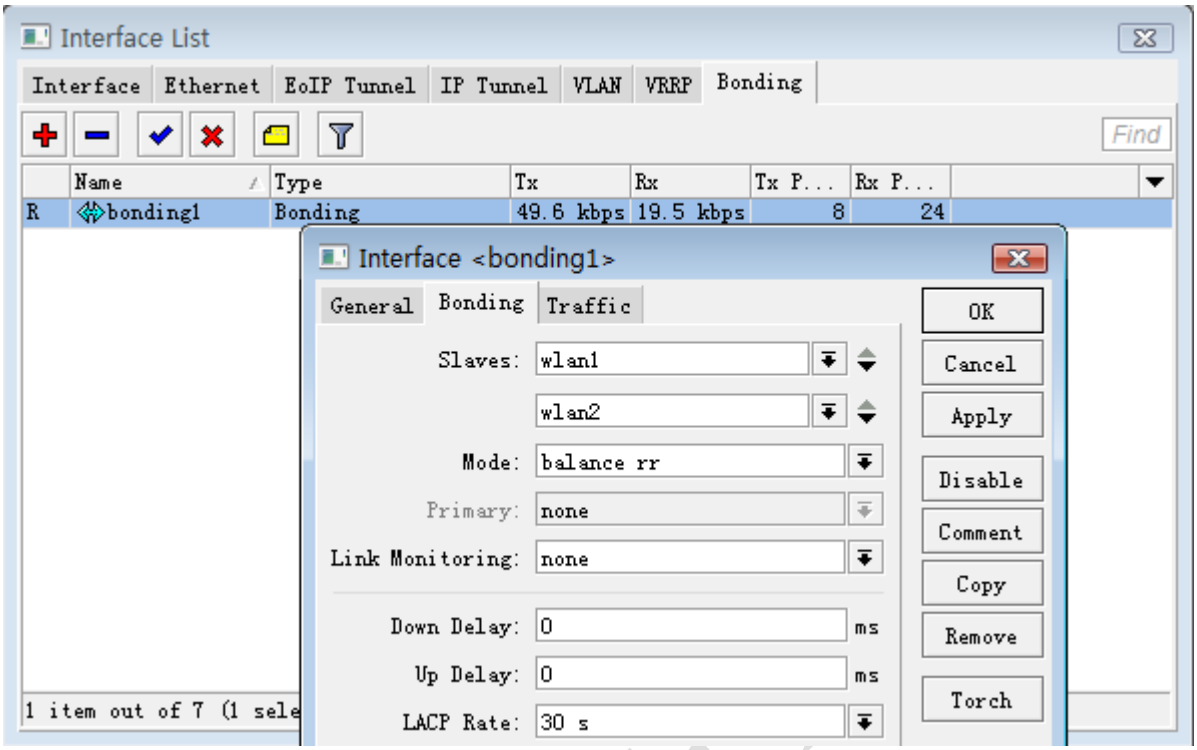
Bonding 配置

首先我們在 RouterOS 中添加 bonding，在 interface 中添加 bonding：

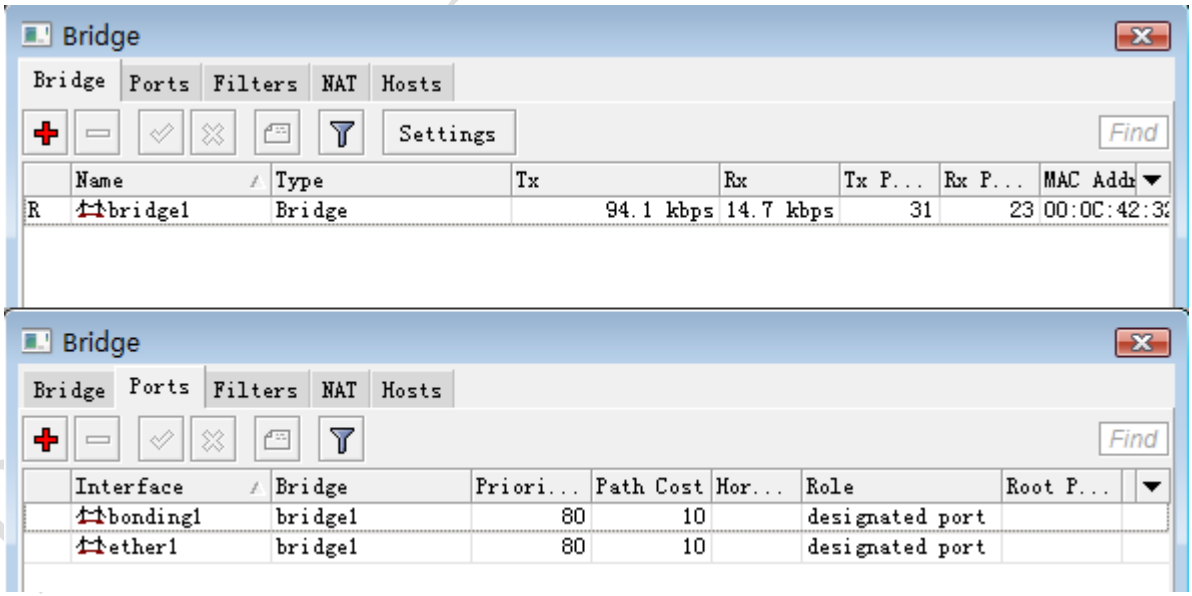
AP 端添加 bonding 時候，需要注意，綁定的不是 wlan1 和 wlan2，而是其自動增加的 wds 介面，這裡分別是 wds3 和 wds1：



配置 station 端，直接添加 wlan1 和 wlan2：

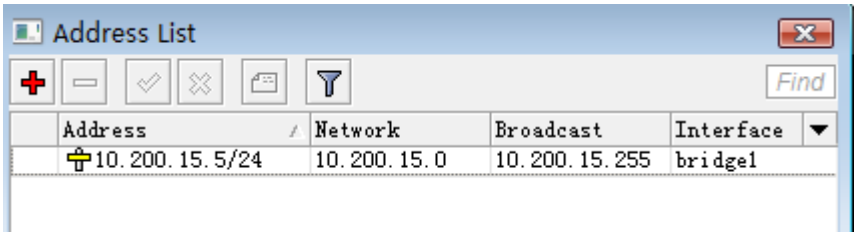


這樣完成了 bonding 的參數，我們需要將綁定好的兩組無線模組做成橋接模式，所以我們需要新建立 bridge 介面：

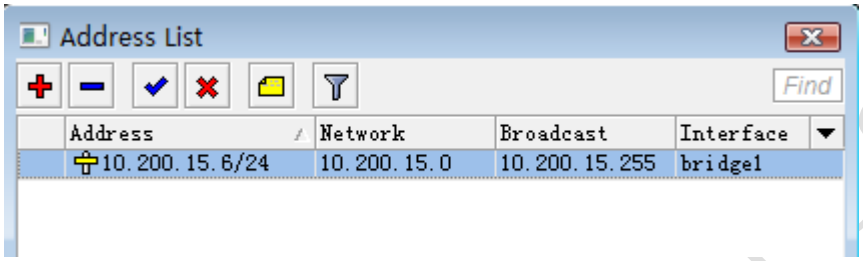


最後分配給 AP 和 Station 配置管理的 IP 位址到 bridge 上：

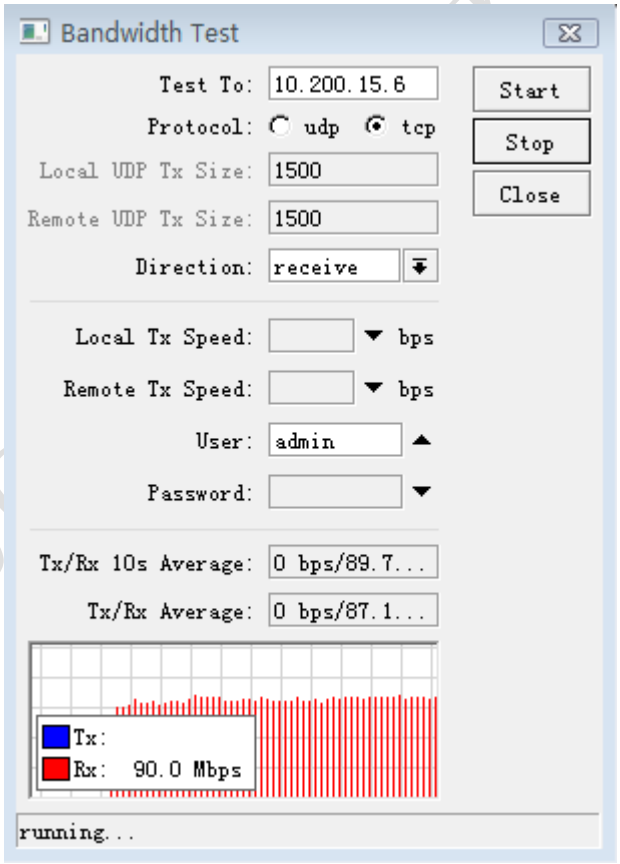
AP 的 IP 為 10.200.15.5



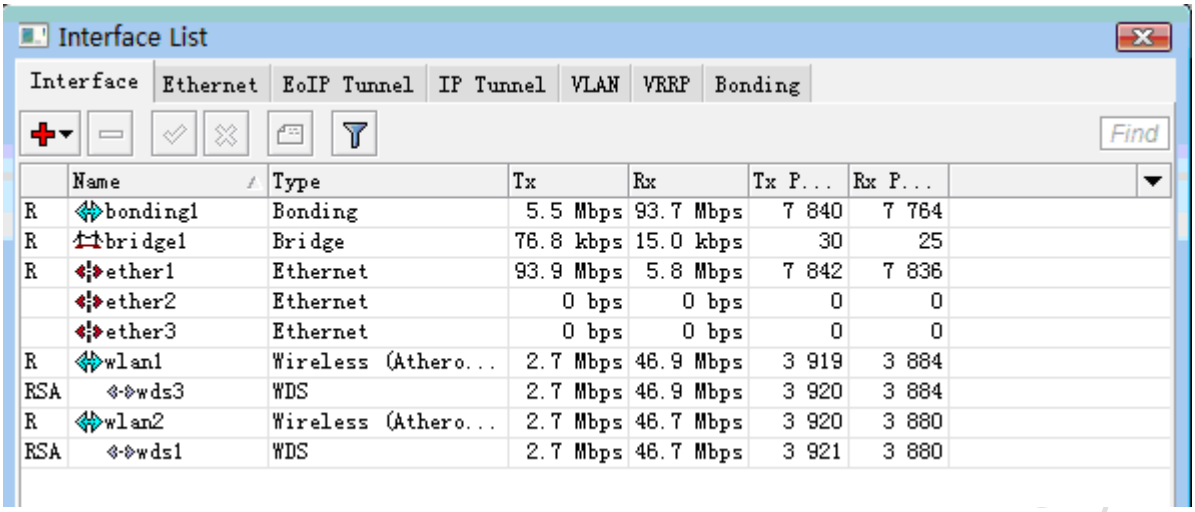
Station 的 IP 地址為 10.200.15.6



配置完 bridge 的橋接參數，bonding 操作就完成了，我們可以通過連接一台 PC 測試 bonding 後的頻寬：



Bonding 模式會以最低頻寬的無線模組為標準，即如果 wlan1 連接頻寬為 60Mbps，而 wlan2 連接頻寬為 40Mbps，則兩個連接頻寬合併應以 wlan2 為准，即 40Mbps*2=80Mbps。頻寬的高低和信號強度有關，如果要獲取高頻寬，需要將信號強度調整到至少-60dBi 以上的信號。



在使用 bonding 模式時候，需要注意設備的 CPU 耗用，如果頻寬沒有達到理想的水準，可以通過提升 CPU 性能，如 RB600A 在運行 2 組無線模組的 bonding 時候，CPU 達到 100%，也就是說 Bonding 對 CPU 消耗非常大，需要考慮更好性能的設備。

7.8 Nstreme Dual 協議與 Bonding

Nstreme Dual 協定與 Bonding 協定都是為提高 WLAN 無線頻寬而設計的，但他們有著不同的應用，通過下面的列表對比下：

功能/協定	Nstreme Dual	Bonding
支援無線模組	2 個模組，一個模組接收，一個模組發送	支援 2 個或 2 個以上模組，並且實現二層網路資料的彙聚
特點	上下行資料分開傳輸，提高網路雙向傳輸效率，CPU 耗用較高	將多個無線模組彙聚，增加無線網路頻寬，CPU 耗用特別高
設備需求	只能基於 1 對設備，完成協定	支援 1 對或多對設備組成，完成協定構建，在頻寬需求〉100Mbps 情況下，需要多對設備完成，且需要高性能 PC 彙聚頻寬
頻寬	雙向理想環境最大：65Mbps/65Mbps	單向理想環境最大（受設備組成數量、網卡速率、CPU 效率等）：〉100Mbps
應用範圍	通過 1 對設備，提高雙向傳輸的效率	對需要獲取高頻寬的環境

在更多的場合下 Bonding 模式更受歡迎，但對 CPU 耗用非常高。但隨著 Nv2 協定的出現，Nv2 協定配合 802.11n 協定能得到更高的傳輸頻寬，且處理器耗用更低。

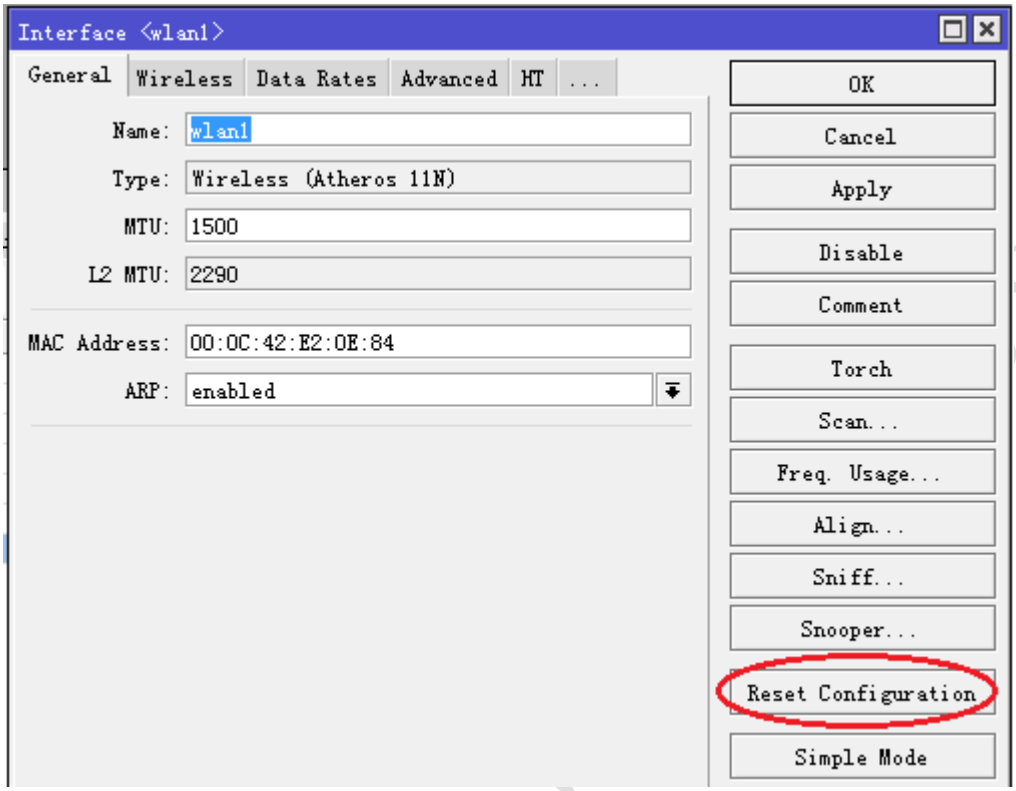
我們也可以使用基於 802.11n 協議的 Bonding 功能，即綁定 2 張 802.11n 的無線網卡，如果 1 張 802.11n 的無線網卡頻寬是 120Mbps，2 張就是 240Mbps，不過這需要使用性能更高的設備如 RB800。

7.9 RouterOS 無線配置參數 FAQ

修改了一些無線網卡參數，造成連接變動的不穩定。

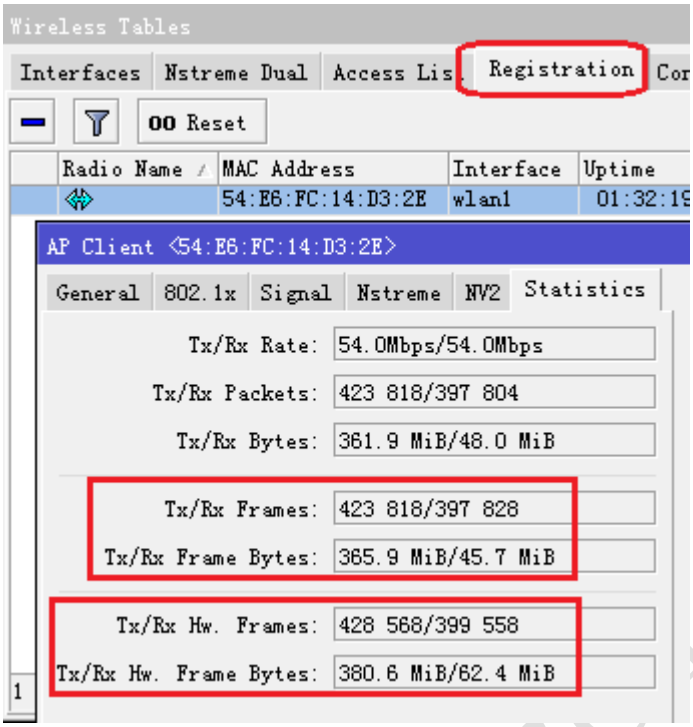
有時當你為了調整或優化一些連接修改了一些無線設置參數，但你又忘記了曾經設置過的參數，在

這樣的情況下你可以在 wireless 配置功能表下使用 reset-configuration 命令，該命令會重定該無線網卡的設置參數到初始化狀態。注意這個命令也會禁用該無線網卡，所以請你小心使用，特別是在通過無線連接到設備情況下調整



什麼是無線重發（wireless retransmits），並如何查看?

無線重發是當網卡發出一個幀，並沒有收到來至對端確認（ACK），你再次發出幀直到你收到對端的確認（ACK）。無線重發會增加延遲和降低輸送量。如果無線連接存在無線重發情況，你需要比較兩個欄位，在 registration table: **frames** 和 **hw-frames** ，如果 *hw-frames* 值大於 *frames* 即無線連接正在重發資料。這兩個值相差不大，可以被忽略。但如果 *hw-frames* 有 2、3 或者 4 倍大於 *frames* ，你需要對無線連接進行故障排查。



我是否能在 Nstreme 連接下比較 frames 與 hw-frames?

frames 只是統計那些包含實際資料的內容。在 Nstreme 方式中，如果沒有其他資料，僅 ACK 會能被一個獨立的幀傳輸，ACK 幀是不會被添加到 **frames** 統計中的，但他們會出現在 **hw-frames** 如果雙向傳輸達到最大速度（即，不會有 ack 幀情況），這時你不能比較 **frames** 與 **hw-frame**

如何設置發射功率（TX-power）?

tx-power 預設的最大 tx-power 值是無線網卡能使用的，且是從 EEPROM（電可擦唯讀記憶體（Electrically Erasable Programmable Read - Only Memory））獲取的值，如果你想使用更大的 tx-power 值，你可以設置該參數，但這個操作**需要由你自己負責**，很可能造成你的無線網卡的損壞！通常，使用這個參數來減低 tx-power。

通常 tx-power 控制應選擇默認設置，修改預設的設置需要參考無線網卡的情況，但如果沒有測試的情況下，大多可能造成距離和輸送量的減低，可能會發生一下的情況：

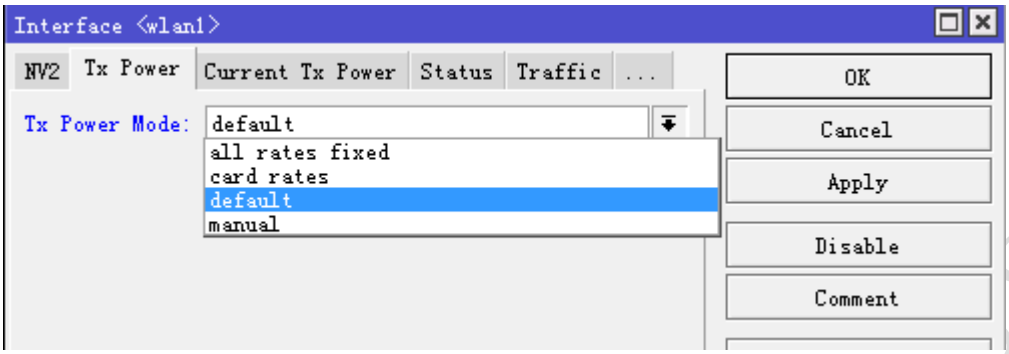
- 功率放大晶片過熱，網卡將會出現低效率，並出現更多的資料誤差；
- 放大晶片超負載，這樣的情況出現更多的資料誤差；
- 無線網卡過大的功耗，可能超出 3.3V 的供電，導致電壓驟降，並出現設備重啟或主機板溫度過高

什麼樣的 TX-power-mode 更好?

TX-power-mode 告訴無線網卡選擇哪一種被使用，默認選擇為 *default*.

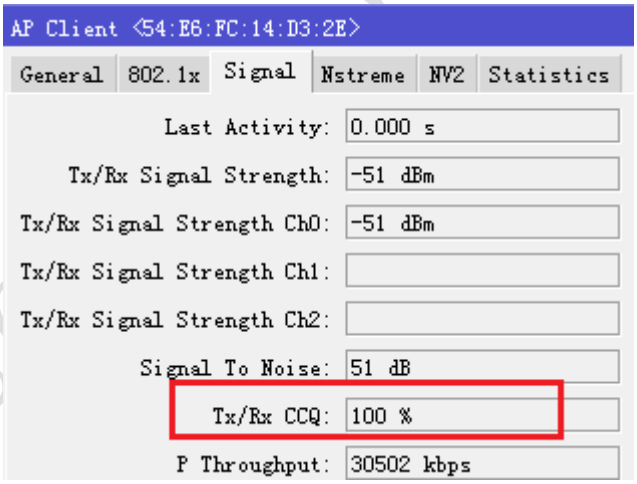
- **default** 即 tx-power 值從無線網卡的 EEPROM 中讀取，並將忽略用戶在 ***tx-power*** 裡這種的參數
- **card-rates** 即通過用戶指定 **tx-power** 值，並從無線網卡的 EEPROM 中獲取的資訊根據傳輸功率演算法計算出不同速率下的 **tx-power** （速率越高功率越低）
- **all-rates-fixed** 即由使用者指定無線網卡將在任何速率下使用相同的 **tx-power** 值

注意：不建議使用'all-rates-fixed' 模式，因為無線網卡 tx-power 在高速率下會降低，如果強迫 tx-power 使用固定值，結果是在高速率下仍然會出現為修改之前的 tx-power 情況，甚至可能對網卡造成損害。在大多情況下，如果你想修改 tx-power 建議選擇 ***tx-power-mode=card-rates*** ，並建議不要增加 tx-power。



什麼是 CCQ，且如何判斷這個值對無線網路的影響？

CCQ 用戶端連接品質（Client Connection Quality）是一個百分比數值，顯示有效的傳輸頻寬，CCQ 值加權平均數 Tmin/Treal，他們通過每次傳輸的幀計算得出，Tmin 是多次在高速率下沒有重新發送要傳送的幀，Treal 是多次在實際情況下傳輸出的幀 ，即我們認為的**理論值**除以**實際值**，我們把 Tmin 看成理論值，即我們要多次傳輸的幀，把 Treal 看成實際發送幀的情況（當然實際情況會受到環境影響造成幀的多次重發），如果重發次數多了 Treal 就會大於 Tmin， 如果 CCQ 是 100%，說明理論發送和實際發送相等，傳輸鏈路正常，但當鏈路出現問題時，Treal 會大於 Tmin，那百分比就會降低。



什麼是 adaptive-noise-immunity 設置?

Adaptive Noise Immunity (ANI) 自我調整噪音免疫，即動態調整各種接收參數減低干擾，這個設置被添加到 Atheros AR5212 無線網卡和更高的網卡晶片中。

RouterOS 無線支援什麼樣的錯誤校正方式?

ARQ 方式能被 Nstreme 協定支援。普通的 802.11 標準協定不包含 ARQ - 損壞的重新發送的幀基於 ACK 協議。RouterOS 支援正向糾錯與編碼速率： 1/2, 2/3, 或 3/4.

增加一個放大器是否會提高我的無線連線速度?

這個需要考慮你的信號品質和噪音情況。記住你要獲得一個好的無線鏈路，需要的是較低的發射功率和好品質的天線，有時候盲目使用放大器會增加噪音和造成無線鏈路的各種故障。

放大器會提高傳輸和接收的信號，因此在一個“安靜”區域，僅你一個無線設備，且很少的噪音和其他無線設備，結果是你可以得到很好的效果。但另外一種情況，“擁擠”的區域，很多無線設備在工作，你將會增強信號，也同樣會增多來至其他設備或者周邊噪音源的信號，可能引起整體信號品質的降低。

你通常從 11b 無線信號上獲得較好的信號，在 802.11g 協定下會產生更多的噪點，因此需要對有用的信號進行過濾。

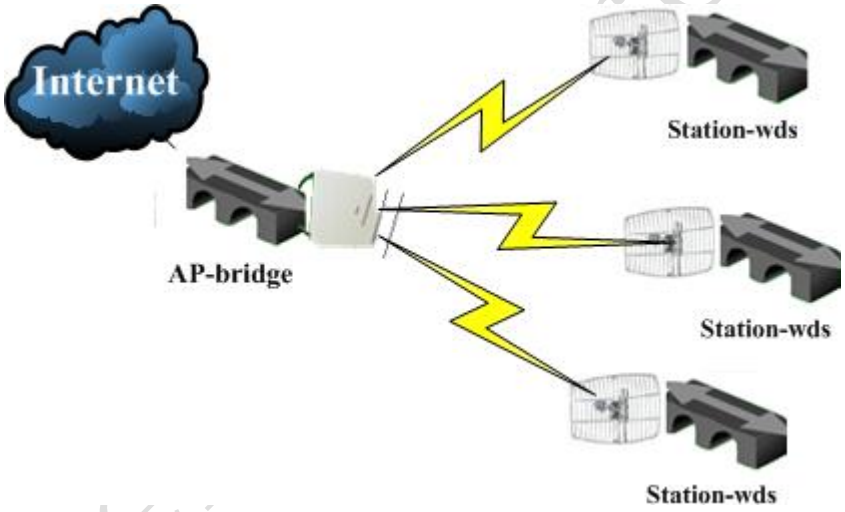
第八章 WLAN 點對多點與中繼

在大型區域和城區的大型的無線網路構建中，會應用到更多中繼和點對多點等方式的無線傳輸，在 Wlan 無線中繼和點對多點的無線傳輸有以下兩種事例，橋接和路由兩種網路構建模式：

- 橋接模式：通過在設備上建立 bridge，在 Wlan 無線網路中實現二層資料連結的傳輸，這種方式適合於小型的無線網路建設。
- 路由模式：通過在設備上建立多級 IP 位址路由，通過傳輸 IP 資料包連接各個設備和終端。這樣的模式更適合於大型的無線網路組建。

8.1 橋接模式的點對多點

橋接模式的點對點與中繼，適用用於二層資料的透傳，對於網路組建和網路傳輸協定更改較容易，橋模式透傳適合於小型的無線網路。基於點對點 WDS 橋模式，即 1 個中心 ap-bridge 設備通過 1 個大扇角平板天線覆蓋周圍的區域，遠端多個 station-wds 設備通過定向天線進行連接，如下圖：



中心點配置為 ap-bridge 的橋接，其他 3 個終端點配置為 station-wds 模式的橋接，這樣的配置和普通 AP-Bridge to Station-WDS 模式完全一樣，只是增加了多個 station-wds 終端機站，即點對點傳輸的一種演變。下圖是一個雙網卡的點對多點的連接狀態圖，wlan1 和 wlan2 都分別連接了 3 個 station-wds 的終端設備：

Wireless Tables

Interfaces | Access List | Registration | Connect List | Security Profiles

	Name	Type	MTU	MAC Address	Mode	Band	Frequ...
R	wlan1	Wireless (Atheros AR5212)	1500	00:11:F5:4B:9F:A3	ap bridge	5GHz	5220MHz
RA	cuilingju	WDS	1500	00:11:F5:4B:9F:A3			
RA	nanguo	WDS	1500	00:11:F5:4B:9F:A3			
RA	rongbao	WDS	1500	00:11:F5:4B:9F:A3			
R	wlan2	Wireless (Atheros AR5212)	1500	00:FF:7C:B0:5F:0C	ap bridge	5GHz	5180MHz
RA	huatianguoji	WDS	1500	00:FF:7C:B0:5F:0C			
RA	xintuo	WDS	1500	00:FF:7C:B0:5F:0C			
RA	yangguang	WDS	1500	00:FF:7C:B0:5F:0C			

橋模式點對多點的特點：

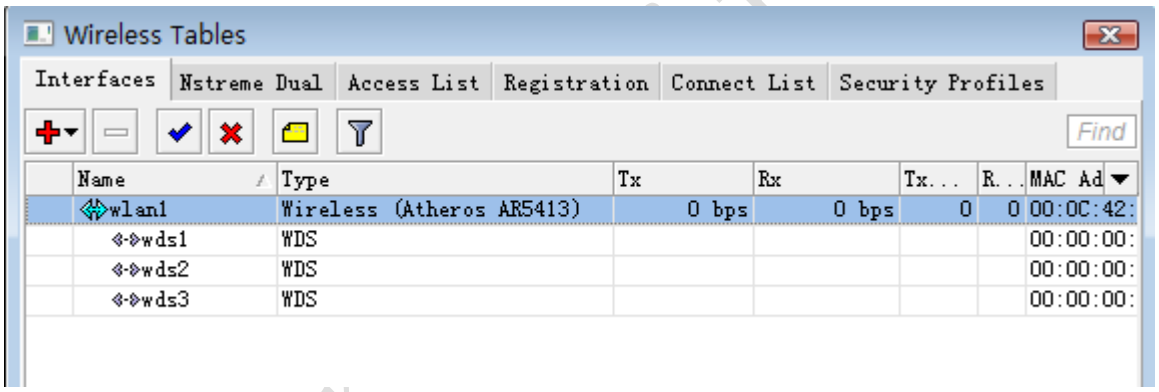
1、 在一個 90 度天線能夠覆蓋的扇形區域內有多個終端點，可以通過點對多點的方式連接。通常中心基站採用 5G 90 度的平板天線，終端設備採用拋物面的定向天線與中心連接，根據具體情況，能在 10-15 公里左右正常連接。

2、 橋模式的點對多點架設和配置簡單，新增節點無需增加或者修改網路參數，適用於 PPPoE 撥號認證和各種三層網路的傳輸認證。

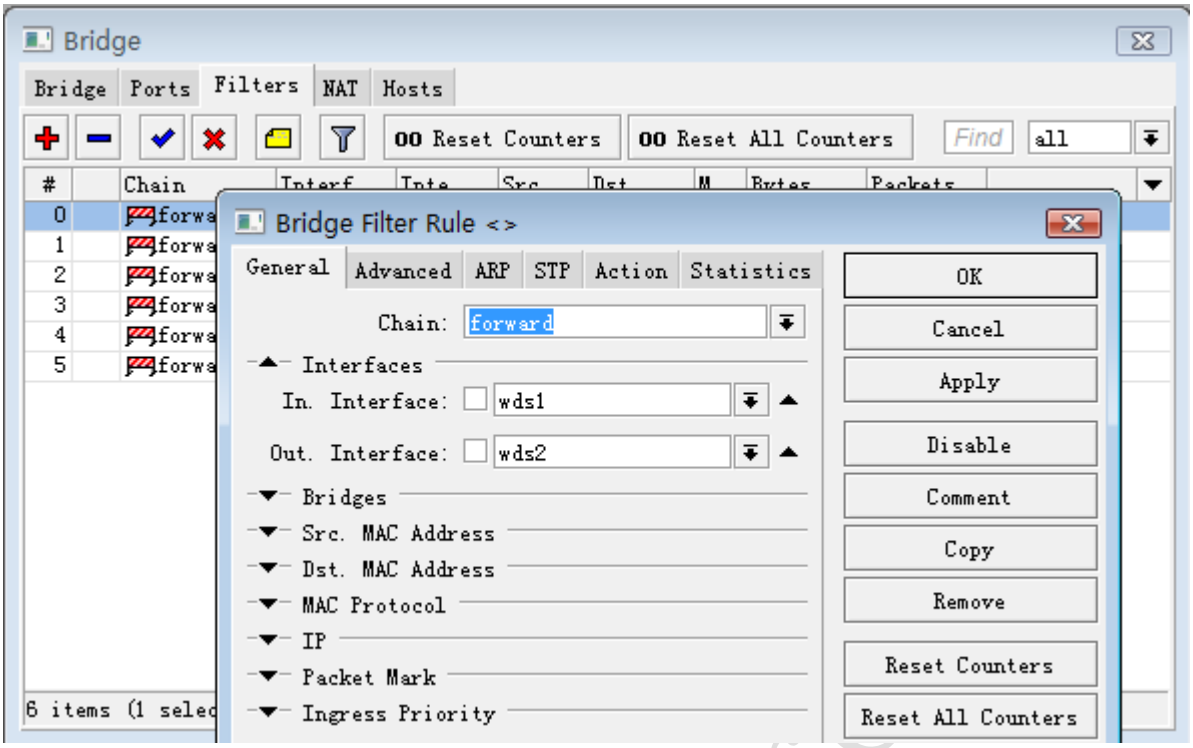
橋模式點對多點的缺點：這樣的點對多點，是通過橋模式連接，所以二層廣播資料會在各個基站之間相互傳輸，降低網路效率，特別是在網路不斷擴大的情況下。通常我們可以通過 bridge filter 來限制各個 wds 介面的廣播資料。

8.2 橋接 WDS 的端口隔離

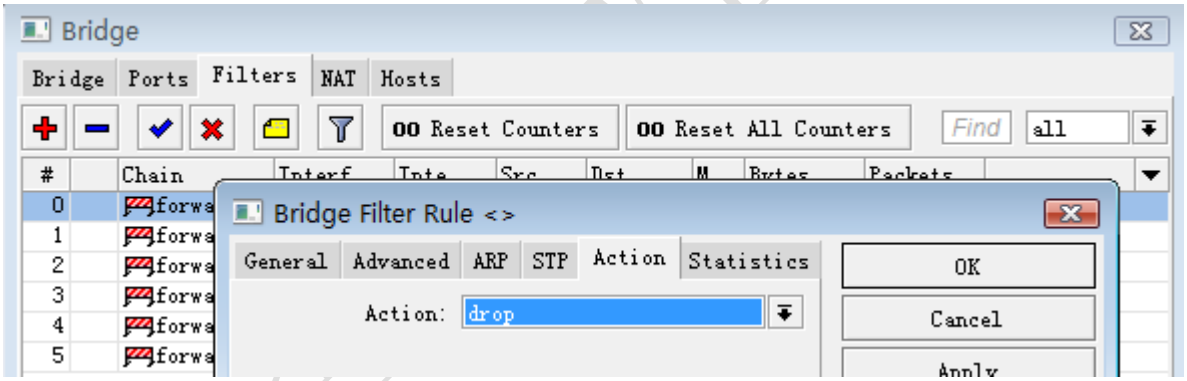
這裡我們假設有一個 wlan1 通過點對多點的橋接連接了 3 個 station-wds 的終端設備，分別是 wds1、wds2 和 wds3。由於 3 個 wds 橋接設備之間會互相廣播二層資料，甚至某些客戶中了 arp 病毒也會通過 wds 橋接攻擊到其他網路的使用者，我們需要通過/bridge filter 隔離 wds1、wds2 和 wds3 的數據。



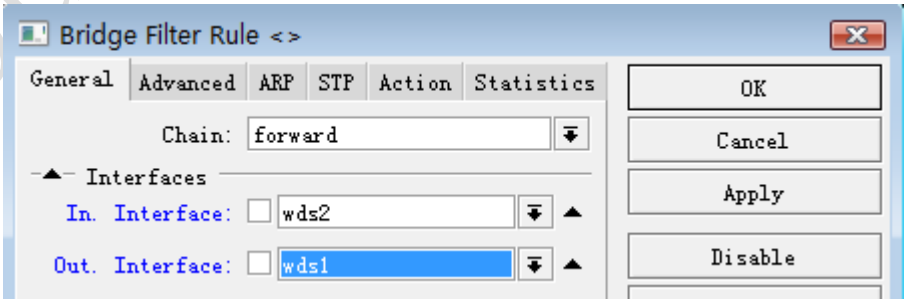
我們進入 bridge filter，通過添加規則，設置 in-interface（資料進入介面），out-interface（資料發出介面），如下圖資料進入介面是 wds1，發出介面是 wds2，由於是通過橋轉發的，所以我們選擇 chain=forward：



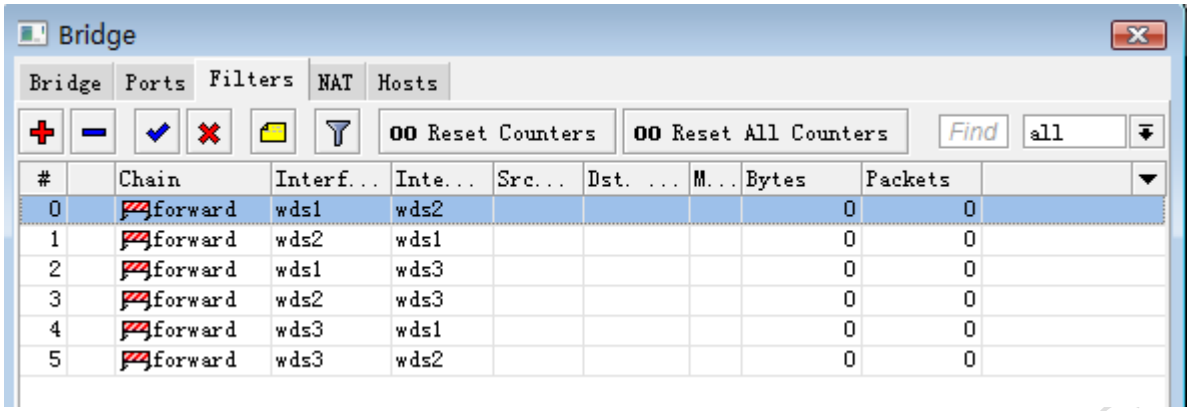
接下來我們配置 action=drop 丟棄他們之間轉發的資料：



由於資料傳輸是雙向的，所以我們需要反向填寫 in-interface 和 out-interface，同樣需要將 action 設備為 drop，如圖：



一共有 3 個 wds 介面，每對介面需要配置 2 條規則，所以一共需要配置 6 條過濾規則限制資料廣播：



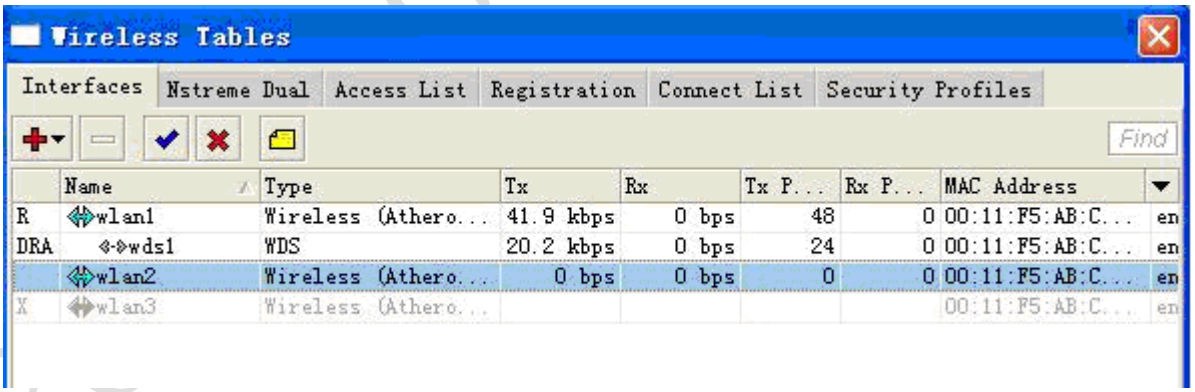
通過以上配置可以限制多個 wds 介面之間的資料廣播，一定程度上提高 wlan 的無線網路傳輸效率，但不能根本性的解決。（關於更多的 bridge 過濾規則，請參考**《RouterOS 中文網路教程》**）

8.3 橋接模式的中繼

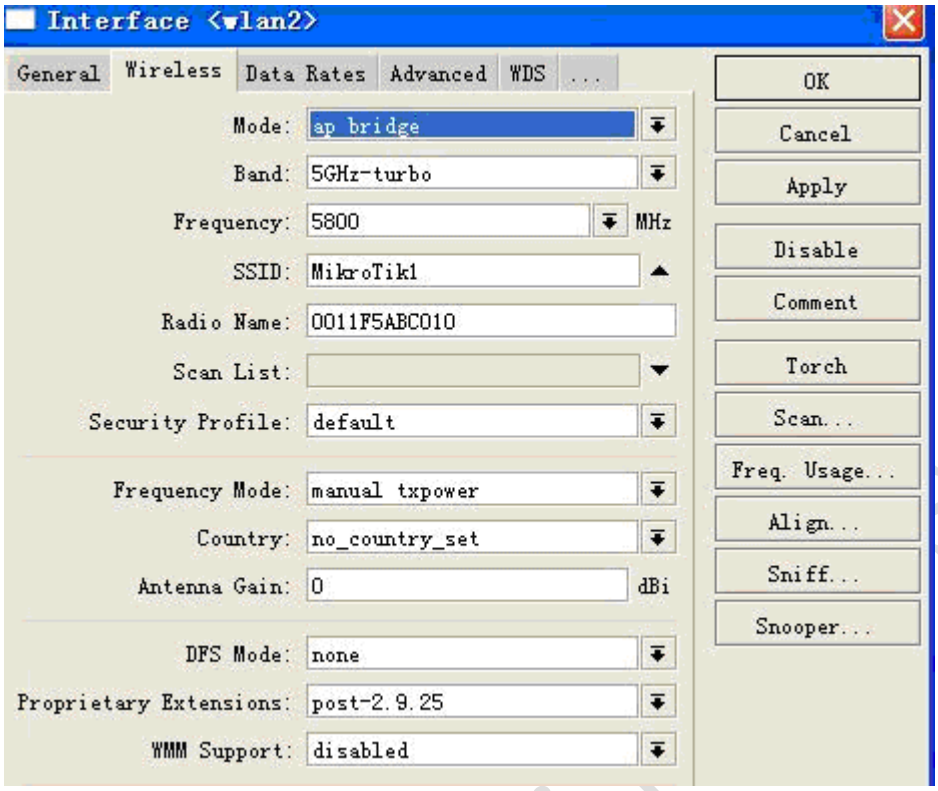
wlan 無線中繼是指在一個設備上添加兩張或兩張以上的網卡，做中繼傳輸，他和點對點或點對多點不同的地方在於，將 2 個周邊的網路進行資料中轉，這樣的情況會有如下：

- 兩個基站之間由於視距阻擋，不能直接連接需要採用中繼的方式；
- 兩個基站之間由於距離太遠，信號衰減較大，需要中繼放大；
- 兩個基站之間由於其他特殊原因無法直接連接。

將兩個無線連接做透明的橋接傳輸，這裡我們用兩張網卡做事例，我們在原有的設備上增加了一張 wlan2 的網卡，與 wlan1 做橋接實現中繼：

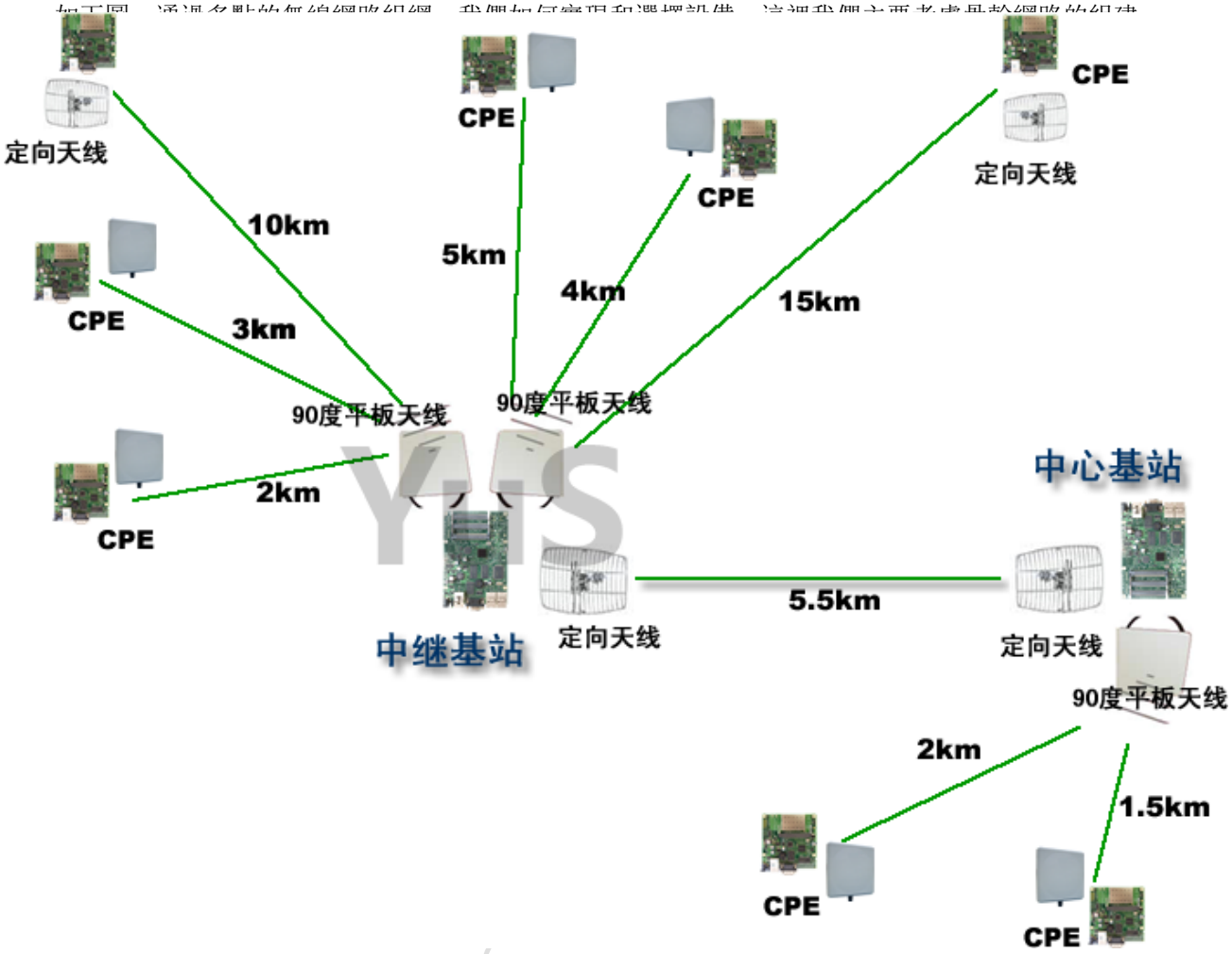


我們設置 wlan2 的參數為 Mode 為 ap-bridge，Band 為 5G-turbo，頻率為 5800，SSID 為 MikroTik1



在 bridge1 中添加 wlan2 的介面，同樣在 WDS 選項我們把 WDS Mode 為 dynamic（動態方式），WDS Default Bridge 為 bridge1，這樣 wlan1 與 wlan2 被系統自動添加到 bridge1 中，中繼的橋接設置就實現了，其實配置和普通的 wds 模式橋接完全一樣，只是將 2 個無線模組都放到一個 bridge 中。

8.4 橋接模式點對多點和中繼的綜合應用



所有的 Wlan 無線設備採用 5G 頻段做為骨幹傳輸，5G 頻段干擾小，傳輸效率穩定，設備的結構組成如下：

終端點 CPE：全部採用 RB411 或者 RB711 型號，配置 5G 定向拋物面天線，用於遠距離的接收。最高能提供 45Mbps 的 TCP 頻寬，如果說 RB711 採用 11n 的 Nv2 協議可以達到 93Mbps。

中心基站：為資料接入的中心節點，這個點為了得到更好的資料和傳輸，我們採用 RB433（300MHz 處理器）或 RB433AH（680MHz 處理器）型號的無線設備，如果考慮高處理能力可以考慮 RB800，全千兆設備。因為該點需要和中繼點和兩外兩個終端接入點連接，所以採用雙網卡發射信號，同時支援 2 個 5G 信號輸出，分別通過一個定向天線與中繼點連接，一個用 90 度的平板天連接兩個終端點 RB411 或者 RB711。

中繼基站：採用 RB433AH 型號設備，安裝 3 張無線網卡，配置一個連接信號點的 RB433AH 的設備。另外兩張連接 2 個 90 度平板天線，對下面的 6 個點分別做點對多點的連接。

*注：*一個無線網卡可以支援 2007 個無線用戶端，但實際環境中可以連接的實際用戶在 20-30 個，因為當用戶端多後會形成相互的無線信號干擾。而這裡我們採用一個平板連接 3 個用戶端，原因是保證每個用戶端的頻寬，因為每個用戶端都會平分頻寬，在每個用戶端的信號強度和環境適宜，中心點能提供最大 60Mbps 的頻寬情況下，3 個用戶端每個可以獲取 20Mbps 的頻寬。

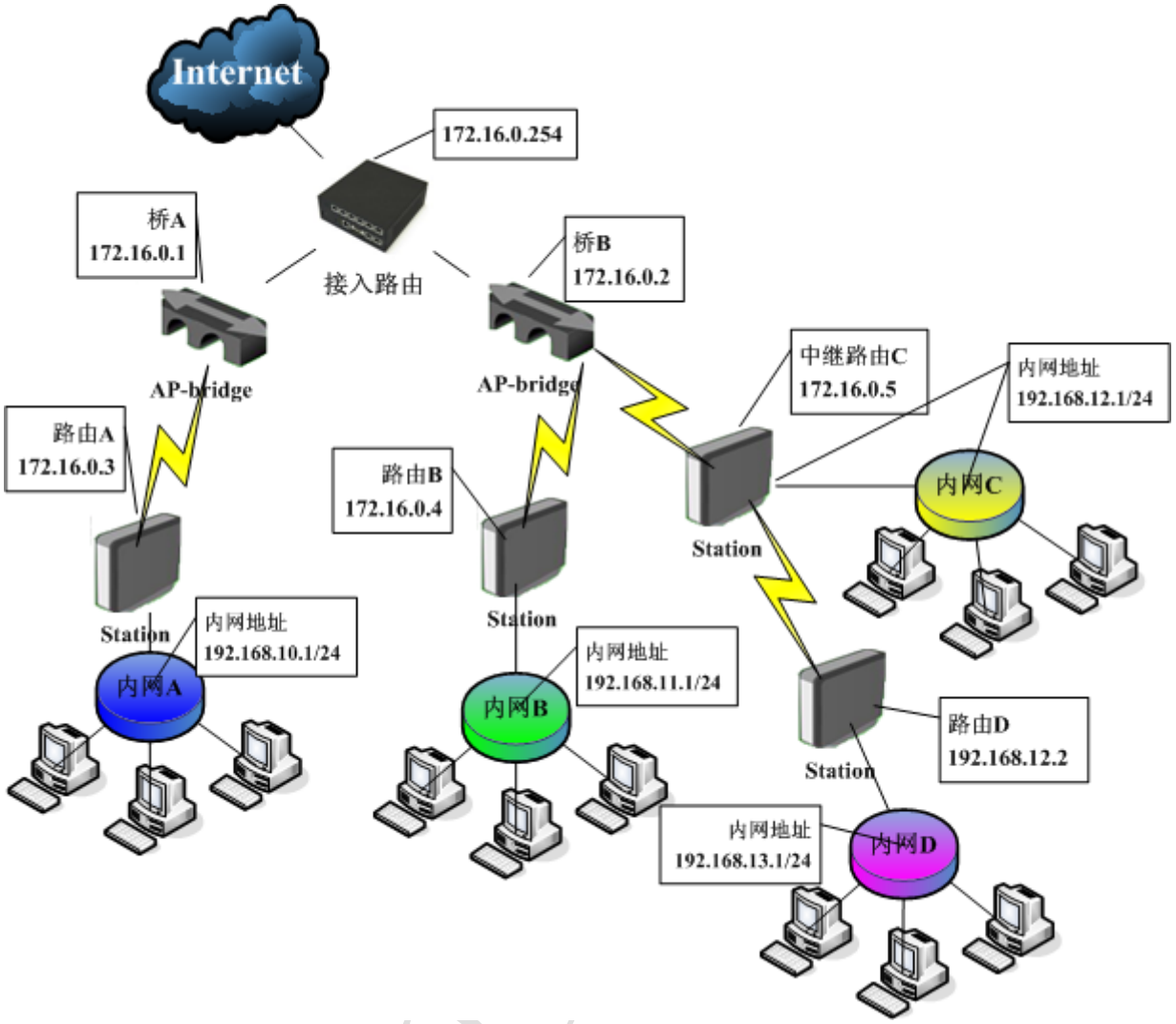
*注：*在點對多點的方式下同樣採用橋接的方式，中心 AP 的設置和點對點的 AP1 設置是完全一樣的，至於遠端的多個 station 客戶的配置也是和點對點的 AP2 一樣設置為 station-wds。

點對多點與中繼下可以選用的模式如下表：

模式	參數	Nstreme 協議	Nv2 協議
AP-Bridge to Station	路由模式，啟用橋接需要配置 EoIP	支援，取決於網卡速率和選擇模式，最高 108M	暫不支持
AP-Bridge to Station-WDS	橋接模式，自動添加到橋接設置中	支援，取決於網卡速率和選擇模式，最高 108M	支持，11n 支持近 200Mbps 頻寬
AP-Bridge to AP-Bridge	支援橋接和路由模式，橋接模式自動添加到橋接設置中，同樣支援 MESH 和 WDS 模式	不支援，取決於網卡速率和選擇模式，最高 108M	不支持
Bridge to Bridge	橋接模式，自動添加到橋接設置中	支援，取決於網卡速率和選擇模式，最高 108M	不支持，
bridge to station-bridge	橋接模式，自動添加到橋接設置中	支援，取決於網卡速率和選擇模式，最高 108M	支持，11n 支持近 200Mbps 頻寬

8.5 路由模式的 wlan 網路

路由模式的 wlan 網路構建要比 wds 橋模式的無線網路較為複雜，但適合於大型 wlan 無線網路的構建，特別是 WISP 的運營，這樣的模式通過多級路由實現，能將二層的廣播限制在一個路由範圍內，降低網路效率。如下圖的一個結構：

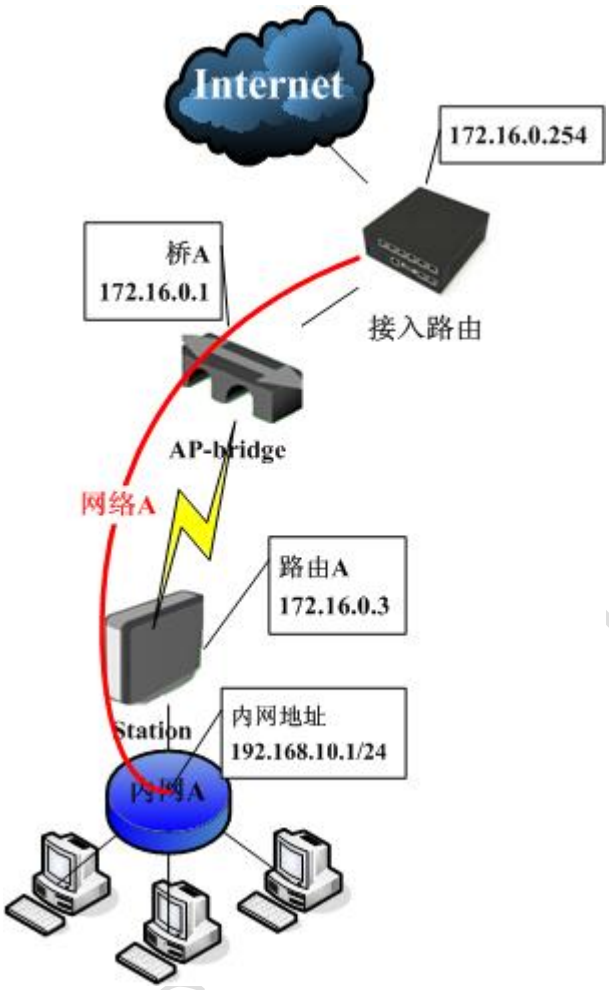


這樣的無線網路要求在接入路由和各級節點設備時配置路由表，隨著設備和網路的不斷擴大路由表的維護也同樣增加，這樣的網路構建我們通常使用 ap-bridge to station 模式，ap-bridge 採用透明橋連接，而 station 則是配置路由模式。這樣網路構建可以看作多個點對點的 ap-brdge to station 模式的組合模式。

在這樣的網路環境需要在接入路由器上配置路由表，首先我們通過分步驟來瞭解網路拓撲，

步驟 1：

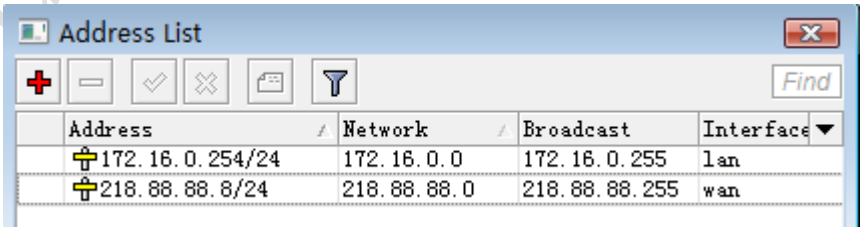
網路 A 的使用者需要訪問 Internet 就必須經過接入路由，這裡從接入路由開始一共有 3 個設備，分別是接入路由、橋 A 和路由 A，如下圖：



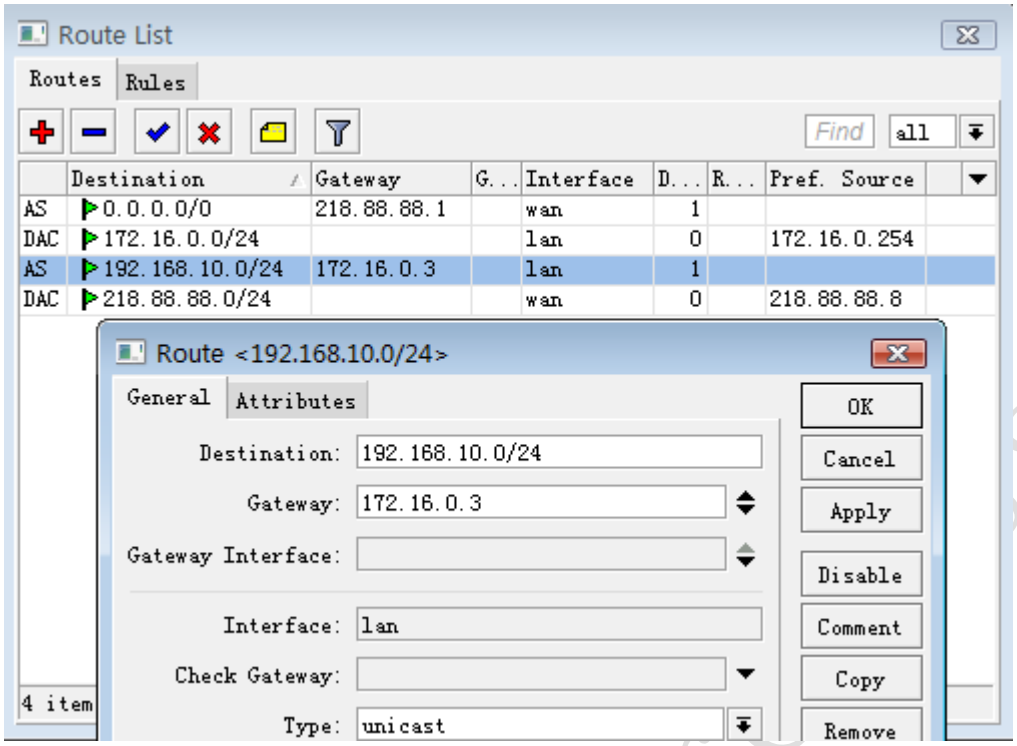
在這個網路中**橋 A** 是一個二層的橋接器設備，對於三層來說他完全是透明的，可以不用考慮路由問題，那在這個網路結構中，我們只需要考慮 2 個設備即**接入路由**和**路由 A**。在這兩個設備分別由 2 個網段組成，分別是 172.16.0.0/24 和 192.168.10.0/24，我們需要通過配置路由表讓資料能正常連接：

接入路由配置：

首先配置接入路由的 IP 位址，wan 口的外網連接 IP 位址是 218.88.88.8/24，內網連接用的 IP 位址是 172.16.0.254/24，在/ip address 配置如下：

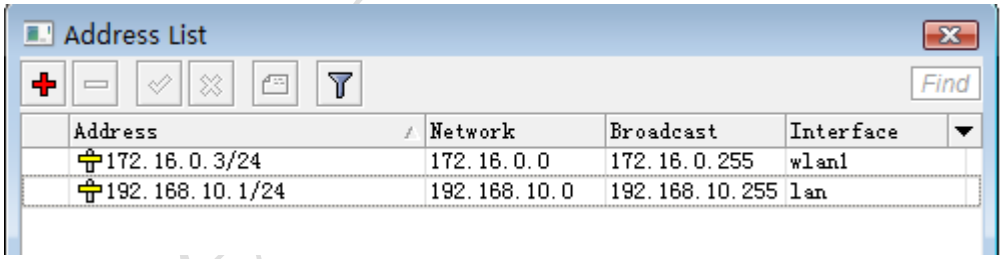


接下來我們需要配置，讓接入路由知道 192.168.10.0/24 這個網段需要經過那條線路可以到達，由於 192.168.10.0/24 網路在**路由 A：172.16.0.3** 後，即配置一條目標位址路由 dst-address=192.168.10.0/24 gateway=172.16.0.3：

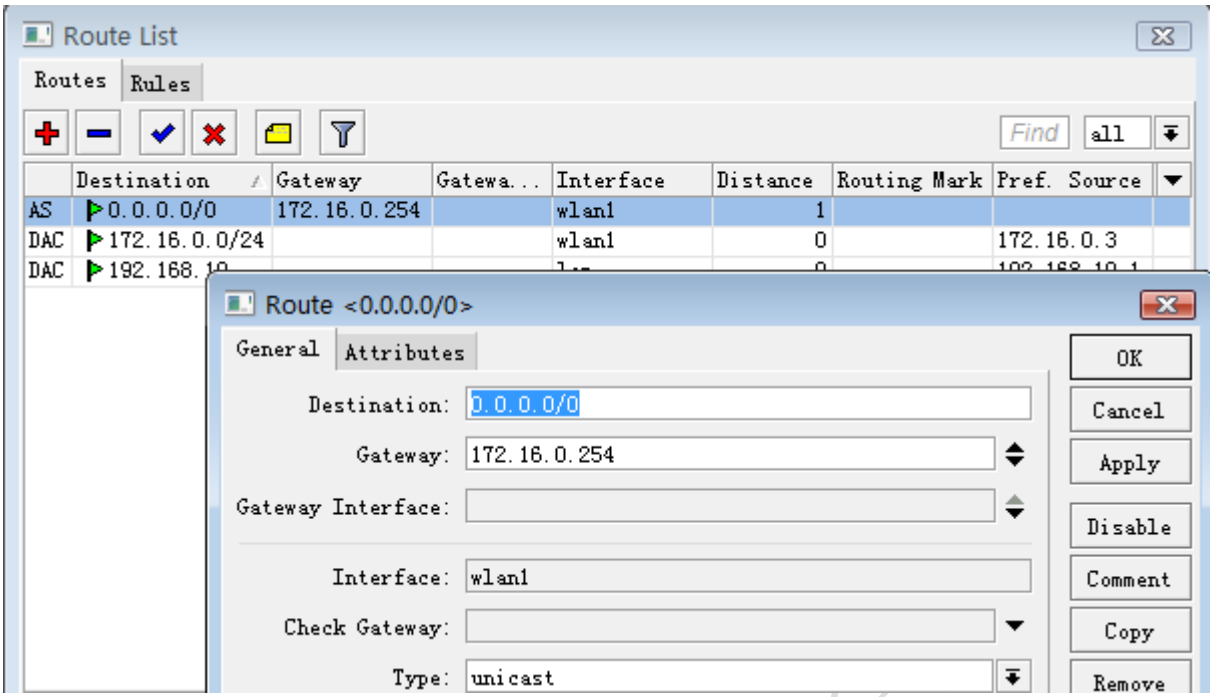


路由 A 配置：

路由 A 我們用 192.168.10.1 做為內部閘道的 IP 位址，連接接入路由的 IP 是 172.16.0.3，IP 位址配置如下：

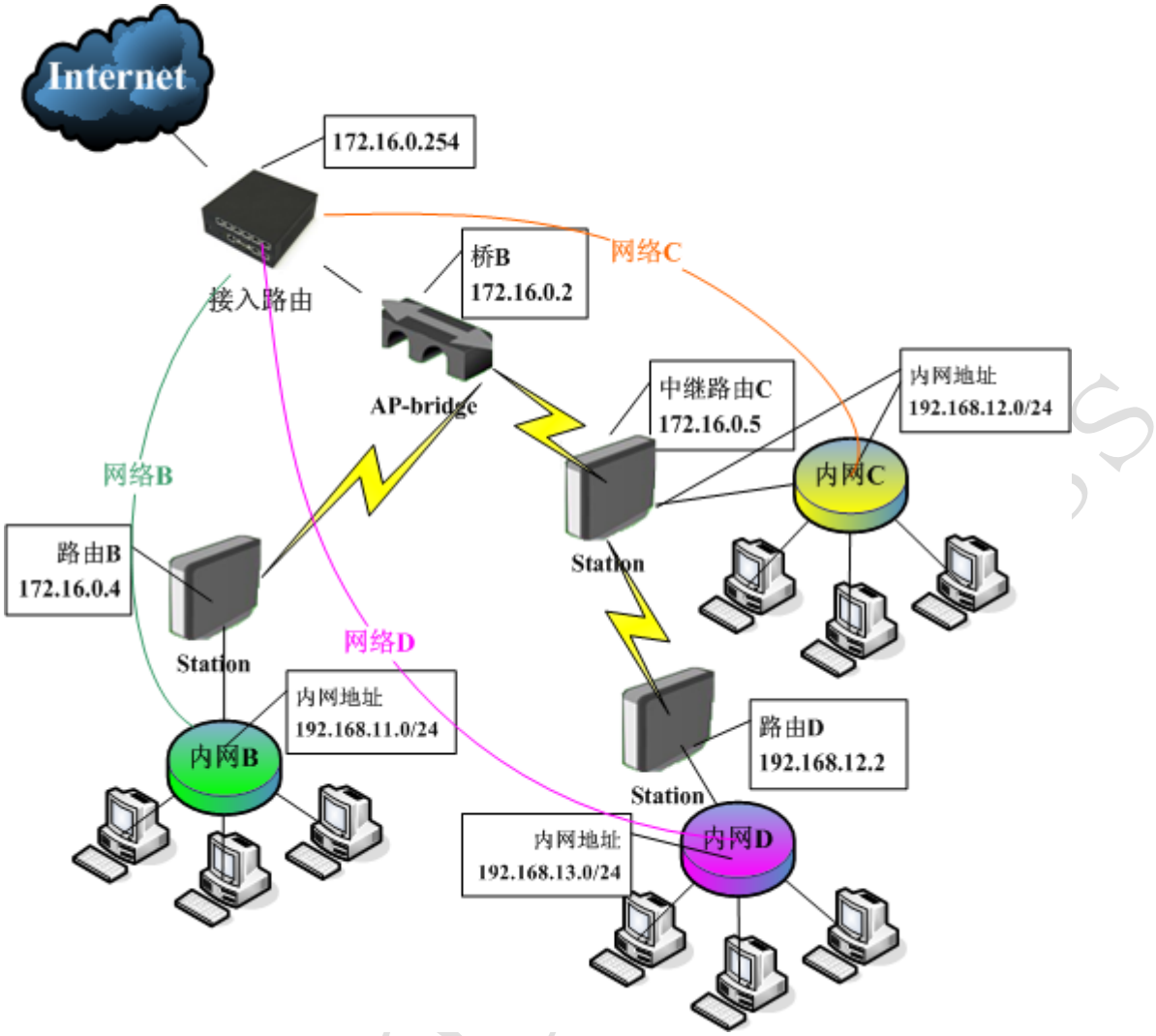


接下來配置路由 A 的閘道，因為只需要讓 192.168.10.0/24 內網路的使用者直接上網，我們可以直接將預設閘道器指向接入路由 172.16.0.254



步驟 2：

現在我們來看看**網路 C** 和**網路 D**，這裡同樣是要將兩個網路連接到 Internet，但這個網路比之前的步驟 1 要複雜一點，因為這裡是串聯了 2 級的網路，**橋 B** 通過點對多點分別連接了**路由 B** 和**中繼路由 C**，而且在**中繼路由 C** 下連接了**路由 D** 的設備，從接入路由開始一共連接了 6 台設備。如下圖：



網路中**橋 B** 為橋接器，看作透明設備，我們只需要配置**接入路由、路由 B、中繼路由 C** 和**路由 D** 的路由表，我們通過下面的表看看：

接入路由關於這三個網路的路由表

目標位址	閘道	介面	備註
192.168.11.0/24	172.16.0.4	Lan	指向路由 B
192.168.12.0/24	172.16.0.5	Lan	指向中繼路由 C
192.168.13.0/24	172.16.0.5	Lan	指向中繼路由 C（因為 192.168.13.0/24 段在中繼路由後）

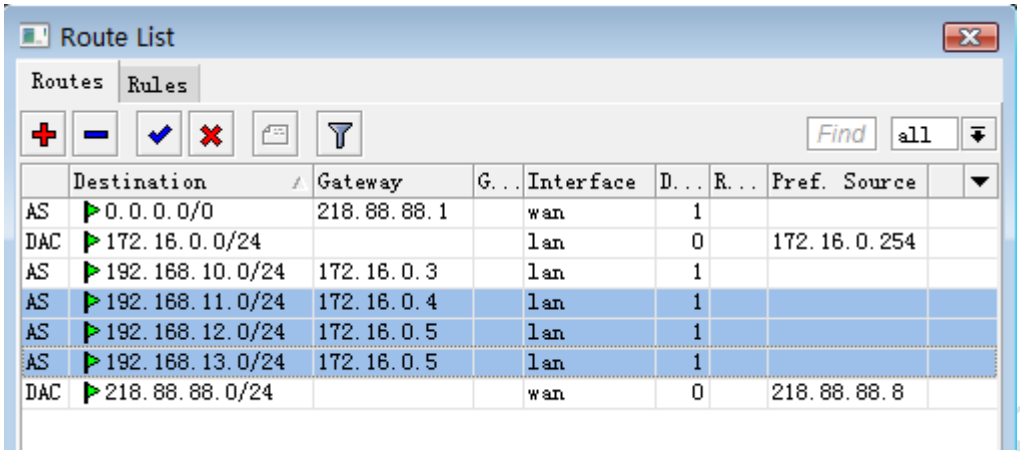
中繼路由 C 的路由表

目標位址	閘道	介面	備註
0.0.0.0/0	172.16.0.254	Wlan1	到接入路由的預設閘道器
192.168.13.0/24	192.168.12.2	Wlan2	指向路由 D

而對於**路由 B** 和**路由 D** 只需要將預設閘道器指向上一級路由的閘道 IP 即可，如步驟 1，因為他們是最後一級設備，不需要配置其他的路由表。

接入路由

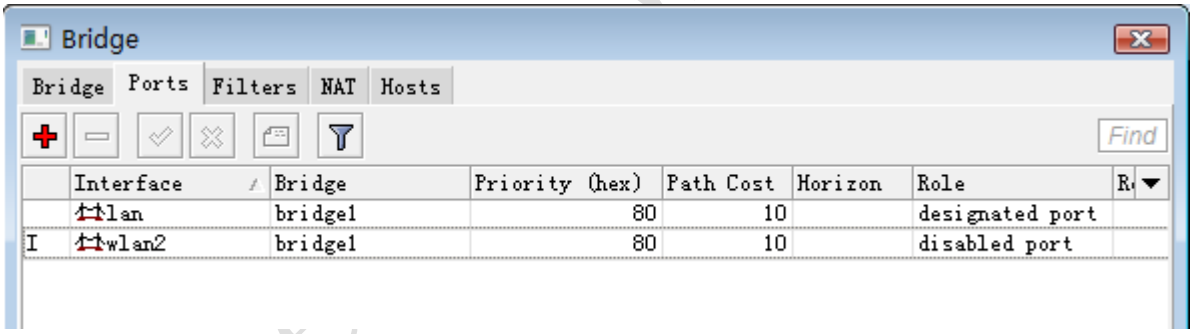
在接入路由上我們只需要配置三條目標位址路由，如下圖：



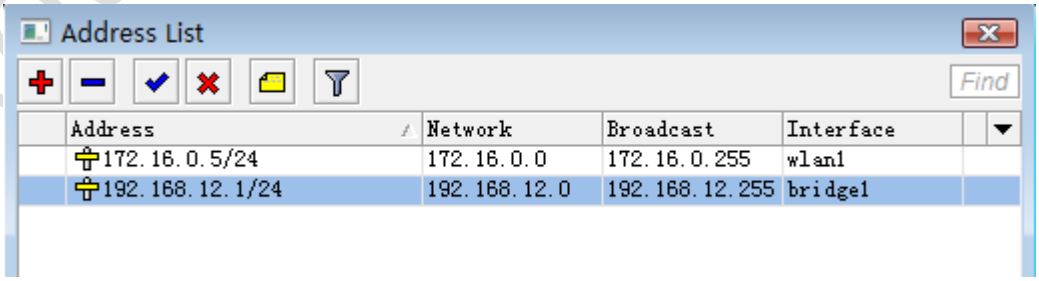
中繼路由 C

這裡我們增加了一個中繼路由，這個路由和以往的路由配置有點不同，中繼路由我們採用的 2 張無線模組，即 1 張用於連接**橋 B** 的網路配置為 station 模式，而另外一張即要和路由 D 連接配置為 ap-bridge 模式，又要和以太網卡做橋接。

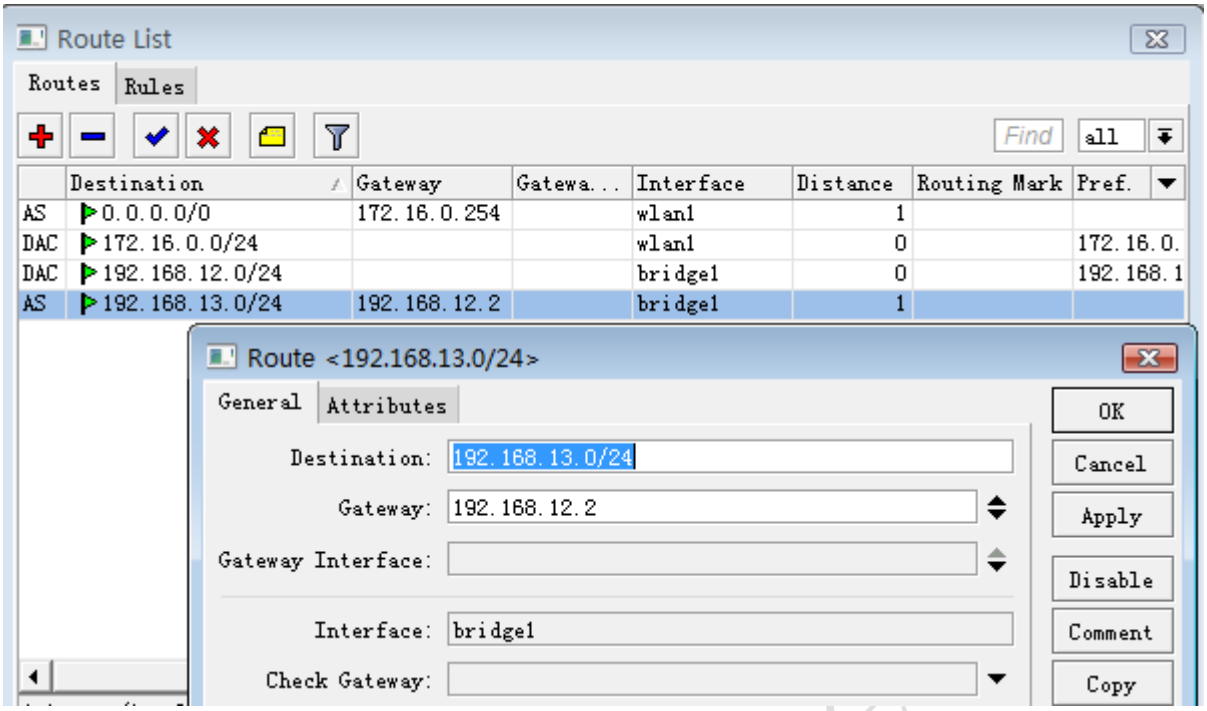
在中繼路由 C 中我們用到 2 張無線模組，1 個以太網卡，wlan1 用於連接橋 B，wlan2 用於連接路由 D，而以太網卡則和中繼路由 C 下的網路 C 相連接。由於 wlan2 和以太網卡屬於一個局域網內，所以我們將他們放到一個橋中，由於 wlan2 使用的是 ap-bridge 模式，所以支援橋接模式，添加橋後配置如下（無線部分請參考之前的 ap-bridge to station）：



接下來我們配置 IP 位址，中繼路由 C 的 wlan1 的 IP 位址為 172.16.0.5/24，內網連接路由 D 和網路 C 的位址是 192.168.12.1/24，我們將 wlan2 和 lan 口放到一個橋中，所以將內網 IP 設置道 bridge1 上：

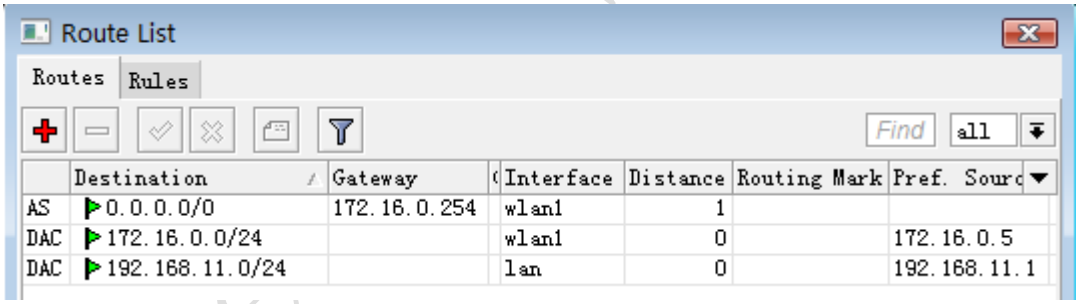


接下來我們配置路由部分，添加 dst-address=192.168.13.0/24 gateway=192.168.12.2：

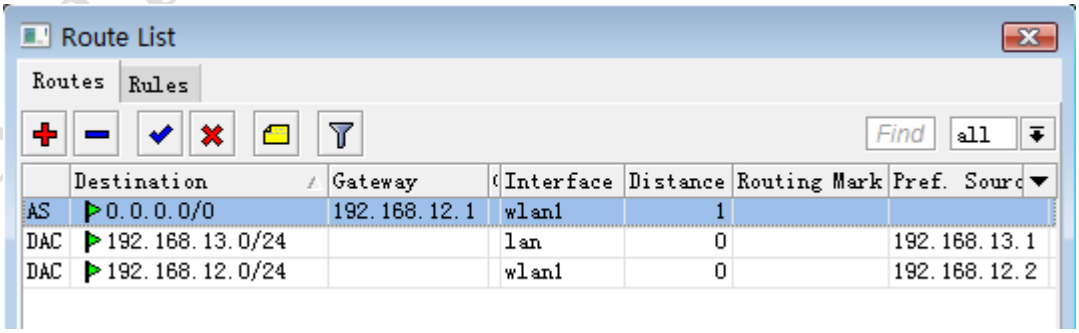


路由 B 和路由 D

添加完相應的 IP 位址後，路由 B 和路由 D 的配置和之前的路由 A 是完全相同的，只是路由 B 與路由 D 所指向的閘道不同，路由器 B 指向的閘道為 172.16.0.254，因為他的 wlan1 與接入路由在同一網段，如圖：



路由 D 因為在路由 C 的 192.16.12.0/24 的網段下，所以需要將閘道指向與他同一網段的 192.168.12.1



第九章 MikroTik Mesh 無線網狀網路

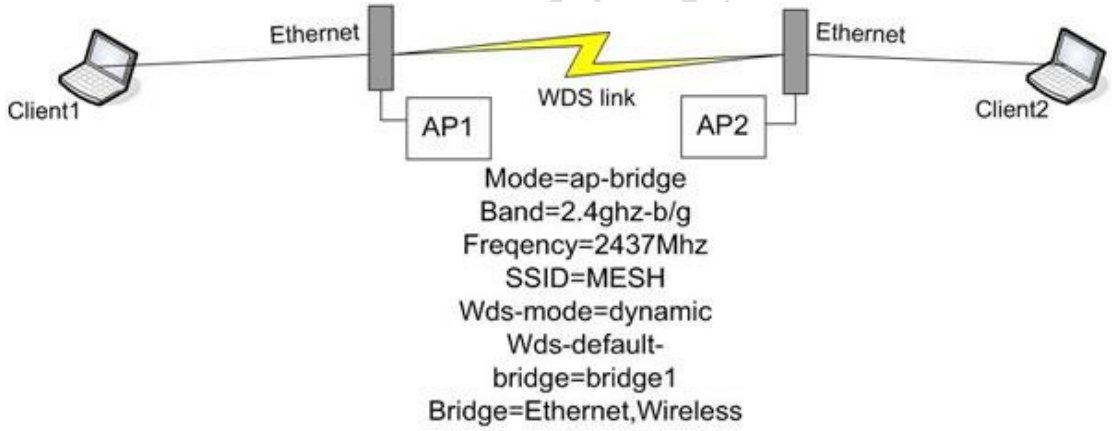
MikroTik 支援的 Mesh 無線網狀網路構建可以分為兩種：

一、基於 STP 的 Mesh：STP（Spanning Tree Protocol）是生成樹協議的英文縮寫。該協定可應用於環路網路，通過一定的演算法實現路徑冗余，同時將環路網路修剪成無環路的樹型網路，從而避免報文在環路網路中的增生和無限迴圈，但是它還是有缺點的，STP 協定的缺陷主要表現在收斂速度上。了解決 STP 協議的這個缺陷，定義了快速生成樹協議 RSTP（Rapid Spanning Tree Protocol）。RSTP 協定在 STP 協定基礎上做了重要改進，使得收斂速度快得多（最快 1 秒以內）。通過 RSTP 構架 Mesh 網路，通過路徑成本和優先順序計算優化整個無線網路。

二、HWMP+協議的 Mesh：基於來至 IEEE802.11s 草案 Hybrid Wireless Mesh Protocol (HWMP)，能用於替代 STP 生成樹協議確保環路的最優路徑。 HWMP+ 協議並不能相容 HWMP 的 IEEE 802.11s 草案。

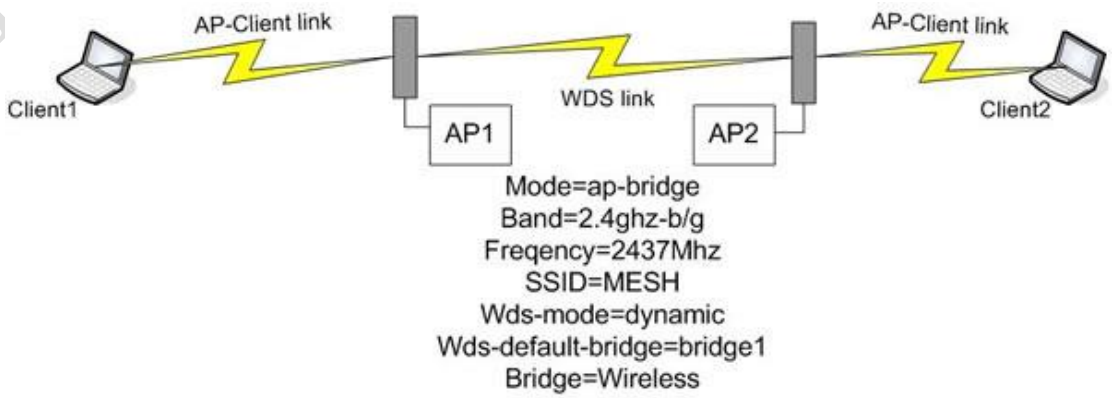
9.1 MikroTik 無線網狀網路的構建

我們如何理解無線網狀網路（Mesh），首先我們通過下面的拓撲圖來具體瞭解一下 Mesh 的概念。首先我們來看看建立一個點對點的無線傳輸：



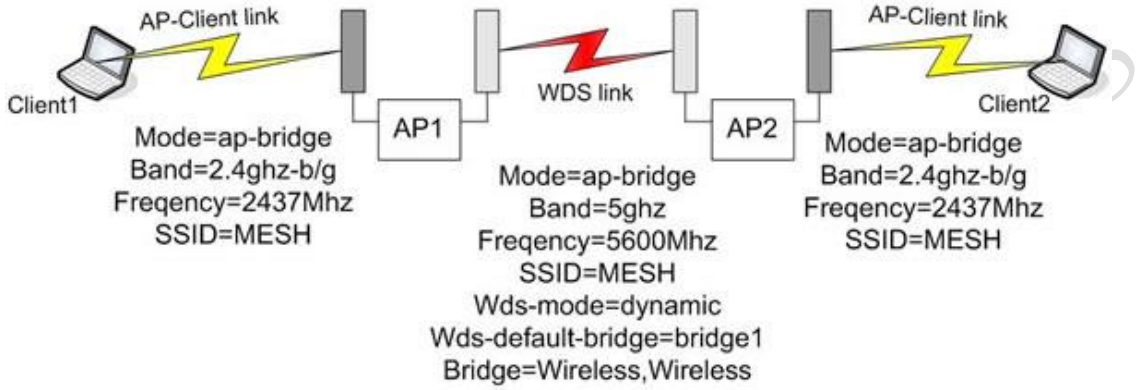
上圖是通過 AP1 和 AP2 使用 WDS 模式將兩個遠端的以太網路橋接起來，這樣 Client1 和 Client2 能直接通過二層互訪。

可能因為兩個以太網傳輸或者用戶端需要，我們將以太網路該無無線網路：



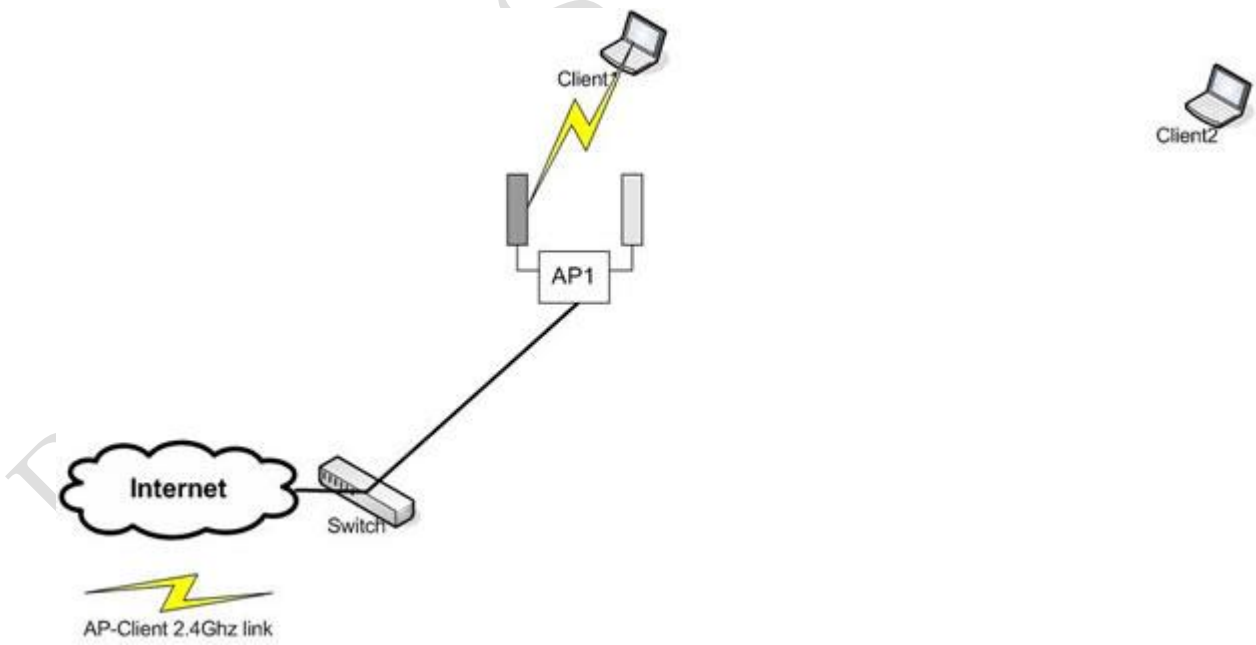
上圖我們同樣使用 WDS 模式，並通過無線中繼的方式或者稱為無線跳躍方式，通過 AP1 和 AP2 採用扇形或者全向天線中繼信號，將 Client1 和 Client2 連接起來。這樣我們通過無線方式替代了，有線的以太網路。如果相對於複雜的施工環境來說，在無線方面相對於有線更加方面和快捷，甚至在某些情況下無線更加的節約成本。

在上面的方案中，我們的兩個 AP 都使用的是一張網卡，相對來說一張網卡做轉發效率並不高，在大資料情況下延遲會明顯增加，為了改善這樣的情況，我們在每個 AP 上多增加一個網卡，一張做 AP 間的傳輸，一張做用戶端的覆蓋。如下圖：

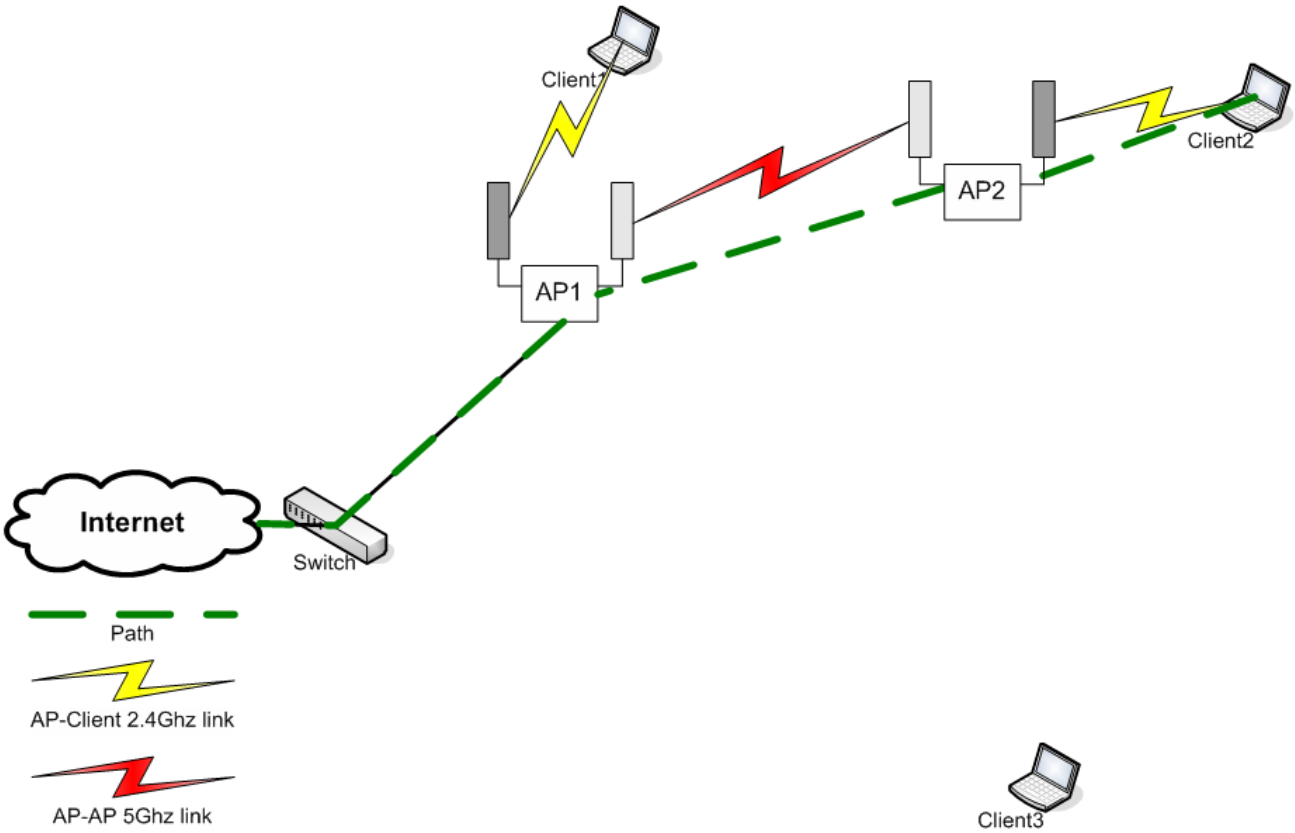


從這上面可以看到紅色的線路代表 AP 之間的通信，採用的是 5G 頻率，而 AP 到用戶端則採用 2.4G 的頻率，這樣的方式增加了無線網路的轉發效率。

以上是 MikroTik 如何使用點對點的無線網路拓撲。接下來我們逐步看看無線網狀網路 Mesh 的構建過程：

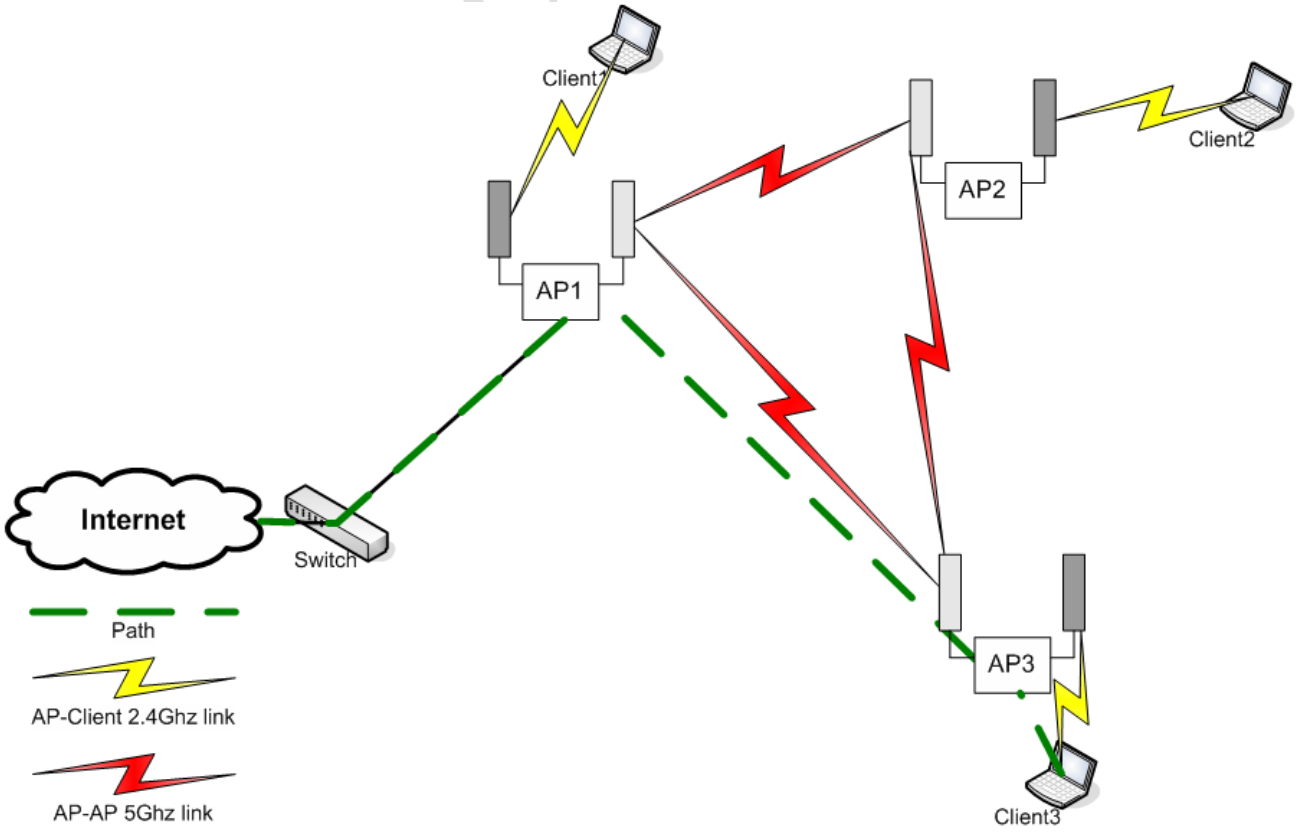


首先我們看看上面的圖，Internet 通過交換機連接到 AP1，AP1 在做橋接後，將資料傳輸到 Client1，但在遠端（在幾公里外）的 Client2 需要接入 Internet，我們通過下圖來看如何實現 Client2 連接如 Internet：



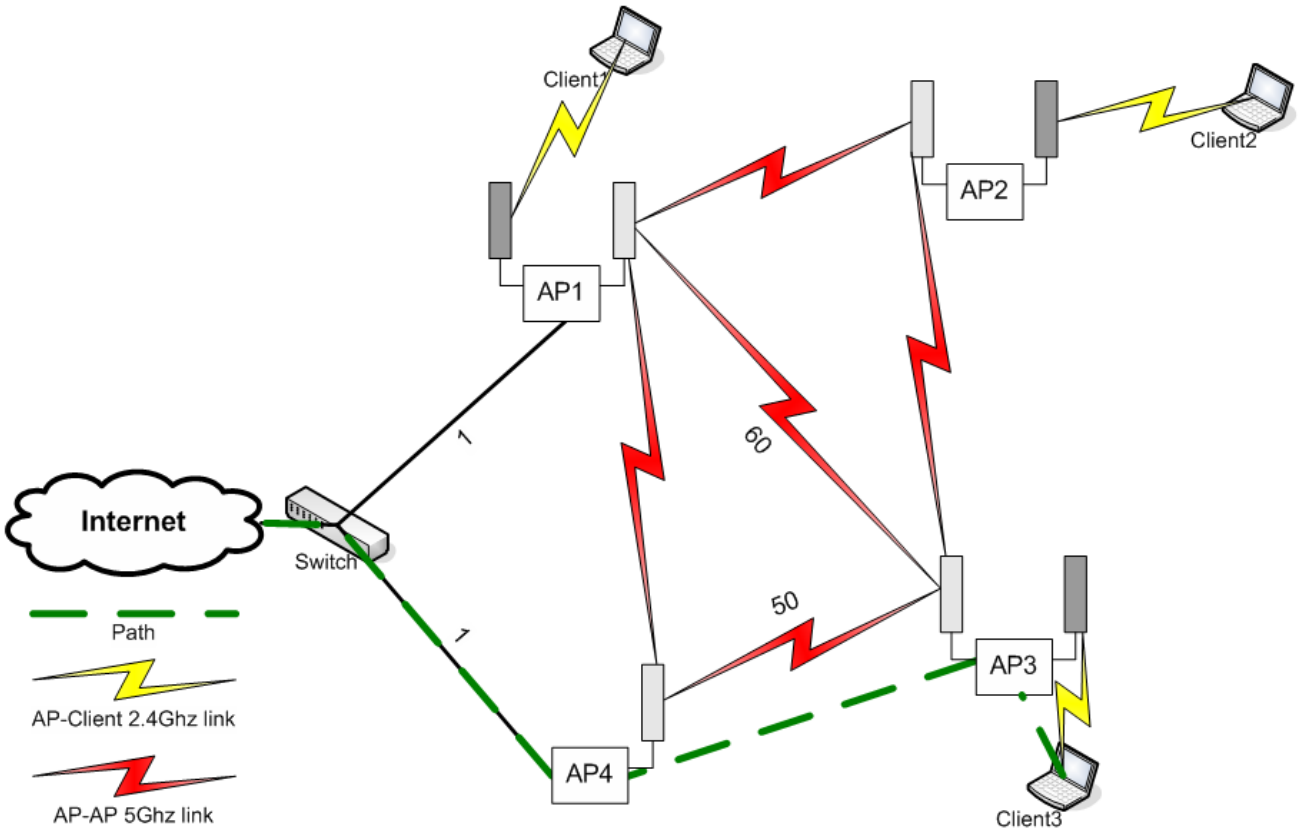
從該圖中，我們通過以太網接入 Internet 並連接到 AP1，AP1 使用了兩個無線發射天線，深灰色的天線使用 2.4G 面向 client1，而淺灰色的天線通過 5G 頻率連接到遠端的 AP2，由 AP2 將信號中繼併發送給 Client2。

這時在我們遠端網路中還有一個 Client3 需要連接，我們可以通過下圖：



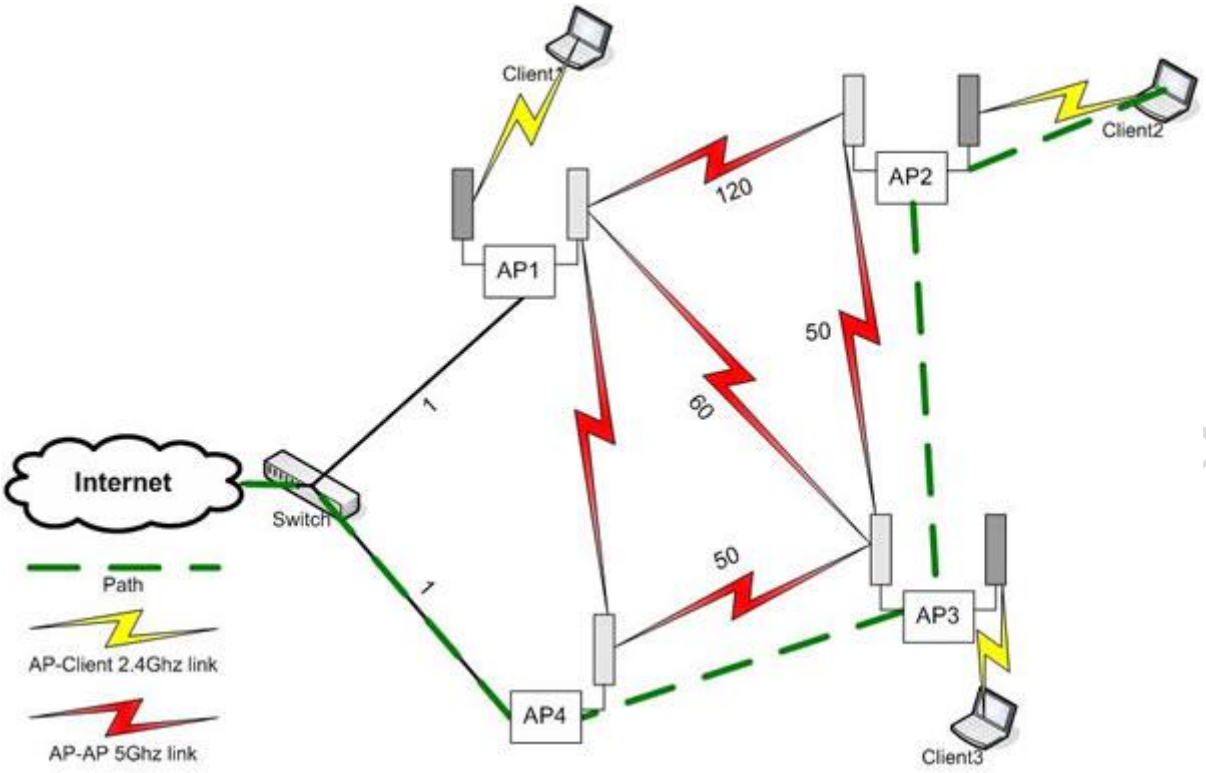
通過增加 AP3 做中繼，並將 Internet 資訊傳送給 Client3，我們通這個拓撲圖可以看出在網路的原有基礎上只是增加了一個 AP3，而沒有增加其他任何設備，而且可以從 AP1 和 AP2 中任意一個設備獲取信號。

當我們增加一個設備 AP4 後，我們的網路拓撲將會有完全的改變，如下圖：



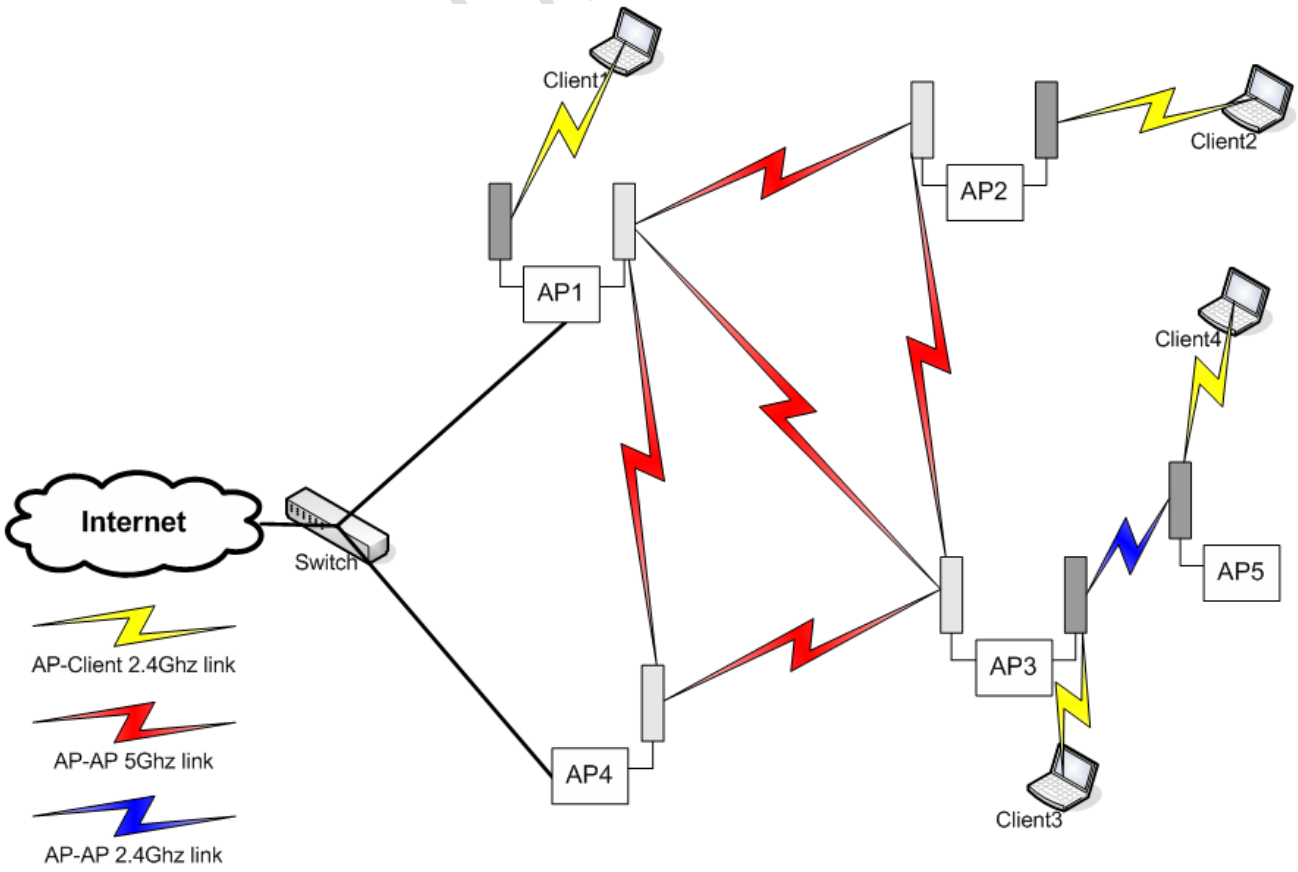
從這個圖上我們可以看到，在增加了 AP4 後，他和 AP1 是兩個同樣接入 Internet 的設備，在拓撲圖中的紅色連接線代表每個設備之間的通信，也就是說 4 個設備相互之間都可以實現相互互訪和線路的切換，當一個設備 AP1 中斷後，Internet 資料可以從 AP4 傳入整個網路，保證訪問 Internet 網路的通暢。

由於在當前的網路中有 4 個設備可以實現相互的訪問，訪問效率的高低，網路延遲的大小則成了上網速度的關鍵，在 MikroTik 的設備中，可以做到根據網路路徑的耗用成本大小來計算出最佳的訪問路線，如下圖：



從這一張圖看到，這裡從交換機到 AP1 和 AP4 的時間延遲都為 1（因為使用以太網），Client2 需要訪問 Internet，我從拓撲圖上可以看到，他最近的訪問路線是從 AP1 直接到 AP2，但這段線路可能存在故障需要花費 120 的時間延遲，由於 AP4-AP3-AP2 的線路花費是 50+50=100 的時間延遲，這樣到 Client2 的線路則會優先選擇最低延遲的那條。

在這個網路中 4 個設備構建了一個骨幹路，在無線網狀網路是可以靈活的拓展你的設備的如下圖：



當我們又增加一個用戶 Client4 時，我們可以通過在 AP3 下面增加一個 AP5 的中繼設備連接到 Client4，這樣通過無線設備的增加和變化使得整個網路構建變的靈活可靠。

只是從網路結構上來說，無線網狀網的優點應該在於：

- 無論固定組網還是移動組網，都能夠迅速按需形成任意拓撲；
- 拓撲遭遇節點高速、高頻變換時，無線網狀網能夠自動調整拓撲並維持連接；
- 能夠採用靈活的多跳傳輸，可隨需擴展，非常適合有線不方便或成本很高的場合；

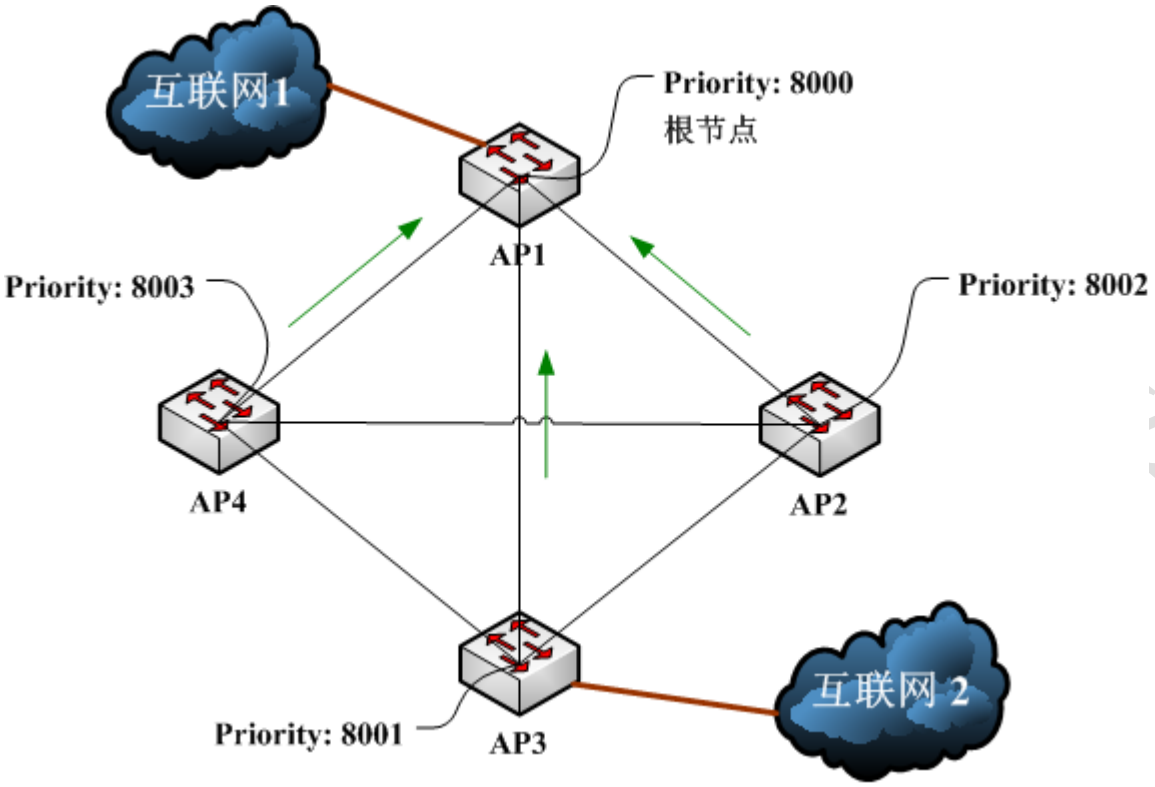
9.2 RSTP MESH 網路配置

RSTP MESH 原理

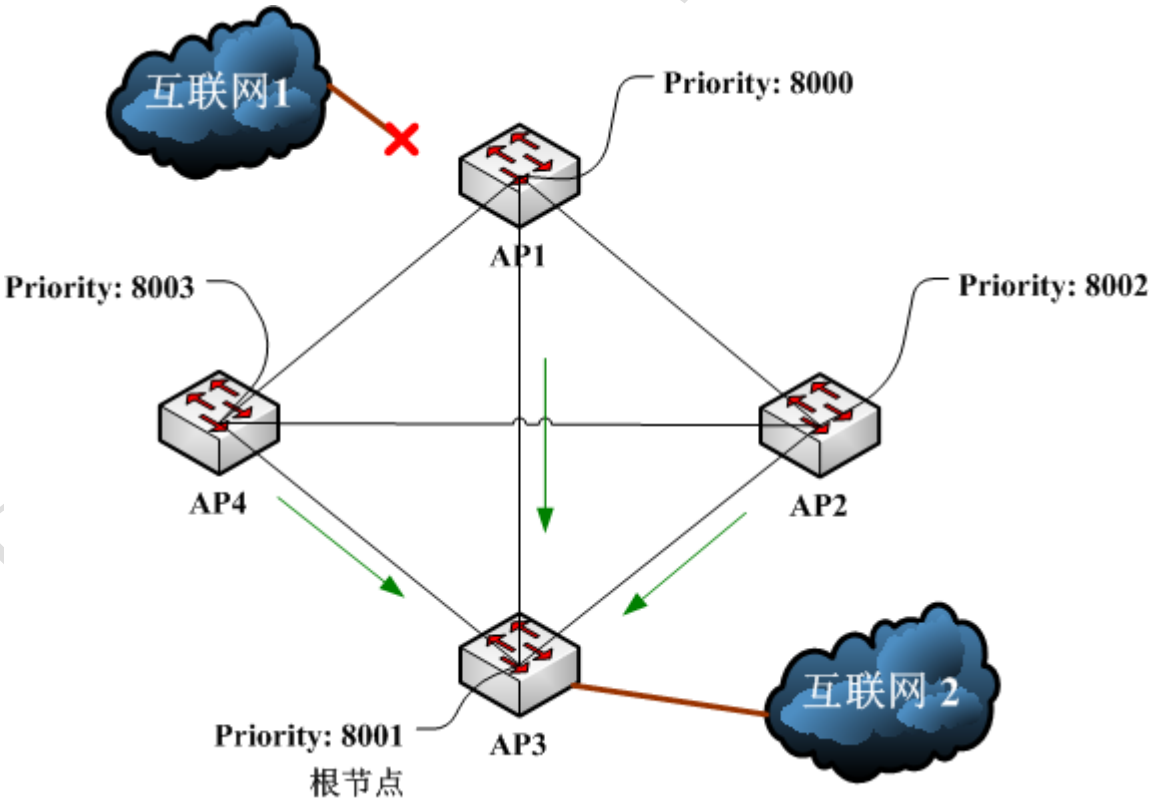
STP 在當拓撲發生變化，新的配置消息要經過一定的時延才能傳播到整個網路，這個時延稱為 Forward Delay，協議預設值是 15 秒。在所有橋接器收到這個變化的消息之前，若舊拓撲結構中處於轉發的端口還沒有發現自己應該在新的拓撲中停止轉發，則可能存在臨時環路。為了解決臨時環路的問題，生成樹使用了一種計時器策略，即在端口從阻塞狀態到轉發狀態中間加上一個隻學習 MAC 位址但不參與轉發的中間狀態，兩次狀態切換的時間長度都是 Forward Delay，這樣就可以保證在拓撲變化的時候不會產生臨時環路。但是，這個看似良好的解決方案實際上帶來的卻是至少兩倍 Forward Delay 的收斂時間！

通過最新的 RSTP 收斂速度會更快，MirkoTik 在 RouterOS 3.0 版本後增加了 rstp 協議，這樣提高了無線網路的性能。

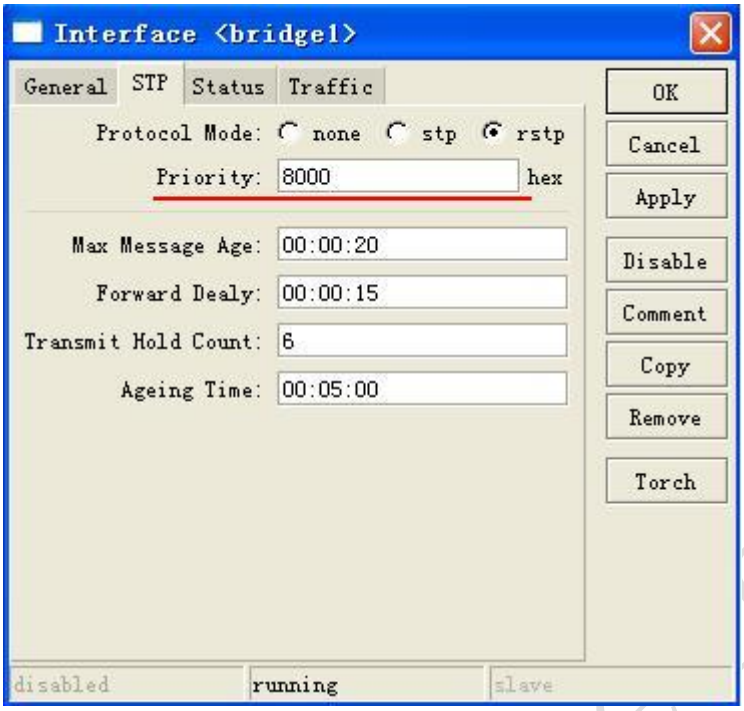
在下面的圖中，可以看到，四個 AP 設備，組成了一個 WDS 的網狀網路，在 AP1 和 AP3 分別接入的 Internet 網路，每個 AP 設備都會分配一個節點優先順序 priority 參數（數字越小，優先順序越高），Rstp 運行情況如下下面是一個正常的 rstp 網路拓撲，AP1 的優先順序 priority 被設置為 8000，即為整個網路的根節點：



當 AP1 因故障斷開後，根節點自動轉換到優先順序為 8001 的 AP3 上：



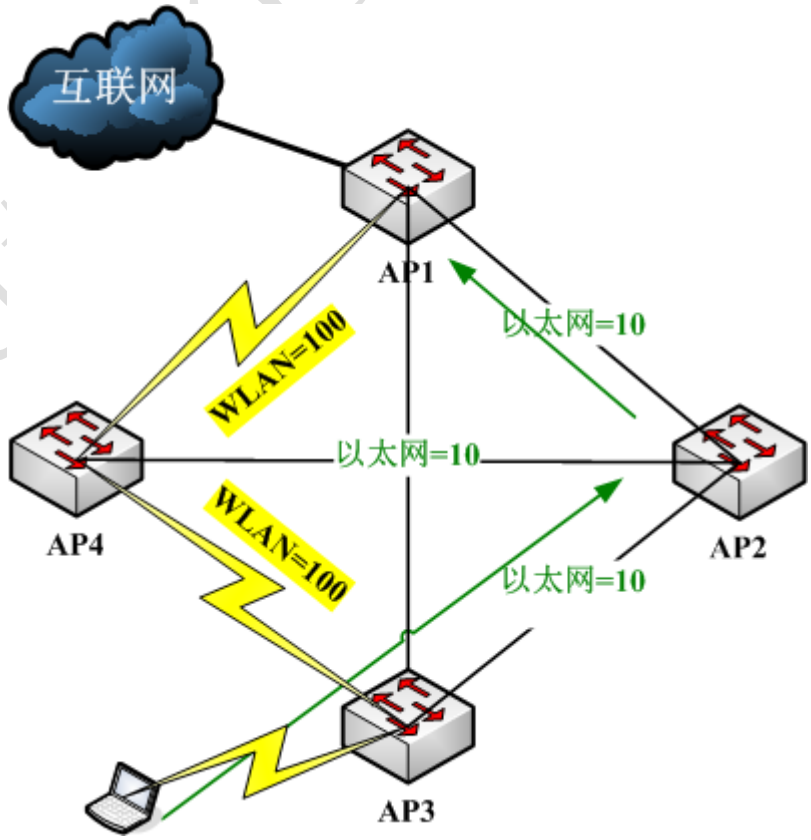
下面是在 RouterOS 上如何配置優先順序 priority 參數，RouterOS 的 rstp 默認 priority 為 8000：



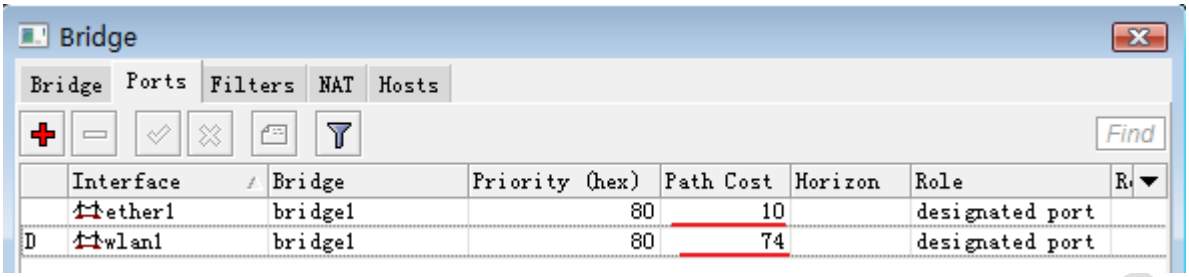
根據每個 AP 的情況，分配不同的優先順序參數，這樣一個完整的 WDS 網狀網路便配置完成。

RSTP 的成本計算

WDS 會通過成本計算，並選擇走最佳的路徑，如下圖我們可以看到，WLAN 默認的成本是 100，而以太網的成本為 10，根據最小值，AP 會選擇最低成本的路徑：

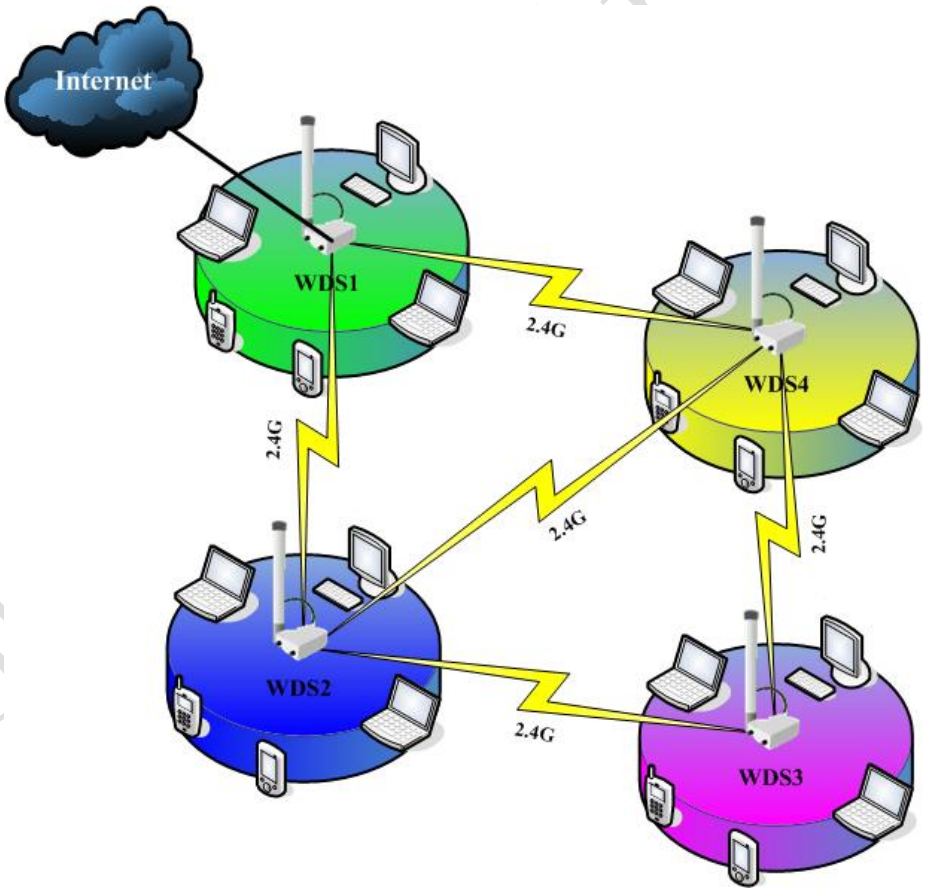


我們可以在 bridge 的 Ports 中看到 Path Cost 參數，ether1 成本為 10，wlan1 成本為 74，系統會自動判斷成本最低的路徑



9.3 WDS 漫遊模式

RSTP 的 MESH 最基本的一種方式為 WDS 漫遊模式，即每個基站都配置 1 個無線模組，安裝 1 個全向天線，對周邊的區域進行覆蓋，WDS 漫遊都採用橋接方式，每個設備的 bridge 都需要啟用 RSTP 協定，避免環路的出現，同時實現設備間的冗余。所以設備間的配置都需要將 Mode 設置為 ap-bridge，Band、頻率、SSID 需要相同的設置就可以了，如下圖：

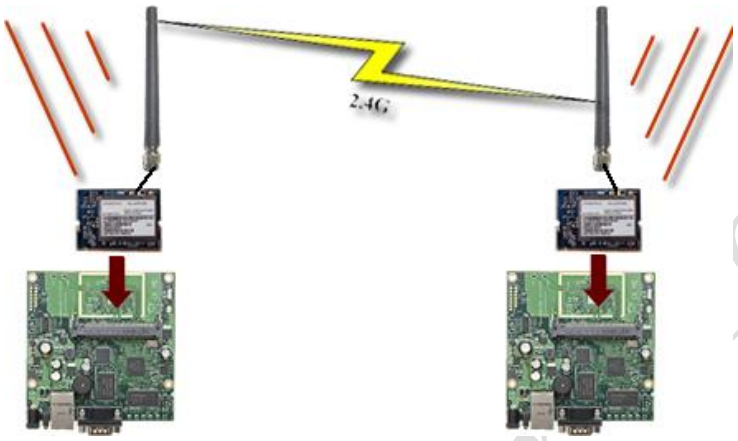


在 RouterOS3.0 中 WDS 選項增加了一組 mesh（無線網狀網路）的設置，該參數能讓 WDS 無線漫遊更好的工作和選擇最佳路徑。推薦使用 dynamic mesh。

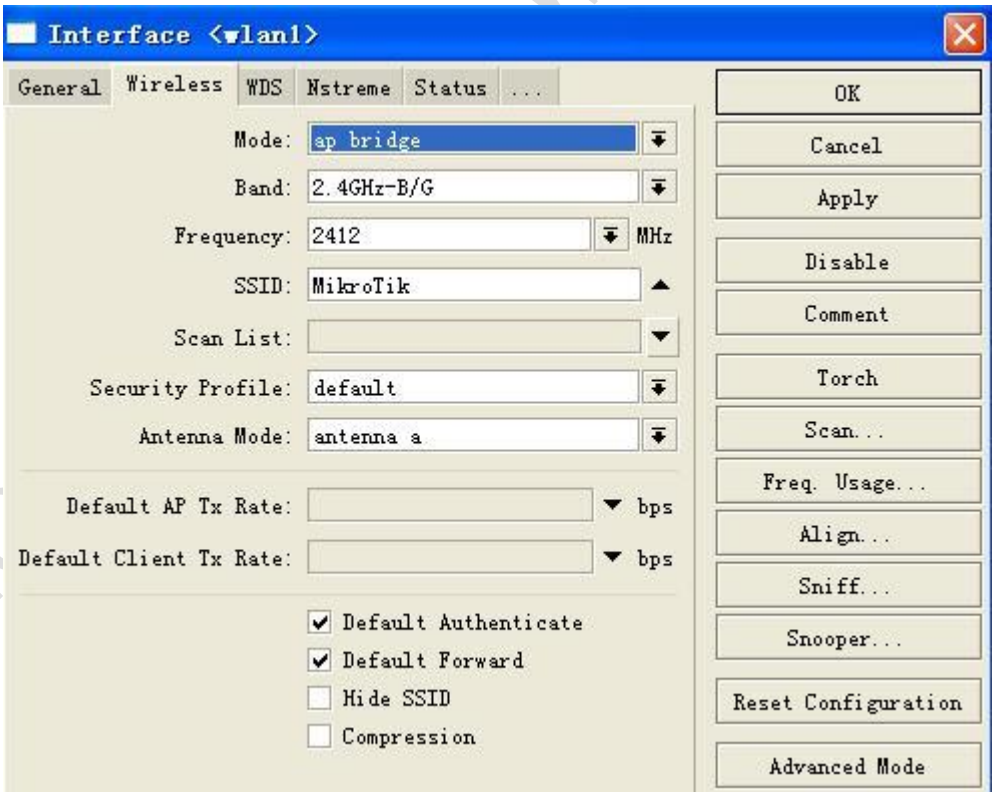
這樣的模式僅能用於小範圍的網路覆蓋，適用於網路訪問要求不高的區域，如大型的展會、辦公樓層覆蓋，以及較小環境的社區，由於這樣的模式只有一個模組和全向天線組成，所以 1 個模組既要做無線覆蓋，又要做各個基站直接的網路信號傳輸，降低了無線模組和設備的工作效率。如果要提高這樣的網路品質，我

們只能通過在每一個 WDS 設備上連接網線，然後骨幹資料通過以太網交換，這樣會帶來一個問題就是網路佈線成本的提高。

在 MikroTik Wireless 選項中我們通過配置多個 MikroTik 無線設備構建一個 WDS 漫遊的無線網路。這裡我們主要通過 802.11bg 的 2.4G 頻段構建 Mesh 網路，每個 AP 安裝一個無線模組，並採用 2.4G 的 bg 協議同時做 AP 間的連接傳輸又做網路的覆蓋，如下圖，我們通過 RB411A 安裝 1 個無線模組，使用全向天線做設備間的傳輸，又作使用者網路的覆蓋：

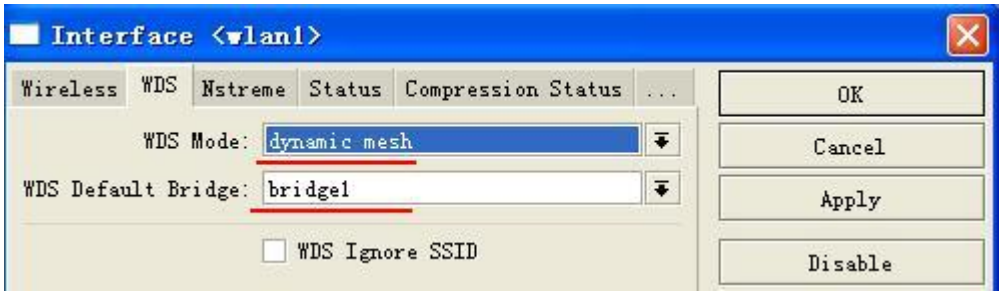


配置這樣的網路時，我們需要將同一個區域內所有 MikroTik AP 設備 Mode 配置為 ap-bridge，然後設置相同頻段，相同頻率，配置為 WDS 模式，然後配置橋模式，如下圖：



這裡我們配置參數:： Mode:: ap-bridge

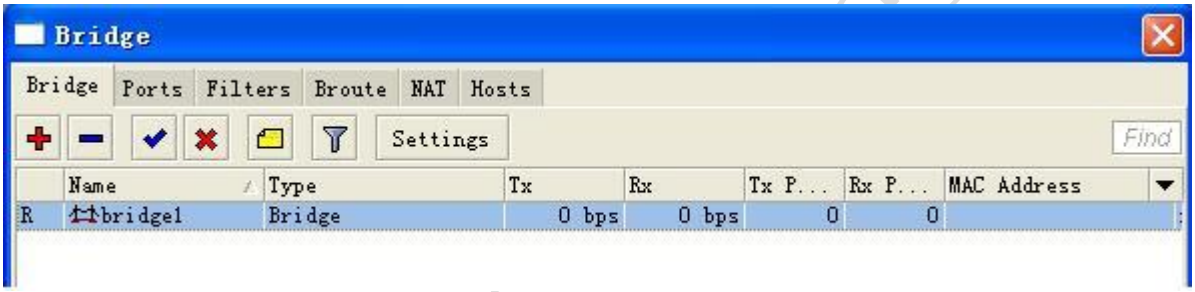
- Band：2.4GHz-B/G
- Frequency：2412
- SSID：MikroTik



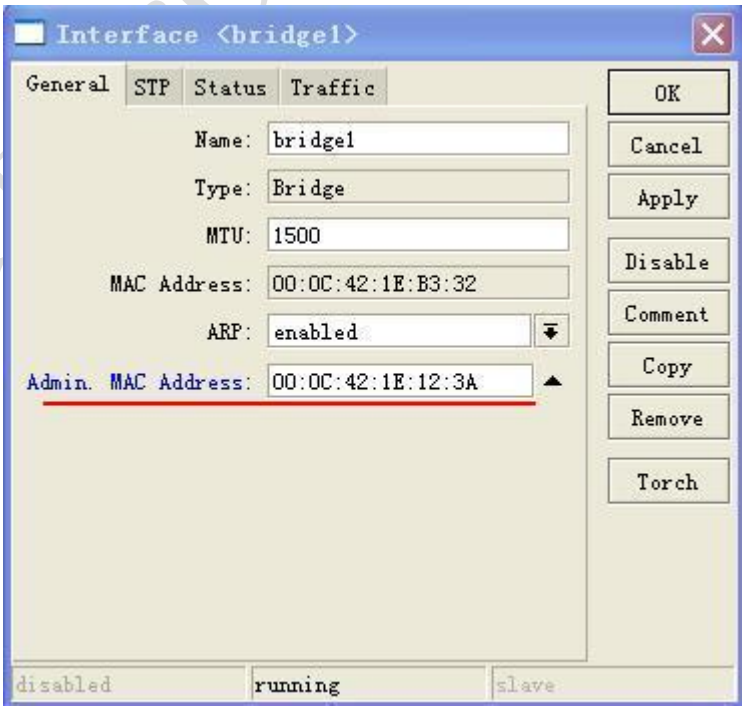
WDS 參數，我們配置 WDS Mode 為 dynamic mesh，並添加到默認 bridge 為 bridge1。採用 dynamic mesh 在 3.0 版本中要比在 dynamic 模式下運行的更穩定快速。

以上是無線網卡的配置參數，在同一區域內的 MikroTik AP 設備無線參數幾乎都是相同的。但為了更好的讓區域內的多個 AP 在轉發資料時，達到最優路徑我們需要通過 bridge 的 rstp 來完成。這裡我們需要瞭解如何配置 bridge 中的 rstp：

首先我們在 Bridge 選項中添加一個 bridge1：



這裡我們需要對 bridge1 的的 MAC 位址配置，因為在建立 bridge 後，橋接介面會自動產生 MAC 位址，可能在運行時候，MAC 位址會自動變換，為了保證網路品質的穩定我在這裡配置一個靜態 MAC 位址給 bridge1。

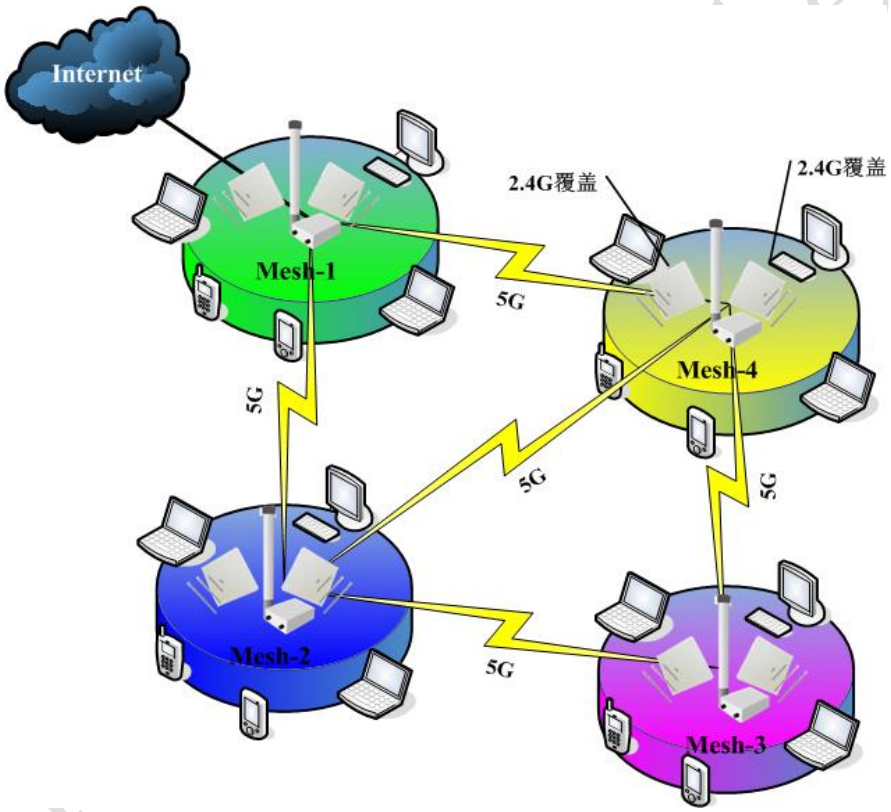


採用靜態 MAC 位址有助於 RSTP Mesh 網路的穩定，我們可以自己設置一個 MAC 位址，或者使用設備上任何介面的 MAC 在 bridge 上。

注意：採用這樣的 WDS 網路配置，一般使用於小型的無線漫遊網路（如多層的辦公室、大型的展廳和小範圍的用戶覆蓋），AP 數量在 5-8 個左右的網路。採用這樣的 WDS 模式受到連接使用者數的限制，如果用戶過多會造成大量的無線干擾，降低網路連接效率，這樣的網路一個 AP 的使用者連接數最好限制在 15-20 個以內。

9.4 多介面 Mesh 網路事例

多介面 Mesh 是解決前一例覆蓋中只適用 1 張無線模組既做覆蓋又做傳輸時帶來的無線網卡效率過低的問題，在做城區或者社區覆蓋時，由於全向天線效果不太理想，我們會考慮使用定向的磁區天線，但磁區天線覆蓋範圍有限，但我們可以通過增加無線網卡介面與天線彌補這個問題，RB433/AH、RB493/AH、RB600 等都支援擴展 3 個或者 4 個無線介面。多介面的 Mesh 以採用 3 個介面的無線模組為例，我們通過 1 個無線模組為骨幹傳輸使用，保證骨幹有足夠的頻寬，其余 2 個對周邊區域進行覆蓋，增加覆蓋面積和增強信號。



用多介面的無線 Mesh 覆蓋，能彌補信號的盲點，增強信號。其好處在能減少設備的投入，普通的 AP 一般只能提供 1-2 個無線模組，但這樣在覆蓋中的效果相對較弱，如果需要彌補信號，則需要增加 AP 設備，但如果通過多介面的 RouterBOARD 產品，則只需要增加無線模組和天線，大大節約了成本，並增強的無線設備的可管理性。在配製中，我們同樣採用 RSTP 與 WDS 結合的方式，即 AP1 的互聯網出口為網路的根節點，即設置高的 RSTP 優先順序，其余的 AP 則根據需要設置優先順序別。、

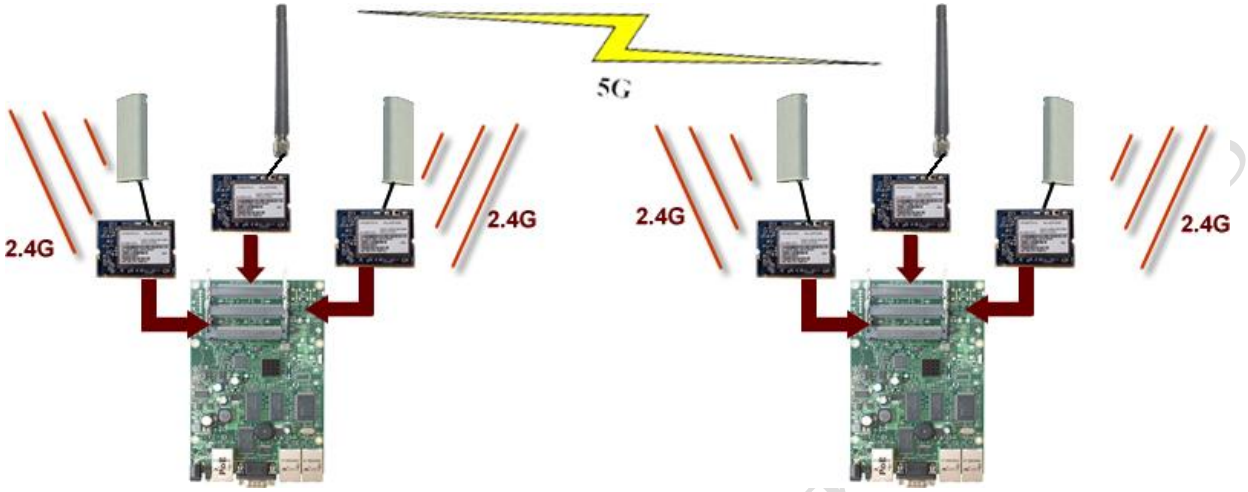
多介面 Mesh 無線配製：

我們將所有的無線模組設置為 AP-bridge 模式，只是骨幹和覆蓋模組發射頻段不同，一般骨幹我們採用 5G 模式，2.4G 用於覆蓋。

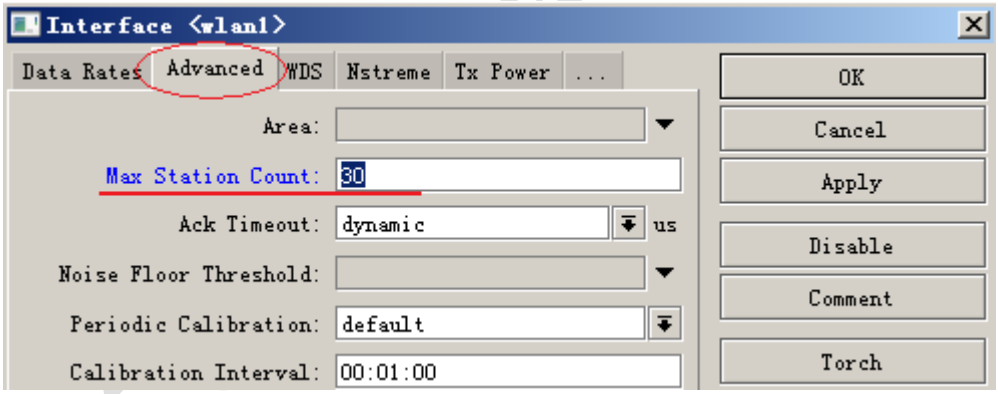
- 骨幹與其他骨幹間採用相同的發射頻率和 SSID，模式為 ap-bridge，但骨幹與覆蓋的 SSID 不同。

- 覆蓋之間的 SSID 相同，發射頻率可以不同，以避免頻道重複出現的干擾。
- 覆蓋的 AP 需設定一個可連接的終端值，限制過多使用者擁擠到一個覆蓋 AP 上，避免增加干擾和堵塞。
- 將三個無線介面定義到一個 bridge 中，並設置 RSTP 的優先順序和相應的參數。

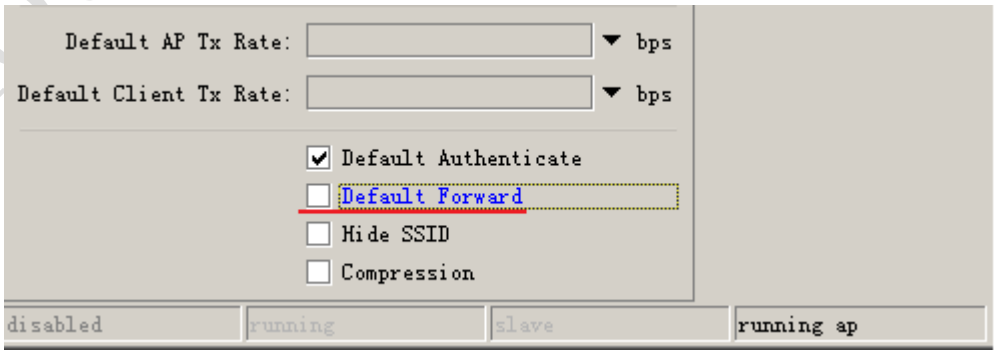
我們通過在一個 RB433 上增加 3 個無線模組，一個用於骨幹連接，另外兩個則用於網路覆蓋：



我們在 wireless 的覆蓋無線網卡上配製終端連接數，一般配置 30 個終端比較合適：



在無線的 wireless 參數種，我們關閉掉 Default Forward 參數，禁止無線終端之間相互通信：



以上兩種配置，能提高無線覆蓋的效率和穩定性。

9.5 Station 模式下通過腳本切換 AP 基站

關於 RouterOS 的無線 RB 設備作為 station（即終端設備）時，在多個相同 SSID 的 AP 基站間如何漫遊，很多人完全歸結於 AP 設置，其實 AP 設置占了一部分，作為終端設備如何選取 AP 基站是很關鍵的，在 AP 基站間做切換，什麼時候切換都不是 AP 決定的，而是終端設備自己選擇。切換 AP 信號最通俗的理解就是信號強度，當信號強度無法滿足需要的時候，我們就需要讓終端設備切換到其他信號較強的 AP 上。

根據信號強度我們可以通過判斷當前連接 AP 基站的信號值，如果低於多少，我們作出要求 station 終端設備切換的操作。無線設備的網卡初始化連接時，周圍存在多個相同 SSID 的 AP 基站，始終會選擇信號強度最好的 AP 基站，因此根據這個特性，可以理解為將 RB 無線網卡初始化一次（不是重定），也就將當前信號連接從 registration-table 中刪除，重新選擇新的信號。

當然刪除信號操作，在 RouterOS 內部肯定不是手動操作的，必須通過週期性的執行腳本完成，腳本讀取當前信號強度，判斷是否超過閥門值，超過後，就執行禁用和啟用網卡操作，實現切換的目的。這樣的漫遊肯定會造成網路延遲，因為承載網路的實體信號都會中斷，肯定會掉包，只是看初始化和建立連接的時間會有多長，根據之前的經驗一般會掉 2-3 個包左右。

下面腳本僅供參考：

```
:loacl sig
:set sig [/interface wireless registration-table get [/interface wireless registration-table
find interface="wlan1"] signal-strength ]
:set sig [:pick $sig 0 [:find $sig "d" ]]
:if ($sig < -80) do={
  /interface wireless registration-table remove [find interface=wlan1]
}
```

腳本運行，需要在 RouterOS 計畫任務在完成，設置週期執行時間，最小單位為“秒”，配置進入/system scheduler（配置操作可以參考《RouterOS 入門到精通》第 39 章節）

9.6 基於 Connect-list 的 Station 無線漫遊

無線設備的網卡在初始化連接時，周圍存在多個相同 SSID 的 AP 基站，始終會選擇信號強度最好的 AP 基站，這裡我考慮使用 connect-list 去完成這個操作，通過 connect-list 完成對 AP 基站的匹配（事先要輸入 AP 基站的 MAC 位址）信號強度，當連接 AP 基站的信號強度低於一定設定值後，會斷開無線連接，這樣 station 會嘗試連接到 connect-list 中其他 AP 基站，且信號範圍在給定的值內。

- 測試的設備：RouterBOARD 設備 RB951Ui-2HnD
- RouterOS 版本：v6.38.1
- 基本配置情況：2 台做 ap-bridge，1 台做 station-wds，所有設備建立 rstp 的 WDS 漫遊網路，2 台設備 ap-bridge 無線參數配置相同，啟用 wds-dynamic，添加到 bridge1
- 測試環境：室內

根據以上的配置，下面通過配置腳本的形式給出

AP1 配置參數

```
/interface bridge
add name=bridge1 priority=0x8001 protocol-mode=rstp
/interface bridge port
add bridge=bridge1 interface=ether1

/ip address
add address=192.168.11.50/24 interface=bridge1 network=192.168.11.0

/ip route
add distance=1 gateway=192.168.11.1

/interface wireless
set [ find default-name=wlan1 ] area=mik band=2ghz-b/g/n channel-width=20/40mh
    disabled=no mode=ap-bridge radio-name=AP1 ssid=mik1 wds-default-bridg
bridge1 wds-mode=dynamic-mesh wireless-protocol=802.11
```

AP2 配置參數

```
/interface bridge
add name=bridge1 priority=0x8002 protocol-mode=rstp
/interface bridge port
add bridge=bridge1 interface=ether1

/ip address
add address=192.168.11.51/24 interface=bridge1 network=192.168.11.0

/ip route
add distance=1 gateway=192.168.11.1

/interface wireless
set [ find default-name=wlan1 ] area=mik band=2ghz-b/g/n channel-width=20/40mh
    disabled=no mode=ap-bridge radio-name=AP2 ssid=mik1 wds-default-bridg
bridge1 wds-mode=dynamic-mesh wireless-protocol=802.11
```

station 配置參數

```
/interface bridge
add name=bridge1 priority=0x8010 protocol-mode=rstp
/interface bridge port
add bridge=bridge1 interface=ether1
/ip address
add address=192.168.11.55/24 interface=bridge1 network=192.168.11.0

/ip route
add distance=1 gateway=192.168.11.1

/interface wireless
```

```
set [ find default-name=wlan1 ] area=mik band=2ghz-b/g/n disabled=no \
    mode=station-wds radio-name=Client ssid=mik1 wds-default-bridge=bridge1 \
    wds-mode=dynamic-mesh wireless-protocol=802.11
/interface wireless connect-list
add interface=wlan1 mac-address=E4:8D:8C:60:B6:CD security-profile=default \
    signal-range=-50..1
add interface=wlan1 mac-address=E4:8D:8C:BD:14:D1 security-profile=default \
    signal-range=-50..1
```

測試結果：

在不使用腳本判斷網卡信號強度低於多少的情況下，通過 connect-list 判斷信號來切換 AP 是可行的，切換時會丟 2-3 個包。注意由於是室內環境測試，signal-rang 設置為-50 到 1dBm 較高範圍，因此如果是實際的應用場景，需要更加實地信號勘察後作出 signal-rang 的配置。

以上測試是基於 802.11 協議，當改為 nv2 協議後，connect-list 切換會失效，修改 connect-list 參數時，wireless 應用導致 CPU 100%（問題已經回饋給 mikrotik，並得到回復在後續版本會修正），如果禁用啟用網卡方式切換，會丟 38 個包

9.7 HWMP+ Mesh 無線網狀網路

HWMP+是 MikroTik 為無線網狀網路 Mesh 定義的 2 層路由式通訊協定。基於 IEEE802.11s 草案 Hybrid Wireless Mesh Protocol (HWMP)，能用於替代 STP 生成樹協議確保環路的最優路徑。 HWMP+ 協議並不能相容 HWMP 的 IEEE 802.11s 草案。

注：這種分散式系統不僅能應用到無線分佈系統（WDS）。HWMP+網狀網路同樣也支援以太網介面的網狀網路，因此你可以用於簡單的以太網分佈系統，或者同時連接 WDS 和以太網。

RouterOS Mesh 選項基於 HWMP，Mesh 與 RSTP 的無線組網區別在於，HWMP 是基於跳躍級數選擇路徑，而 RSTP 則是根據路徑成本開銷選擇路徑

interface mesh 屬性

PREQ 路由請求

PREP 路由應答

admin-mac (MAC 地址, 默認: 00:00:00:00:00:00) - 管理分配的 MAC 位址，當 auto-mac 設置為 disable 後起作用。
arp (disabled | enabled | proxy-arp | reply-only; 默認: enabled) - 位址解析通訊協定設置
auto-mac (boolean, 默認: no) - 如設置為禁用，這時 admin-mac 將會被要求設置 mesh interface 上，否則會使用一些端口的位址。
hwmp-default-hoplimit (integer: 1..255) - 路由式通訊協定包產生最大的跳躍總數，在一個 HWMP+ 資料包被發送後，達到最大跳躍限制數，將會被自動丟棄。
hwmp-prep-lifetime (time, 默認: 5m) - 為創建線路從收到 PREP 或 PREQ 資訊的生存時間
hwmp-preq-destination-only (布林值, 默認: yes) -- 是否只有目的地可以回應 hwmp + preq 訊息

hwmp-preq-reply-and-forward (布林值, 默認: yes) --是否中間節點應該發出 hwmp + preq 消息後，回應它。 應用於僅當 hwmp-preq-destination-only 被禁用
hwmp-preq-retries (整型, 默認: 2) - 當位址無法到達情況下，多長時間重試探測指定 MAC 位址的路徑。
hwmp-preq-waiting-time (時間, 默認: 4s) - 多長時間等待一個響應第一個 PREQ 資訊。
hwmp-rann-interval (時間, 默認: 10s) - 間隔多長發送 HWMP+ RANN 資訊
hwmp-rann-lifetime (時間, 默認: 1s) - 為創建路徑從收到 RANN 資訊的生存時間
hwmp-rann-propagation-delay (數字, 預設: 50) - 多久前等待發送 RANN 資訊。值為百分之一秒計算 (100cs = 1sec)
mesh-portal (布林值, 默認: no) - 是否設定這個介面為一個 Mesh 網路的入口。
mtu (數字, 預設: 1500) - 最大傳輸單元
name (字元) - 介面名稱
reoptimize-paths (布林值, 默認: no) - 是否定期向外發送 PREQ 資訊詢問網路中的 MAC 位址，如果網路拓撲經常變動基於 Turing 設置是非常有用的。注意：如果沒有接收到一重新優化 PREQ 資訊，將保持現有的路徑（直到探測超時）

/interface mesh port 屬性。

hello-interval (時間, 默認: 10s) - 發送 HWMP+ Hello 資訊最大時間間隔。只能用於以太網卡類型的端口。
interface (interface name) - 介面名稱，那一個介面包括在 Mesh 中
mesh (interface name) - 屬於那一個 Mesh 介面
path-cost (整型: 0..65535; 默認: 10) - 介面的路徑成本，通過最佳路徑決定使用的路由式通訊協定。
port-type (WDS | auto | ethernet | wireless) - 使用的端口類型

- auto - 根據介面類別型自動決定端口使用的類型
- WDS - 一個無線分散式系統介面，一種點對點無線連接。遠端 MAC 位址通過無線連接資料得知。
- ethernet - 遠端 MAC 位址通過每次 HWMP+的 Hello 資訊、接收到的源 MAC 位址或者轉發的傳輸資料中學習到
- wireless - 遠端 MAC 位址獲取通過無線連接資料

port-type-used (唯讀, wireless | WDS | ethernet-mesh | ethernet-bridge | ethernet-mixed) - 端口類型和確認的狀態

/interface mesh fdb 屬性： 唯讀狀態下的 Mesh 介面轉發資料庫(FDB).

mac-address (MAC 位址) - MAC 位址相對應的 FDB 記錄專案
seqnum (整型) - 序列編號使用到路由式通訊協定中避免環路
type (local | outsider | direct | mesh | neighbor | larval | unknown) - FDB 記錄項的類型

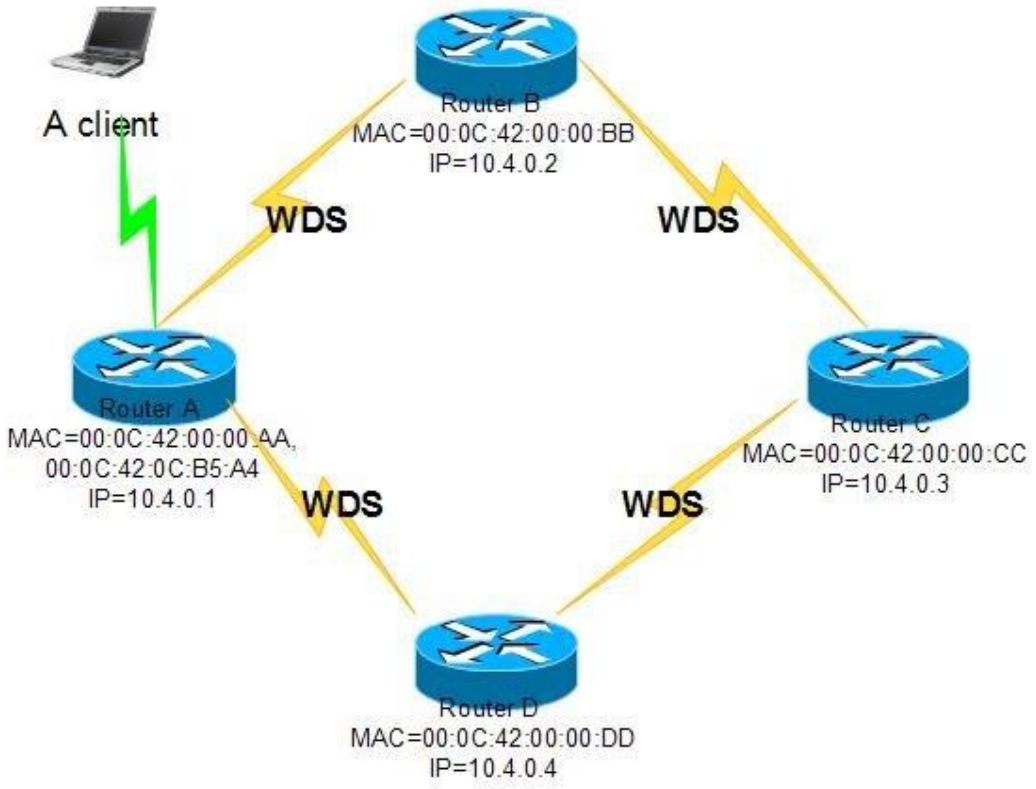
- local - MAC 地址屬於本地路由器
- outsider - MAC 位址屬於 Mesh 網路外部的設備
- direct - MAC 位址屬於在一個 Mesh 網路中一個介面上的無線用戶端
- mesh - MAC 位址屬於一個設備到達了 Mesh 網路，可以是內部或者外部網路
- neighbor - MAC 位址屬於一個 Mesh 路由器，是一個直接的鄰居路由器
- larval - MAC 位址屬於一個未知到達 Mesh 網路
- unknown - MAC 位址屬於一個未知的設備

mesh (interface name) - 屬於這個 Mesh 介面的 FDB 項
on-interface (interface name) - Mesh 端口用於傳輸轉發，一種 next-hop 值
lifetime (time) - 如果記錄項沒有使用傳輸轉發，則會定義生存時間。
age (time) - FDB 項的時期
metric (integer) - metric 值是設置通過路由式通訊協定決定最佳路徑

使用 **wds-default-cost** 和 **wds-cost-range** 無線介面參數會被路由式通訊協定使用， 該 WDS 的成本將被用來作為 **path-cost** 端口動態添加到網格介面。

應用實例

使用 wds-default-cost 和 wds-cost-range 無線介面參數會被路由式通訊協定使用， 該 WDS 的成本將被用來作為 path-cost 端口動態添加到網格介面。



這個事例使用靜態 WDS 連接，當無線連接被啟動，會自動添加到 Mesh 端口中。兩個不同的發射頻率會被使用：一個為 AP 間的通信鏈路，一個為用戶端的 AP 覆蓋。因此一個設備至少需要兩個無線網卡介面。

下面的配置適用於所有的 AP：

```
/interface mesh add disabled=no
/interface mesh port add interface=wlan1 mesh=mesh1
/interface mesh port add interface=wlan2 mesh=mesh1
```

用於 AP 間互聯的 interface

```
/interface wireless set wlan1 disabled=no ssid=mesh frequency=2437 band=2.4ghz-b/g
mode=ap-bridge wds-mode=static-mesh wds-default-bridge=mesh1
```

用於用戶端連接的 interface

```
/interface wireless set wlan2 disabled=no ssid=mesh-clients frequency=5180 band=5ghz
mode=ap-bridge
```

為每一個 AP 配置一個靜態的 WDS 介面連接

```
/interface wireless wds add disabled=no master-interface=wlan1 name=<descriptive name of
remote end> wds-address=<MAC address of remote end>
```

注意：這裡的 WDS 介面設置需要手動，因為我們採用的是靜態 WDS 模式 ，如果你使用 wds-mode=dynamic-mesh，所有的 WDS 介面將會自動創建。

在真實環境中最好需要注意無線連接的安全問題。可以使用/interface wireless security-profile.

在路由器 A 上的結果（現在有一個用戶端連接到 Wlan2 ）：

```
[admin@A] > /interface mesh pr
Flags: X - disabled, R - running
 0  R name="mesh1" mtu=1500 arp=enabled mac-address=00:0C:42:0C:B5:A4 auto-mac=yes
      admin-mac=00:00:00:00:00:00 mesh-portal=no hwmp-default-hoplimit=32
      hwmp-preq-waiting-time=4s hwmp-preq-retries=2 hwmp-preq-destination-only=yes
      hwmp-preq-reply-and-forward=yes hwmp-prep-lifetime=5m hwmp-rann-interval=10s
      hwmp-rann-propagation-delay=1s hwmp-rann-lifetime=22s

[admin@A] > interface mesh port p detail
Flags: X - disabled, I - inactive, D - dynamic
 0    interface=wlan1 mesh=mesh1 path-cost=10 hello-interval=10s port-type=auto
      port-type-used=wireless
 1    interface=wlan2 mesh=mesh1 path-cost=10 hello-interval=10s port-type=auto
       port-type-used=wireless
 2  D interface=router_B mesh=mesh1 path-cost=105 hello-interval=10s port-type=auto
       port-type-used=WDS
 3  D interface=router_D mesh=mesh1 path-cost=76 hello-interval=10s port-type=auto
       port-type-used=WDS
```

FDB (轉發資料庫 Forwarding Database)在當前狀態下包含的本地 MAC 位址資訊，Mesh 節點能能到達的本地介面和探測到的 Mesh 鄰居：

```
[admin@A] /interface mesh> fdb print
Flags: A - active, R - root
   MESH        TYPE     MAC-ADDRESS       ON-INTERFACE       LIFETIME     AGE
A  mesh1       local    00:0C:42:00:00:AA                                 3m17s
```

```
A  mesh1       neighbor 00:0C:42:00:00:BB router_B                          1m2s
A  mesh1       neighbor 00:0C:42:00:00:DD router_D                          3m16s
A  mesh1       direct   00:0C:42:0C:7A:2B wlan2                            2m56s
A  mesh1       local    00:0C:42:0C:B5:A4                                  2m56s

[admin@A] /interface mesh> fdb print detail
Flags: A - active, R - root
 A  mac-address=00:0C:42:00:00:AA type=local age=3m21s mesh=mesh1 metric=0
     seqnum=4294967196
 A  mac-address=00:0C:42:00:00:BB type=neighbor on-interface=router_B age=1m6s
    mesh=mesh1 metric=132 seqnum=4294967196
 A  mac-address=00:0C:42:00:00:DD type=neighbor on-interface=router_D age=3m20s
     mesh=mesh1 metric=79 seqnum=4294967196
 A  mac-address=00:0C:42:0C:7A:2B type=direct on-interface=wlan2 age=3m mesh=mesh1
     metric=10 seqnum=0
 A  mac-address=00:0C:42:0C:B5:A4 type=local age=3m mesh=mesh1 metric=0 seqnum=0
```

測試 ping ：

```
[admin@A] > /ping 00:0C:42:00:00:CC
00:0C:42:00:00:CC 64 byte ping time=108 ms
00:0C:42:00:00:CC 64 byte ping time=51 ms
00:0C:42:00:00:CC 64 byte ping time=39 ms
00:0C:42:00:00:CC 64 byte ping time=43 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 39/60.2/108 ms
```

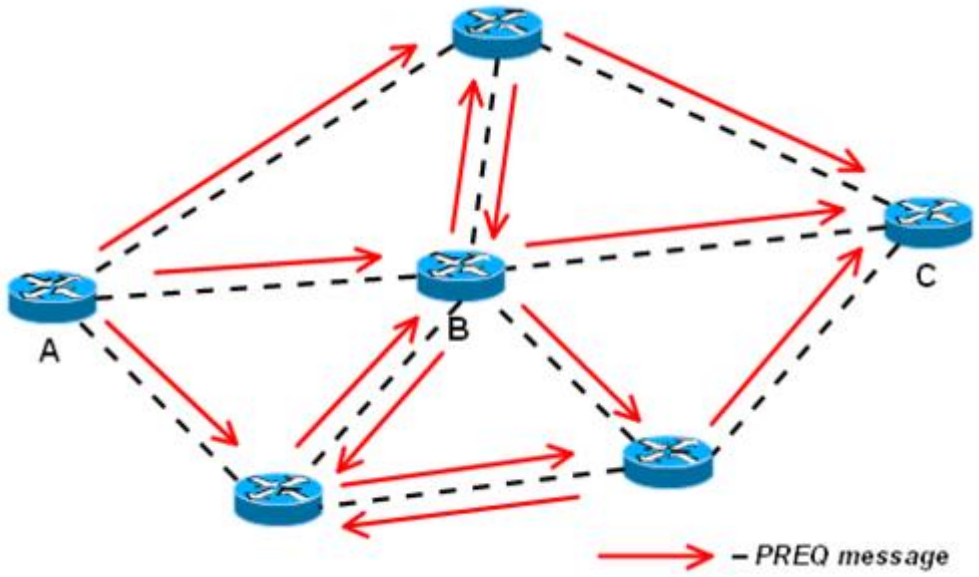
Router A 必須探測到 Router C 的路徑，因此第一個 ping 包延遲稍微大一點。

同樣我們也可以通過 IP 層的 ping 檢測網路 A：

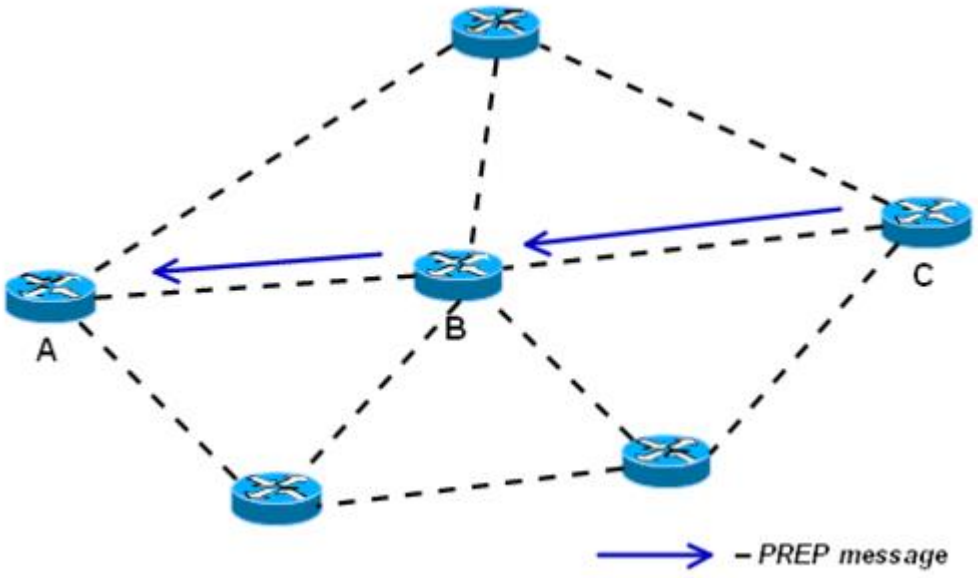
```
[admin@A] > /ping 10.4.0.3
10.4.0.3 64 byte ping: ttl=64 time=163 ms
10.4.0.3 64 byte ping: ttl=64 time=46 ms
10.4.0.3 64 byte ping: ttl=64 time=48 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 46/85.6/163 ms
```

HWMP 協定特性

1、反應方式



Router A 想要探測到 C 的路徑



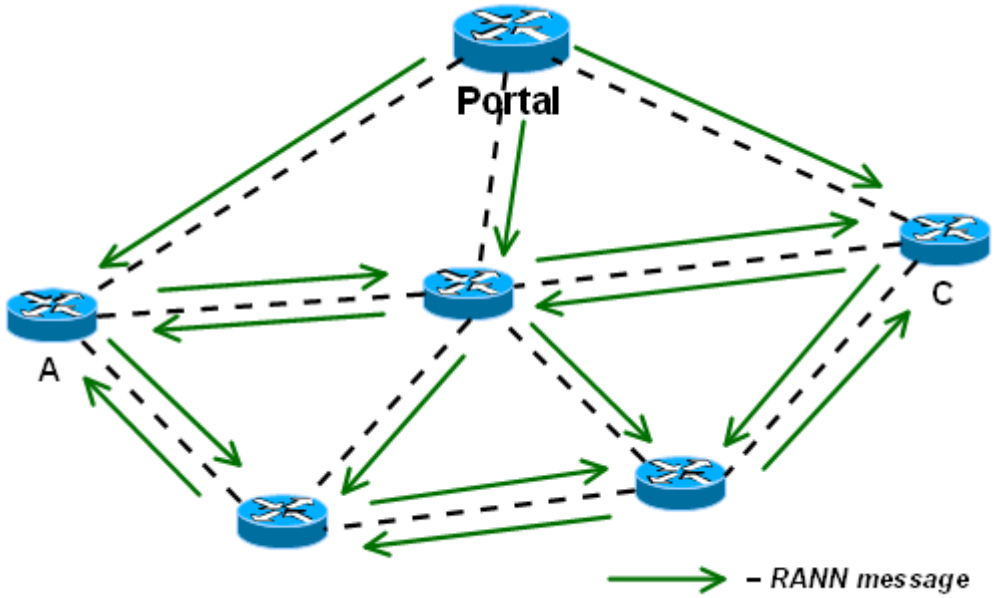
Router C 發送單播資料回復 A

在反應方式中 HWMP+是非常類似 AODV（Ad-hoc On-demand Distance Vector 按英文字面意思是自組網 按需 距離 向量網路，AODV 各移動節點並不持續維護即時描述整個網路拓撲的路由表，僅在有業務到達時才查找建立支持該業務交換的路由，從而節省了大量未必有效的路由管理控制開銷）。

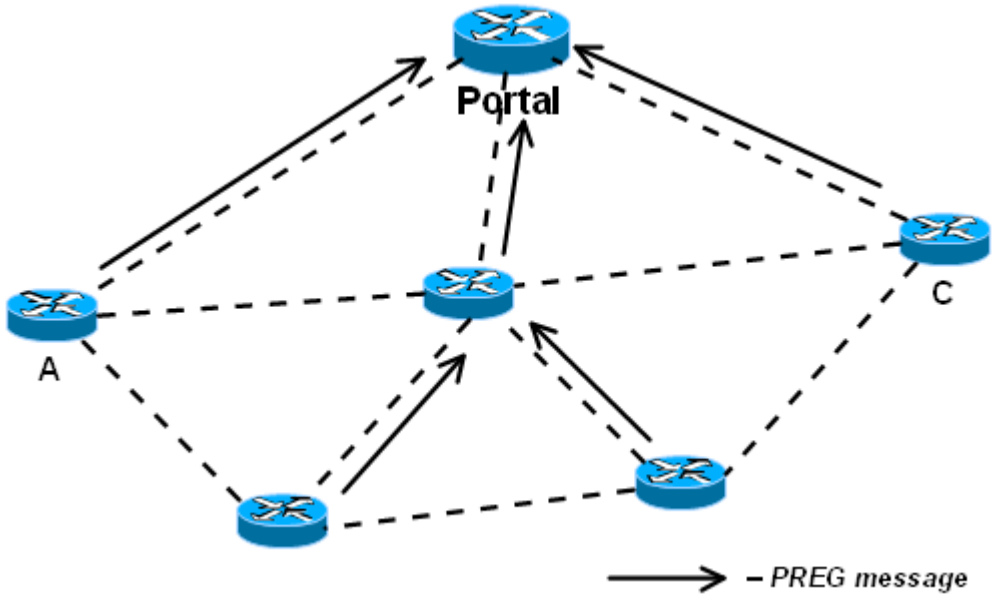
HWMP+在一經要求的情況下所有路徑被探測到，通過在網路中不斷發送路徑請求資訊(PREQ)。目的節點或者在路徑上的路由器會回復路徑資訊(PREP)。注意：如果目標位址屬於一個客戶終端，該 AP 會為下面的客戶終端提供代理(例如：答覆 PREQ 以他的名義)。

這種模式適用於移動網路，或大部分的通信發生的內部 mesh 節點。

2、主動方式



根節點通過不斷發送 RANN 資訊



內部節點回應 PREG

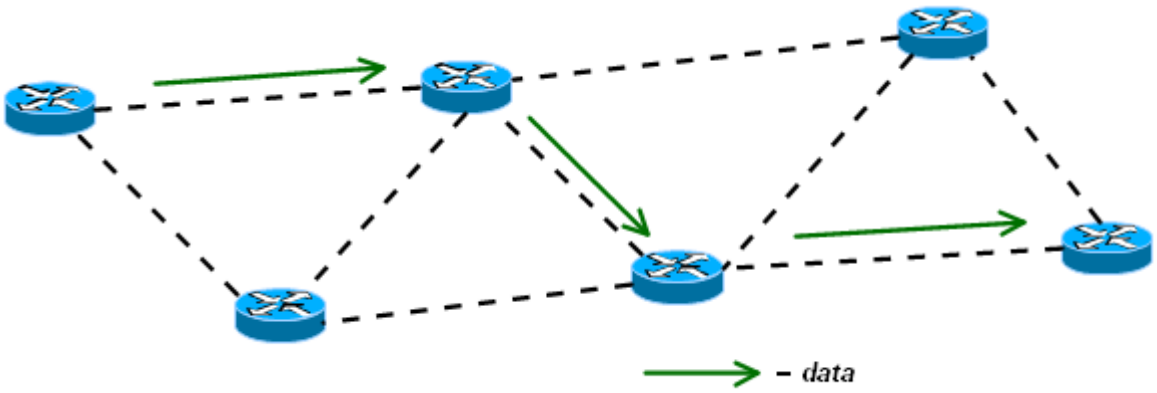
在主動方式下一些路由器配置為入口（protal），一般入口代表路由器有介面連接到其它的網路。

在網路中入口通過發送根消息（RANN）會宣佈他是出入點。內部節點會回應一個路徑註冊資訊（PREG），這樣的結果是入口節點作為路徑樹的根節點。

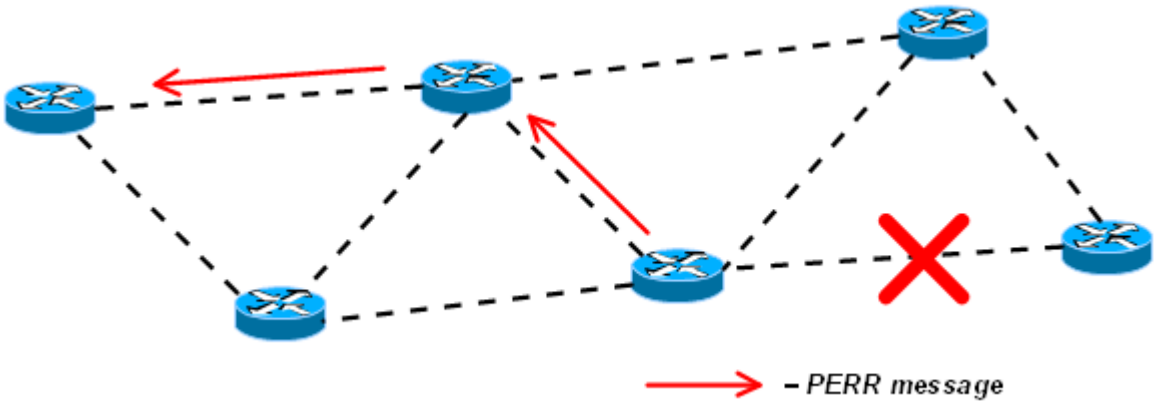
路徑為入口將充當一種默認路由，如果一個內部路由沒有找到指定的目的路徑，將會把所有資料發送到最近的入口節點，如果可能，入口將作為代理路由尋找路徑。這個可以引導向最佳的路徑，除非資料被指定到入口節點本身，或者一些已有的外部網路介面入口節點。

主動方式更適合當多數傳輸在兩個內部 Mesh 網路之間和存在多個入口節點。

3、拓撲變動探測



資料流程路徑



當連接消失，錯誤的上行資料

HWMP+ 使用路徑錯誤資訊(PERR)通告一個連接消失，這個資訊會發送到所有的上行資料流程節點返回到資料來源，源節點接收到 PERR 後會重啟路徑探測。

第十章 CAPsMAN

Controlled Access Point system Manager(CAPsMAN)，即 AP 控制系統， 能集中管理無線網路，根據需要也能做資料處理。在 CAPsMAN 系統中包含數個 AP（**CAPsMAN 代表集中管理系統，CAP 代表被管理 AP**），CAPsMAN 為他們提供無線連接管理，維護用戶端驗證和資料轉發

10.1 介紹

需要功能包：wireless-fp

當 CAP 被控制管理，將建立與 CAPsMAN 管理端的連接，通常 AP 自身控制的用戶端，將交由 CAPsMAN 管理包括用戶端的驗證等。CAP 僅僅是維護無線連接層面的加密和解碼，根據配置，資料將轉發到管理端進行集中處理，後再轉發到本地的 CAP。

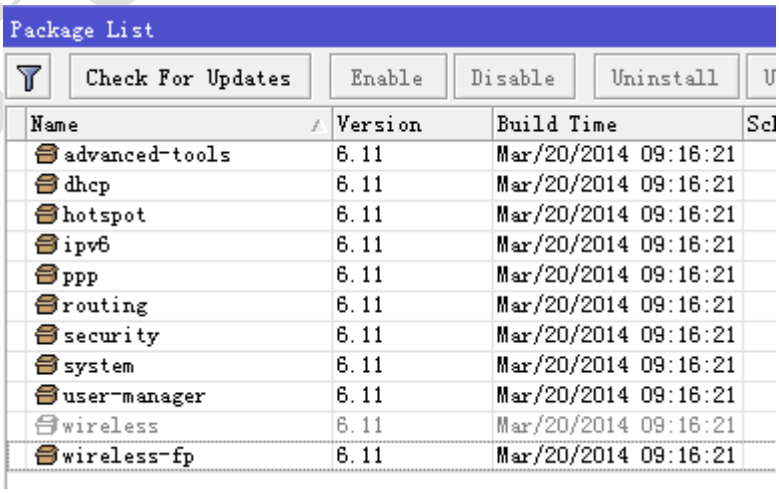
CAPs 管理功能

- RADIUS MAC 驗證
- WPA/WPA2 安全加密
- TBA

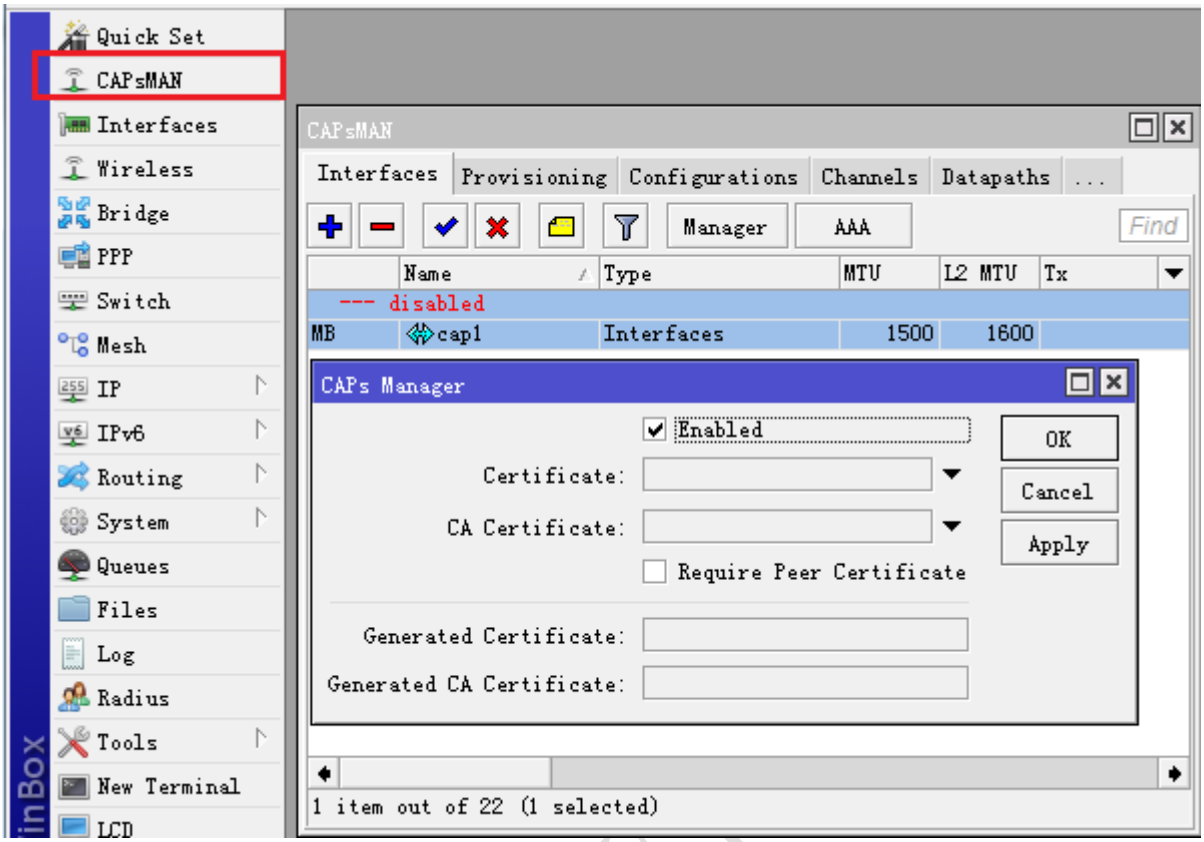
CAPs Manager 特點

- Nstreme AP 支持
- Nv2 AP 支持
- TBA

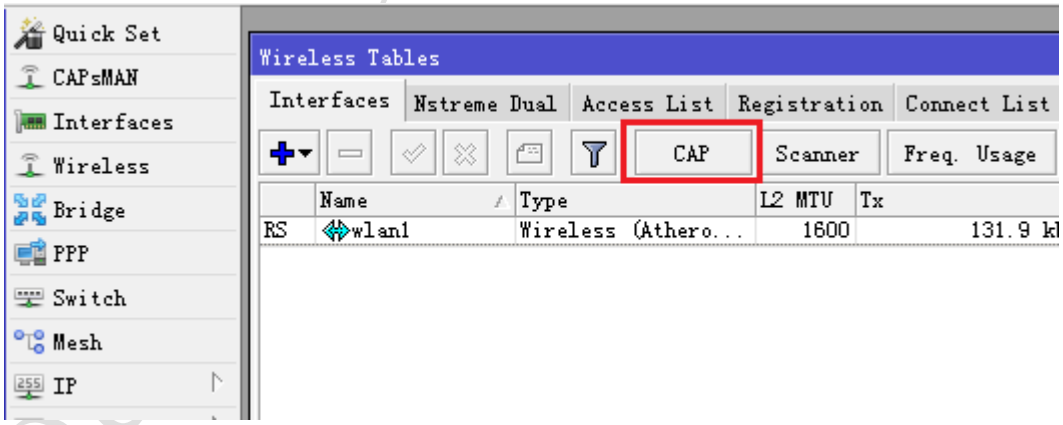
CAPsMAN 的功能包可以獨立安裝到任何平臺的 RouterOS v6.11 以上版本，CAPsMAN 功能還在測試中，所以功能包為獨立的 wireless-fp-6.11-mipsbe.npk，安裝後將替代原理的 wireless 功能包，功能包升級後 wireless 功能包被自動禁用，wireless-fp 啟用（該功能以後可能會合並）。



我們可以在 winbox 中看到 CAPsMAN 功能表：



進入 wireless 功能表，可以看到 CAP 選項



即 CAPsMAN 為 CAP 管理器功能表，CAP 為被管理網卡功能表

10.2 CAP 連接到 CAPsMAN

CAPsMAN 系統要為無線網路提供管理控制，需要至少一個 CAP 必須與 CAPsMAN 建立連接。一個管理連接建立可以使用 MAC 或 IP 層協議和安全的 DTLS。

通常一個 CAP 能傳遞用戶端資料連接到 CAPsMAN 管理器，但資料連接並不安全，因此需要考慮資料安全的加密，例如 IPSec 或其他加密隧道。CAP 連接到 CAPsMAN 過程如下：

1、CAP 連接到 CAPsMAN 能基於二層和三層（MAC 層和 IP 層）建立連接

MAC 層連接特性：

- 沒有 IP 配置到 CAP
- CAP 和 CAPsMAN 必須在相同二層網路中，二層交換或虛擬網路（二層隧道，例如 EoIP）

IP 層連接（UDP）特性：

- 如果需要能穿透 NAT
- CAP 必須通過 IP 協議連接到 CAPsMAN

注意：基於三層連接時，如果 CAP 與 CAPsMAN 沒有在相同二層網路，必須為 CAPsMAN 分配 IP 位址，且兩端路由可達。

2、為了與 CAPsMAN 建立連接，CAP 會執行一個探測操作，在探測週期裡，CAP 會試圖連接 CAPsMAN，並創建一個可以運行的 CAPsMAN 列表。CAP 連接一個可運行的 CAPsMAN 會進行如下操作：

- 配置管理 IP 位址
- 從 DHCP 伺服器獲取 CAPsMAN IP 地址
- 配置介面能通過 IP 和 MAC 層廣播

當獲取 CAPsMAN 列表建立後，CAP 選擇一個 CAPsMAN 連接基於以下規則：

- 如果 **caps-man-names** 參數指定管理名稱（**/system identity** 作為 CAPsMAN）CAP 將優先選擇。如果該參數為空，將連接其他 CAPsMAN。
- MAC 層連接優先順序高於 IP 層連接

3、當 CAPsMAN 管理被選擇後，CAP 嘗試建立 DTLS 連接，以下是驗證模式：

- 沒有證書在 CAP 和 CAPsMAN，不用驗證
- 只有 CAPsMAN 管理器能配置證書 - CAP 會檢查 CAPsMAN 的證書，這時 CAPsMAN 必須配置為 **require-peer-certificate=no**，CAP 無需配置證書，即可以驗證
- CAP 和 CAPsMAN 都配置證書，證書相互驗證

4、在 DTLS 連接建立後，CAP 會檢查 CAPsMAN 證書提供的 **CommonName** 欄位 caps-man-certificate-common-names，這個欄位參數列表包含被允許的 CommonName 值。如果這個清單不為空，CAPsMAN 必須配置證書，如果清單為空，CAP 不會檢查 CommonName 欄位。

CAP 自動鎖定 CAPsMAN

CAP 能配置自動鎖定特定的 CAPsMAN， 鎖定被執行通過記錄 CAPsMAN 的證書 CommonName 欄位，CAP 鎖定並檢查所有後續的 CommonName 連接 ，這個功能執行使用證書的 CommonName 欄位。

鎖定功能啟用通過一下命令：

```
[admin@CAP] > /interface wireless cap set lock-to-caps-man=yes
```

10.3 Datapath 配置

Datapath 設置對各 CAP 資料相關的轉發，在 CAPsMAN datapath 中設置 datapath 策略，路徑為 /caps-manager datapath 或者直接在 configuration 中設置，也能在 interface 中配置

Datapath 有兩個主要轉發模式：

- local forwarding mode（本地轉發模式），即由 CAP 通過無線本地介面轉發資料
- manager forwarding mode（管理轉發模式），CAP 將所有接收到資料發送到 CAPsMAN，CAPsMAN 處理後再發送資料到 CAP。在這個模式包括用戶端到用戶端之間轉發都通過 CAPsMAN 控制

轉發模式配置針對的是每張無線網卡，而非每台設備，因此如果一個 CAP 有 2 張無線網卡，一張可頤配置為本地轉發模式，一直為管理轉發模式。這也同樣適用於 Virtual-AP 網卡，不管是 Virtual-AP 還是其主無線網卡（master-interface）都可以選擇不同的轉發模式

大多情況下 datapath 設置會選擇管理轉發模式，因為本地轉發模式 CAPsMAN 不能統一控制 CAP 的資料轉發，對於統一的 CAP 群控制和使用者資料轉發有好處

下面是 datapath 設置：

- bridge - 添加到指定的 bridge 介面，即將 CAP 加入指定的橋接，前提是 bridge 介面已經配置
- bridge-cost - 當 bridge 端口添加，可設置該端口成本開銷值
- bridge-horizon - 當 bridge 端口添加，可設置水準分割橋接預防橋接環路
- client-to-client-forwarding - 用戶端到用戶端連接到無線網卡後，用於控制他們之間的資料轉發，在本地轉發模式該功能由 CAP 自行處理，管理轉發模式則有 CAPsMAN 處理
- local-forwarding - 控制轉發模式
- openflow-switch - 當 openflow 協定配置後，可添加到 OpenFlow 交換器中
- vlan-id - 當 vlan 模式啟用，且用 vlan 標籤，這時設置的 vlan id 將分配到網卡介面上
- vlan-mode -指定 vlan 模式，即 vlan 設置為打標籤、還是不打標籤

Local Forwarding 模式

在這個模式下無線網卡作為 CAP 處理類似于一個普通的 AP，並直接將正常資料轉發。CAPsMAN 將不參加資料轉發和處理任何資料幀，只控制無線網卡的配置和用戶端連接和分配處理

當無線網卡啟用為 CAP，配置將會被修改，即狀態和一些相關參數會與 CAPsMAN 相關聯（例如：mac-address、arp 和 mtu），注意無線網卡關聯的配置不會應用到實際的無線網卡配置中，而是通過 CAPsMAN 控制 ：

```
[admin@CAP] /interface wireless> pr
Flags: X - disabled, R - running
 0  R  ;;; managed by CAPsMAN
       ;;; channel: 5180/20-Ceee/ac, SSID: master, local forwarding
      name="wlan2" mtu=1500 mac-address=00:03:7F:48:CC:07 arp=enabled
      interface-type=Atheros AR9888 mode=ap-bridge ssid="merlin"
      frequency=5240 band=5ghz-a/n channel-width=20/40mhz-eC scan-list=default
      ...
```

Virtual-AP 網卡在 local forwarding 模式下將顯示為啟用和動態的 Virtual-AP 網卡：

```
[admin@CAP] /interface> pr
Flags: D - dynamic, X - disabled, R - running, S - slave
 #     NAME                                TYPE         MTU L2MTU  MAX-L2MTU
 ...
 2  RS  ;;; managed by CAPsMAN
        ;;; channel: 5180/20-Ceee/ac, SSID: master, local forwarding
       wlan2                               wlan        1500  1600
 3 DRS  ;;; managed by CAPsMAN
        ;;; SSID: slave, local forwarding
       wlan6                               wlan        1500  1600
 ...
[admin@CAP] /interface> wireless pr
Flags: X - disabled, R - running
 ...
 2  R  ;;; managed by CAPsMAN
       ;;; SSID: slave, local forwarding
      name="wlan6" mtu=1500 mac-address=00:00:00:00:00:00 arp=enabled
      interface-type=virtual-AP master-interface=wlan2
```

事實上 Virtual-AP 網卡是動態添加，某些時候設置為靜態有助於 CAP 資料轉發，例如分配 IP 位址到 Virtual-AP 網卡，這樣的配置不會應用到 Master 無線網卡

為有助於資料轉發配置，CAP 能配置到一個 bridge 中，且 bridge 能自動添加 CAP 端口到橋接交換中，當然前提是 CAPsMAN 的 bridge 已經啟用，這個配置在**/interface wireless cap** 功能表下配置

Manager Forwarding 模式

在這個模式 CAP 發送所有無線到接收/發送資料到 CAPsMAN，CAPsMAN 控制所有的 CAP 的配置和轉發資料，包括用戶端到用戶端之間的轉發，Virtual-AP 網卡可以選擇禁用 CAP 模式.

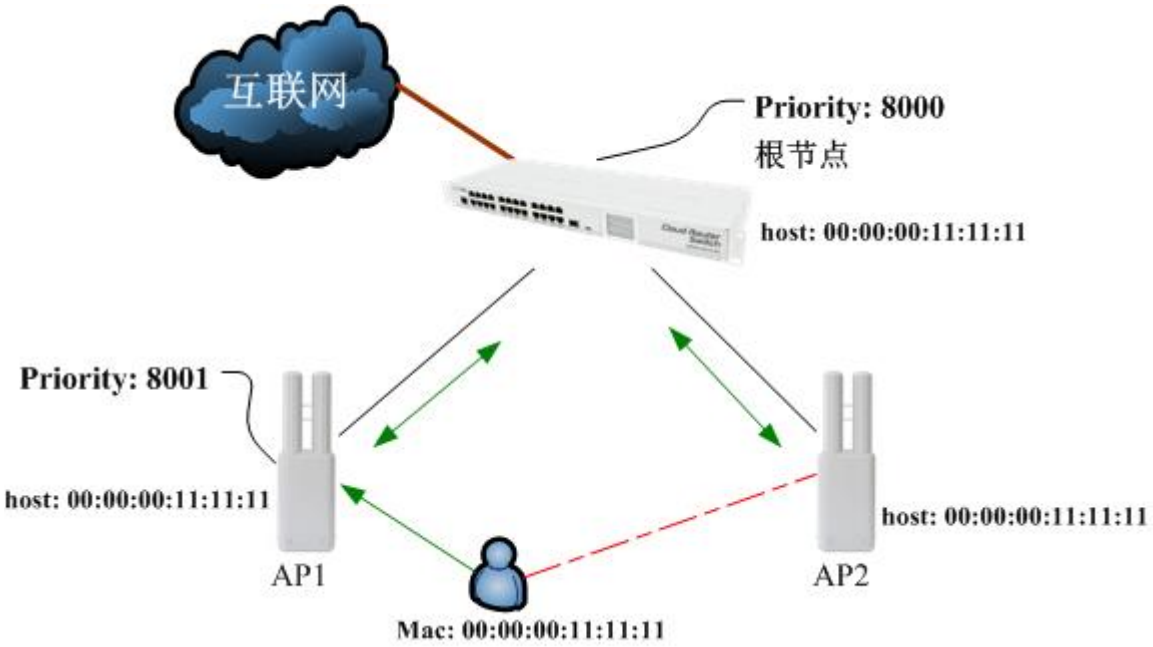
```
 ...
 1 X   ;;; managed by CAPsMAN
       ;;; channel: 5180/20-Ceee/ac, SSID: master, manager forwarding
      name="wlan2" mtu=1500 mac-address=00:03:7F:48:CC:07 arp=enabled
      interface-type=Atheros AR9888 mode=ap-bridge ssid="merlin"
 ...
```

Datapath 實例

關於 Datapath 的使用，這裡我們以 Manger Forwarding mod（管理轉發模式）來講解下，管理轉發模式不僅僅是希望統一管理 AP，也希望通過各個 CAPsMAN 將各個無線訪問節點的用戶端進行集中控制和漫遊等，特別是漫遊方面。其實 CAPsMAN 的 datapath 可以實現這部分，但仍然通過建立 bridge 後將用戶都放到一個二層交換下，通過 RSTP 的模式來完成。

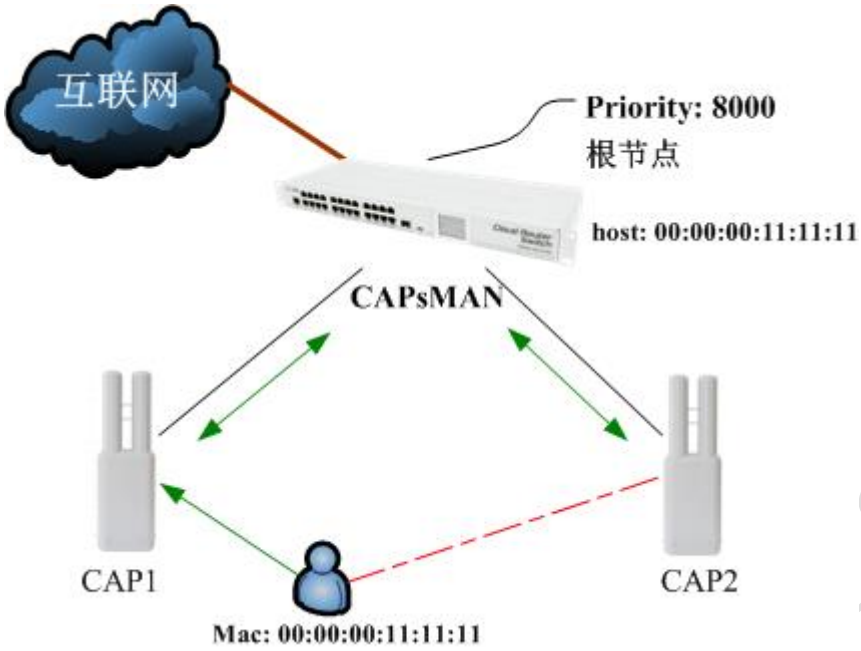
通常我們建立可漫遊的無線網路，是才採用透明橋的 RSTP 網路，即將所有 AP 設置為透明橋接，連接到一個根節點的橋下，根節點的橋作為三層閘道，並通過 DHCP 分配 IP 地址，或建立 nat 和熱點閘道等等。

如下圖，兩個 AP，分別都是獨立的，都建立一個 bridge 和自己的優先順序，有自己的 hosts 列表，當用戶 mac：00:00:00:11:11:11 連接後，這個二層網路沒劃分 VLAN，所以每個設備都能學習到該用戶 MAC，這樣當使用者在無線網路中切換時，使用者幾乎無感覺基本就在 1～2 個包的延遲（注意無線漫遊切換選擇取決於用戶終端設備，而非 AP）



而現在 CAPsMAN 也是類似的 RSTP 網路，但不是每個 AP 都有自己的 bridge 和 hosts 列表，由於建立了 CAPsMAN 管理，下面的 CAP 會將所有無線資料發送到 CAPsMAN 處理，所以 bridge 只有一個，就是 CAPsMAN 管理器本身，可以理解為 CAPsMAN 就是一個大交換機。

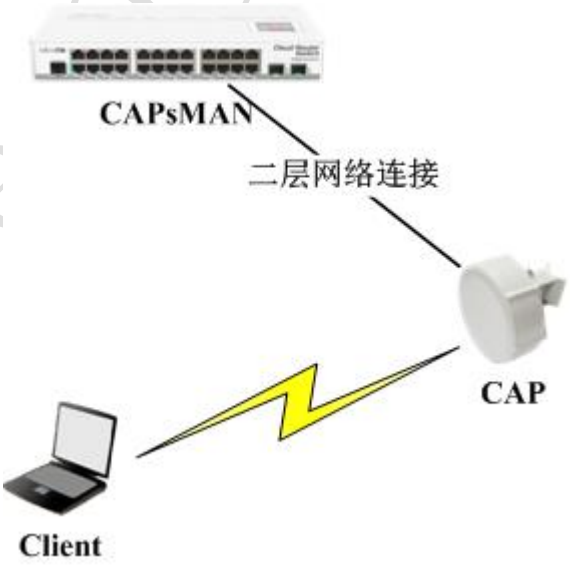
如下圖，CAP1 和 CAP2 除了建立 WiFi 的無線連接，沒有其他任何操作，bridge 建立在 CAPsMAN 上，即 hosts 管理則由 CAPsMAN 完成。



可以理解為，以前的 AP 網路是，每個 AP 都是一個交換機，都上聯到上層的一台彙聚交換機，而現在的 CAPsMAN 則是所有無線終端都連接這台彙聚交換機，中間就不存在其他任何“AP 交換機”。

10.4 CAPsMAN 實例

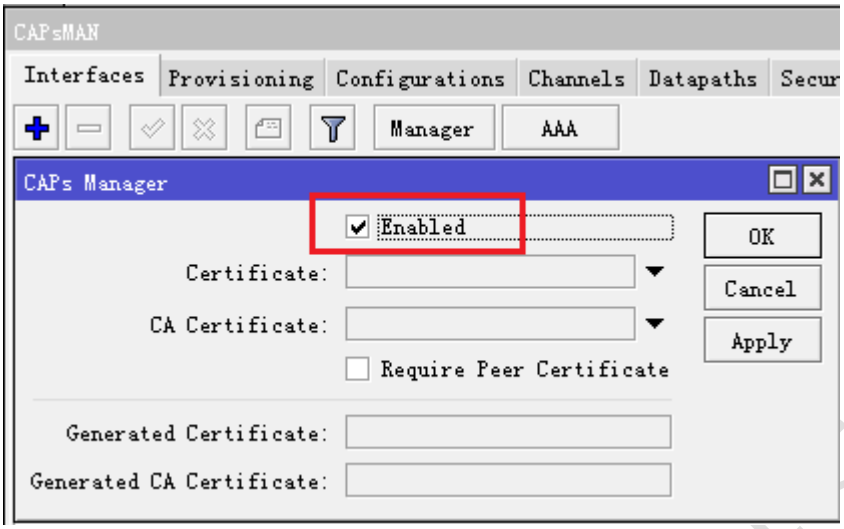
下面是一個簡單的 CAPs 聯網實例，假設我們網路中只有一個 CAPsMAN 和一個 CAP，他們之間通過二層網路連接，即採用二層 MAC 建立互連，無證書驗證。



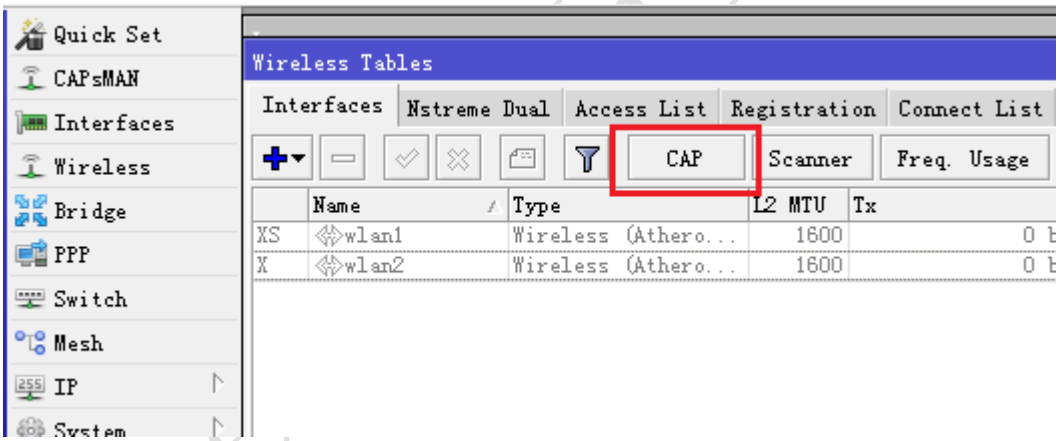
之前的介紹 CAP 與 CAPsMAN 連接，可以通過 MAC 層和 IP 層連接，CAP 也具備自動搜索 CAPsMAN 功能，也可以通過 caps-man-names 連接，這裡 CAPsMAN 與 CAP 連接基於二層，且這個二層網路僅有一台 CAPsMAN，所以我們的建立變的簡單。

啟用 CAPsMAN

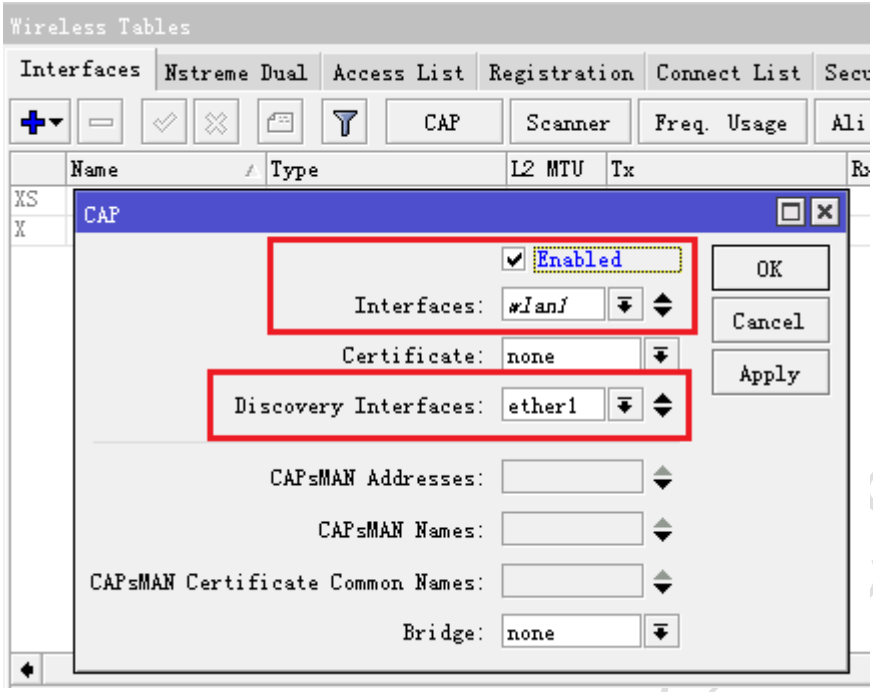
由於 CAPsMAN 與 CAP 基於二層網路互連，且僅只有一個 CAPsMAN，所以我們僅啟用 CAPsMAN，CAPsMAN 功能可以安裝到任何 RouterOS 平臺上，我用 RB750 作為 CAPsMAN 管理器。



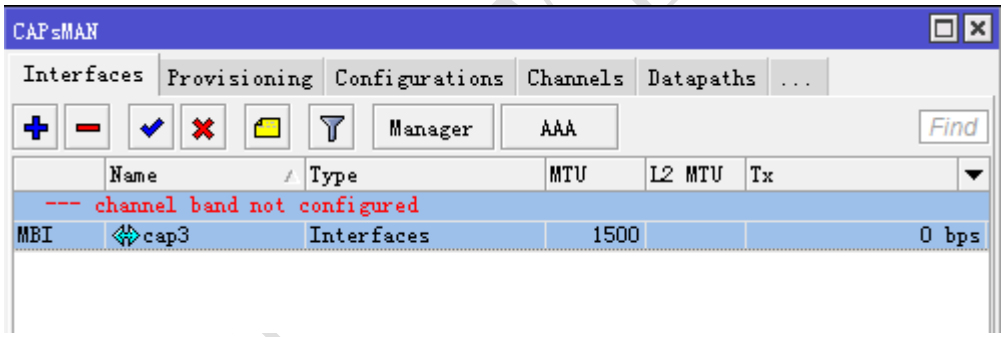
啟用 CAPsMAN 後，需要將 CAP 連接到 CAPsMAN 上，這裡我們是一台 RB411，通過 ether1 連接到 RB750，首先我們需要在 CAP 上開啟功能，進入 wireless 選項，選擇 CAP 功能表



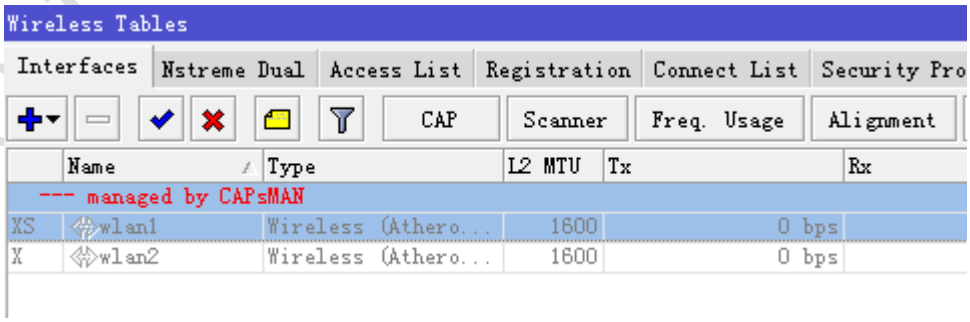
進入 CAP 功能表後，我們開啟 CAP 功能，選擇 wlan1 為被管理網卡，探測 CAPsMAN 網卡為 ether1



配置完成後,CAP 開始自動搜索二層網路內的 CAPsMAN 管理器，大約幾秒鐘後 CAP 連接上 CAPsMAN，如下圖

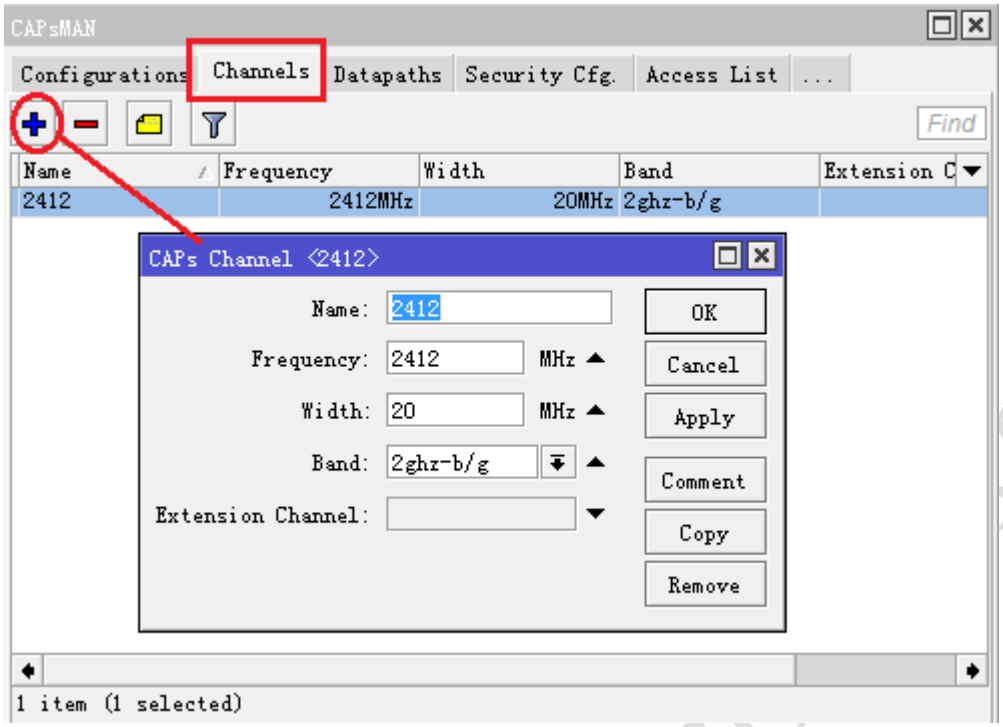


但當前標識狀態看 cap3 為 MBI，M 代表主設備，B 代表被綁定，I 代表未啟動，因為 cap3 沒有配置無線相關參數，我們再來看看 CAP 的狀態

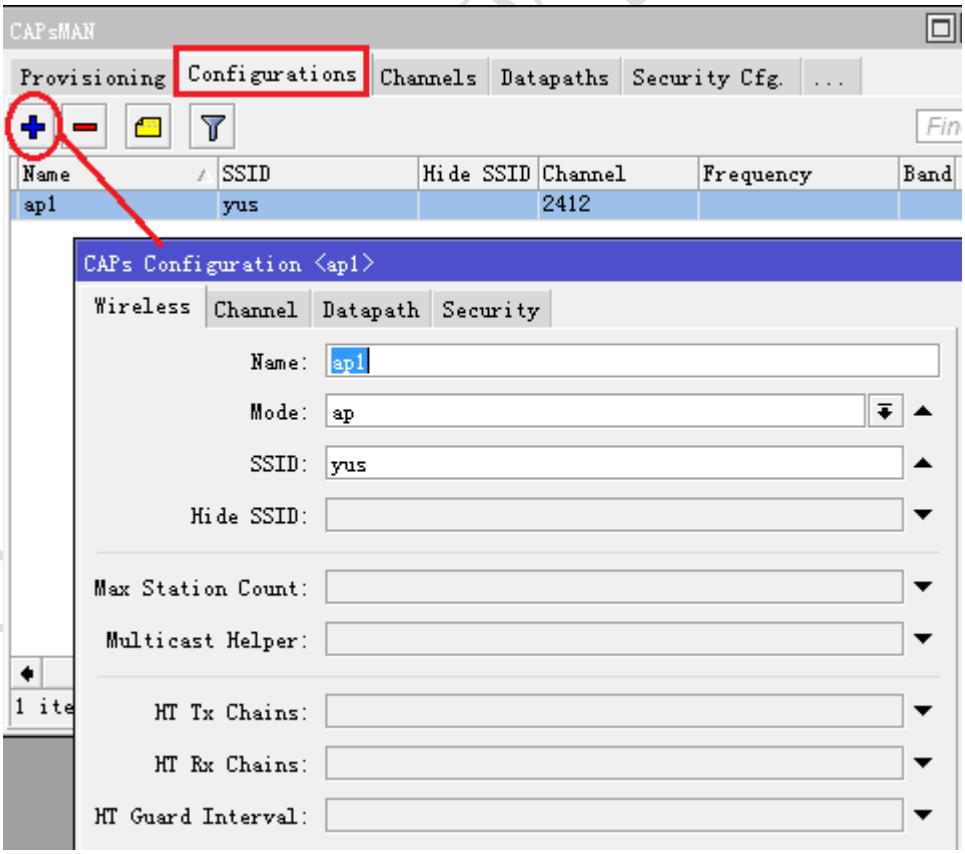


從上圖中可以看到 managed by CAPsMAN，即被 CAPsMAN 管理，wlan1 無線網卡處於禁用狀態，是無法被本地的 RouterOS 管理。

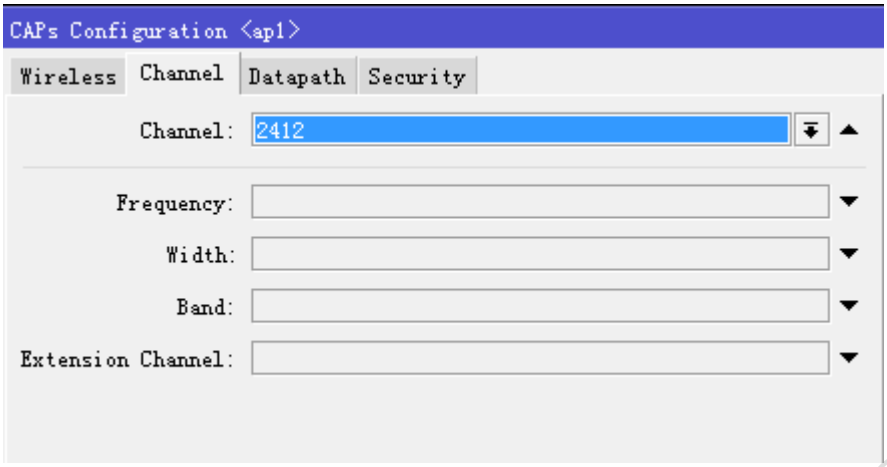
跟著我們要為 CAPsMAN 連接的 CAP 配置無線參數，連接的 wlan1 是一張 802.11bg 網卡，我們需要在 CAPsMAN 配置相關的 bg 無線參數，首先定義頻道，取名 2412，頻段 2ghz-b/g，發射頻率 2412MHz，頻寬 20MHz。



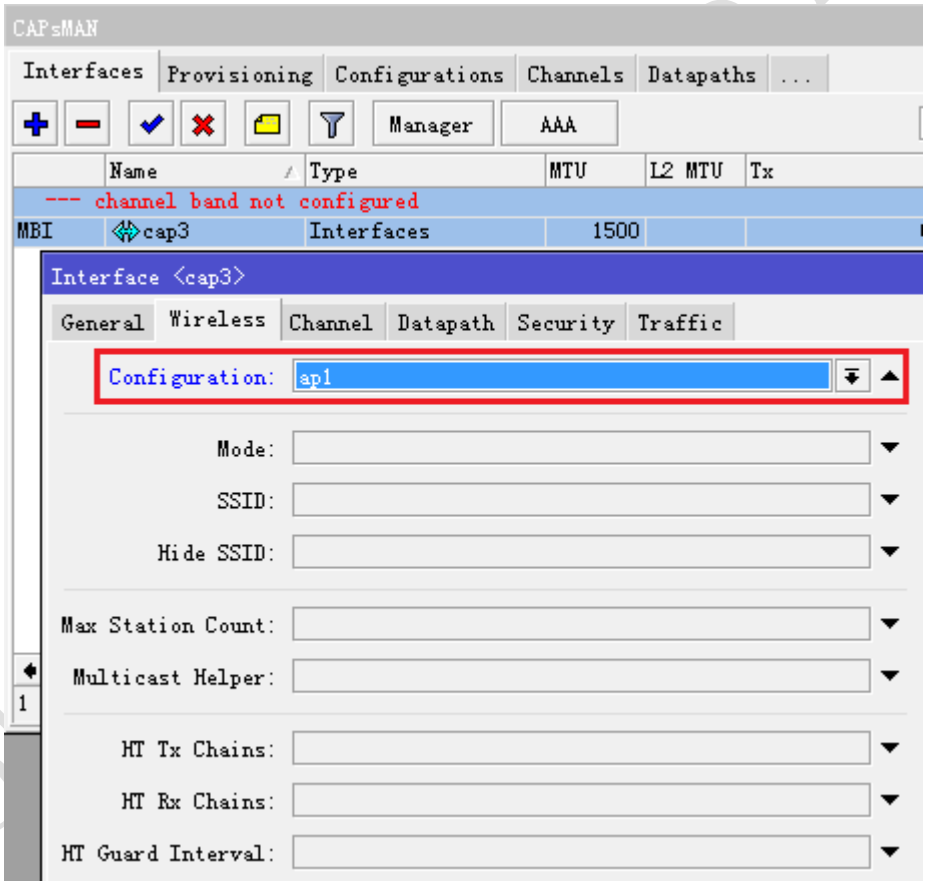
定義配置規則組，進入 configurations 下，添加一個規則組取名 ap1，mod 為 ap，SSID 設置為 yus



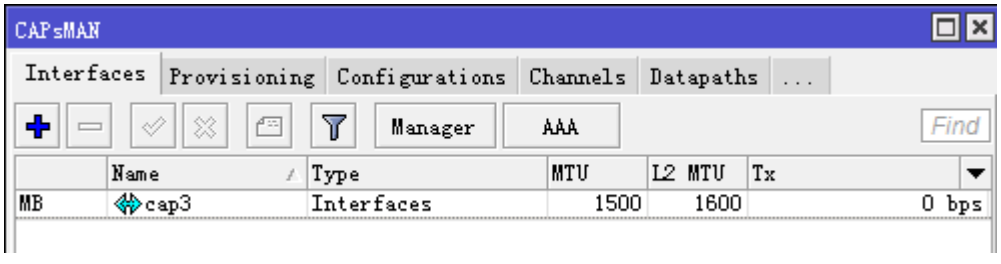
選擇 Channel 設置頻率，我們將之前設置的 2412 頻率選擇上



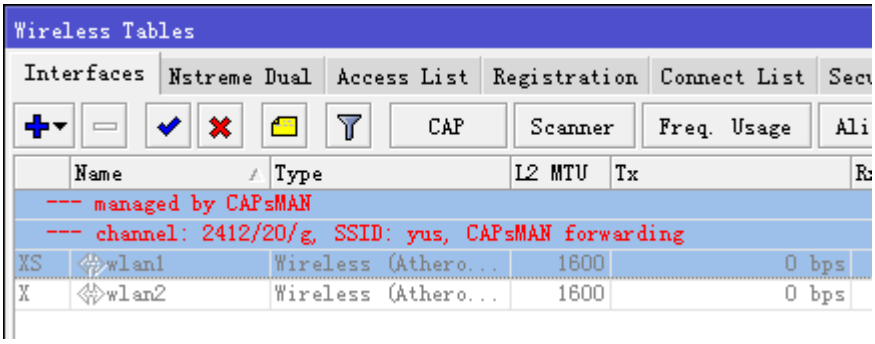
當配置規則組完成後，返回 interface 功能表下，選擇 cap3，進入 wireless 功能表，直接選擇 configuration 為 ap1



配置完成後，我們可以看下 cap3 的狀態，已經沒有 I 標識



再看看 CAP 狀態，清楚的顯示 wlan1 的頻率和 SSID 等資訊



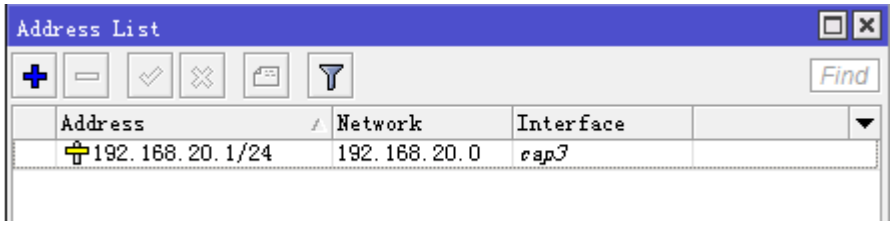
通過電腦搜索到 yus 的無線信號，沒有設置加密所以不安全

這樣基本的無線已經配置完成，但對於這個無線的 CAPsMAN 系統僅僅完成一半，通常情況下我們需要為用戶端分配 IP，不管是熱點認證還是直接上網，都要讓用戶端自動獲取 IP（特殊環境除外，如固定 IP），下一步肯定是要做 DHCP 伺服器，

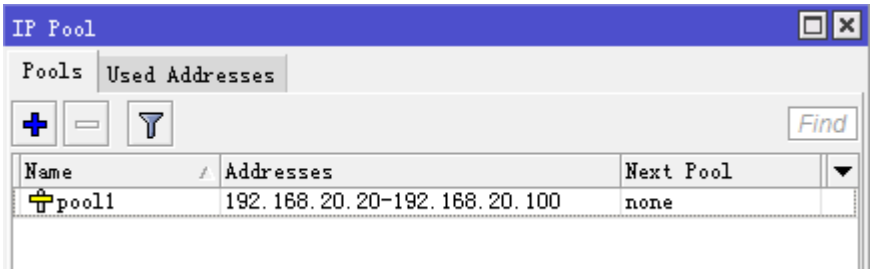
DHCP 伺服器不是在 CAP 上做，而是在 CAPsMAN 上，因為 CAP 的 wlan1 網卡不在接受本地 RouterOS 的管理，已經從屬於 CAPsMAN 的 RouterOS 上，cap3 已經是 CAPsMAN 的 RouterOS 的一張虛擬無線網卡，所以我們的 DHCP 伺服器是在 CAPsMAN 上完成。

DHCP 配置就簡單過下：

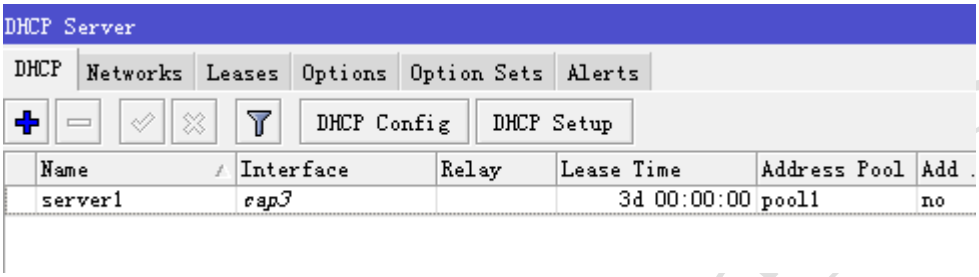
在 ip address 中為 cap3 分配 ip 地址 192.168.20.1/24



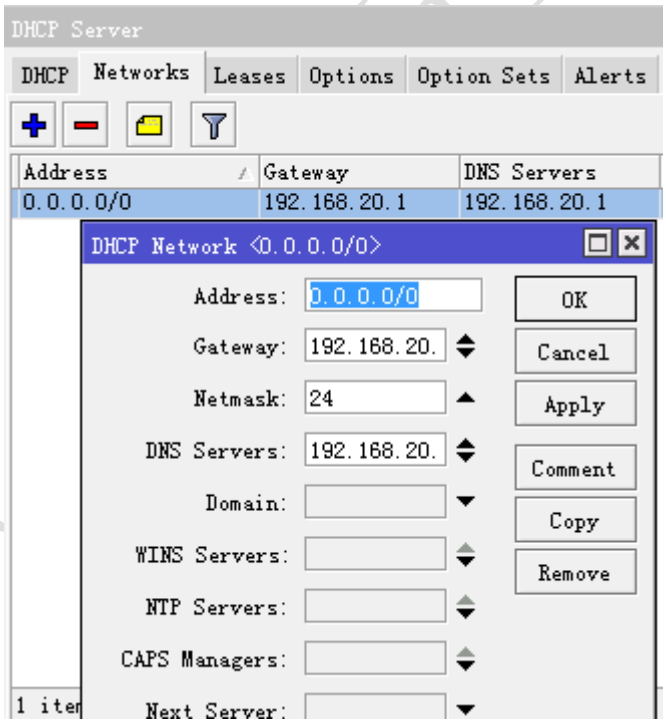
在 ip pool 建立位址集區



建立 DHCP 伺服器到 cap3



設置 DHCP 網路獲取的閘道、子網和 DNS



DHCP 建立完成後，我們可以獲取到 cap3 無線分配的 IP 位址

第十一章 WLAN 的認證服務應用

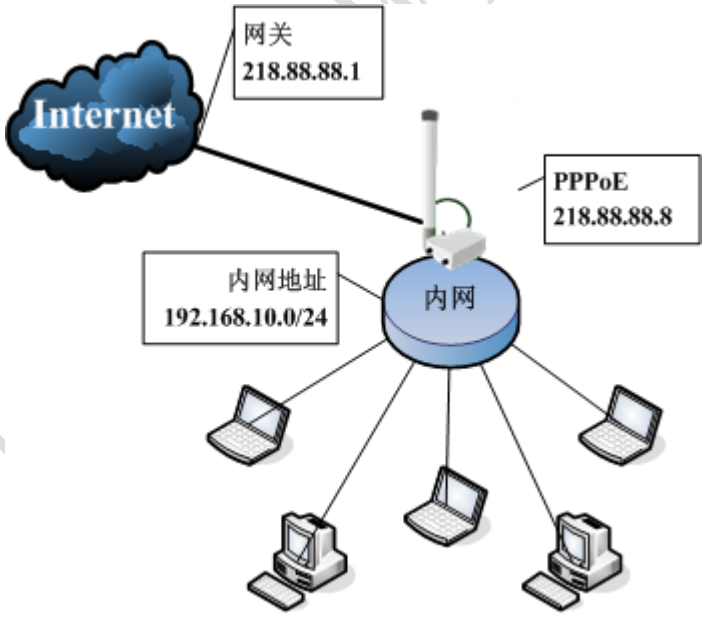
在 WLAN 中我們可能會涉及到網路認證功能，如 PPPoE 認證和 Hotspot 熱點認證等，在這些應用中都是要求客戶通過帳號密碼驗證，才能連接到互聯網或者訪問指定的資料。這裡我們主要介紹 PPPoE 和 Hotspot 的認證方式。

注：PPPoE 主要基於二層鏈路認證，要求 WLAN 網路基於橋接模式傳輸的，如中間有路由結構的 WLAN 的設備，就無法透傳到路由後的網路。Hotspot 則可以基於二層和三層網路的傳輸，

什麼地方選擇什麼樣的認證方式，需要特定的考慮，當前主流的認證方式是 Hotspot 熱點認證，因為這種認證採用基於 web 的 http 驗證更加方便快捷，特別適合在人流量大的地方選擇 Hotspot 熱點認證，如酒店、圖書館、咖啡廳、公園、火車站和機場等，而相對固定的網路可以考慮 PPPoE 認證，但 PPPoE 認證對 pad 或手機終端設備有影響，因為他們默認是不支持 PPPoE 撥號的，這就需要對特點的網路環境進行考察評估，選擇相應的認證方式。（關於 PPPoE 和 Hotspot 認證詳細配置可以參考**《RouterOS 中文網路教程》**）

11.1 基於 PPPoE 的 WLAN 認證

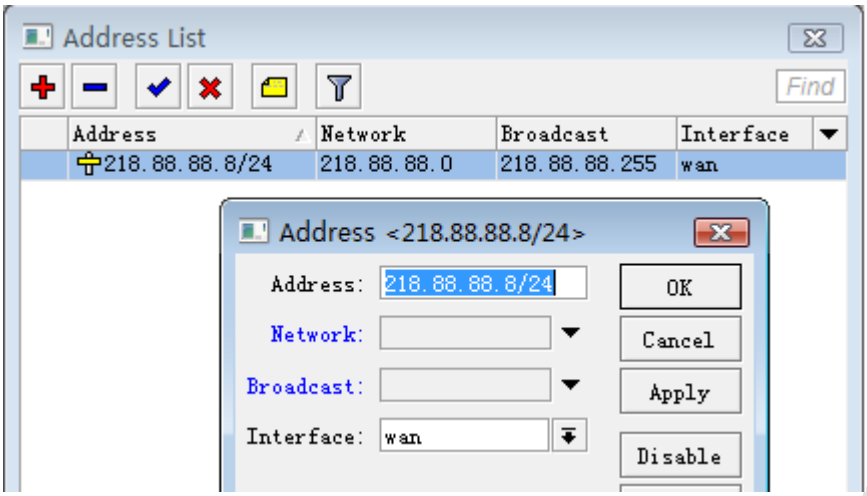
建立 PPPoE 認證我們一般是通過 WDS 橋接模式構建 WLAN 網路，因為只有二層鏈路才能透傳 PPPoE 資料，構建網路如下：



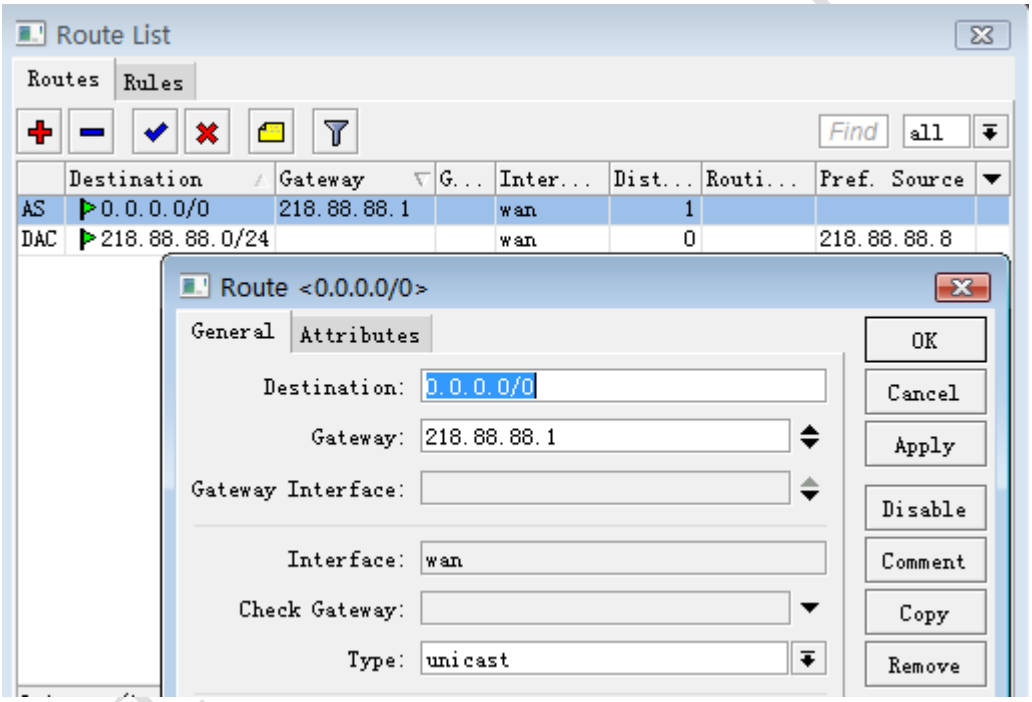
wan 口的外網連接 IP 位址是 218.88.88.8/24，內網因為是 PPPoE 撥號認證，我們可以不用在內網介面設置 IP 位址，只需要建立 PPPoE 伺服器分配建立隧道連接的 IP 即可，構建 PPPoE 的 WLAN 認證網路步驟如下：

1、 配置基本網路參數 IP 位址、閘道、nat 和位址集區，並配置 wlan 的連接；
2、 在 PPP 中添加 PPPoE 伺服器；
3、 配置 PPPoE 的 Profiles 使用者組規則，並在 Secrets 中添加用戶帳號；
4、 測試 PPPoE 撥號連線。

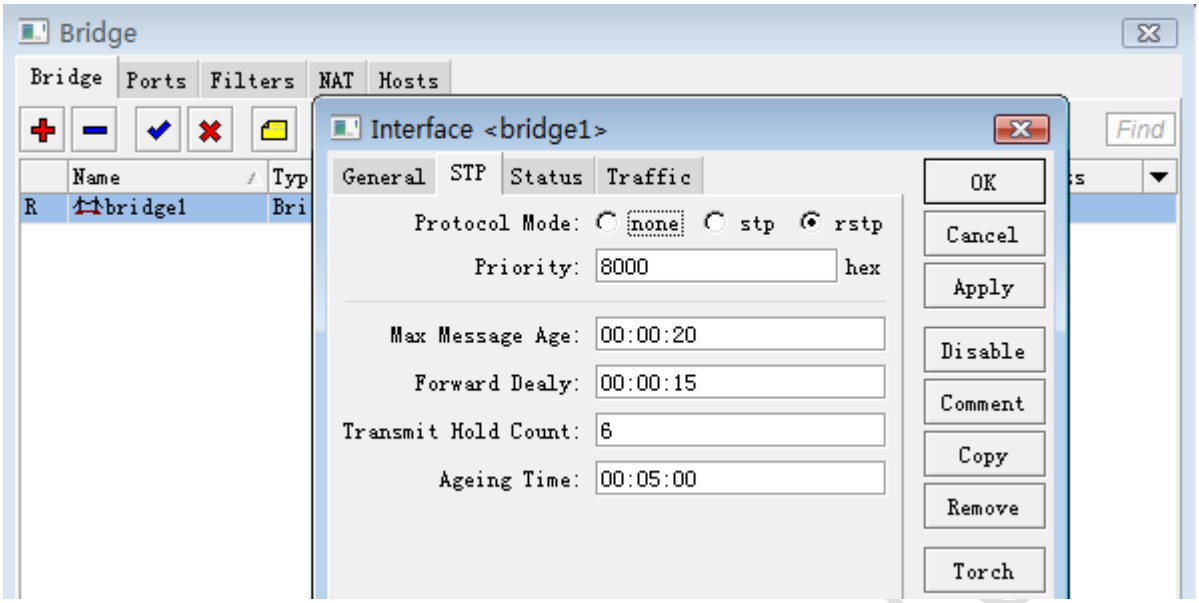
步驟 1：我們先添加 wan 口的外網 IP 地址：



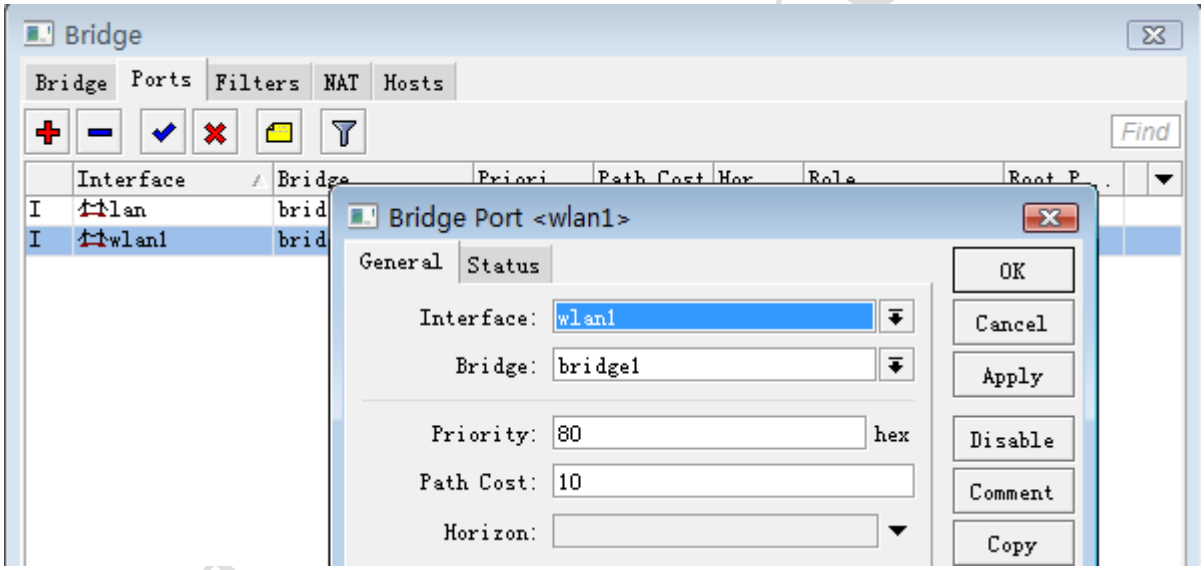
接著我們進入 ip route 配置路由，閘道為 218.88.88.1



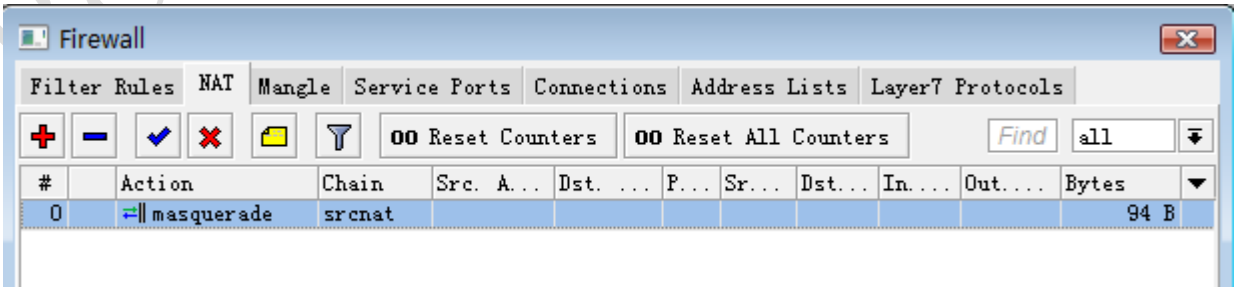
進入 bridge 在我們添加橋接，並設置 rstp 的參數，將 lan 口和 wlan1 設置到 bridge1 中：



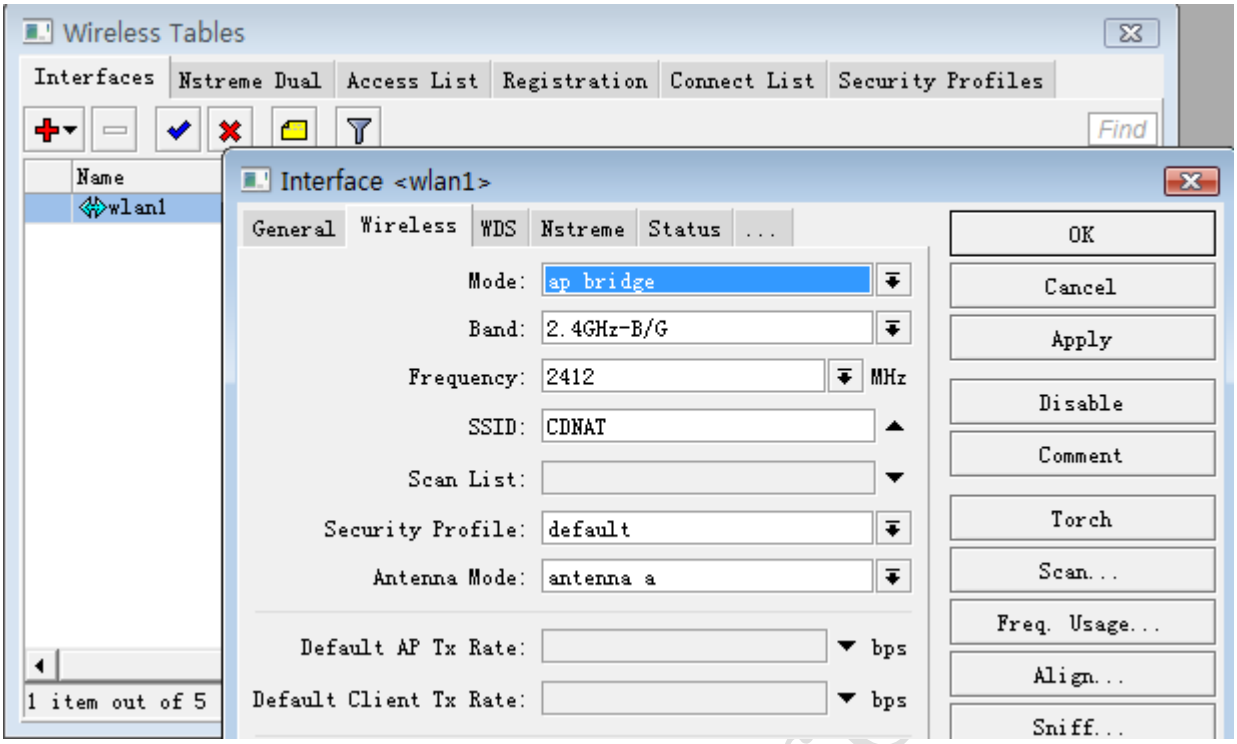
將 lan 和 wlan1 添加如 bridge1 中：



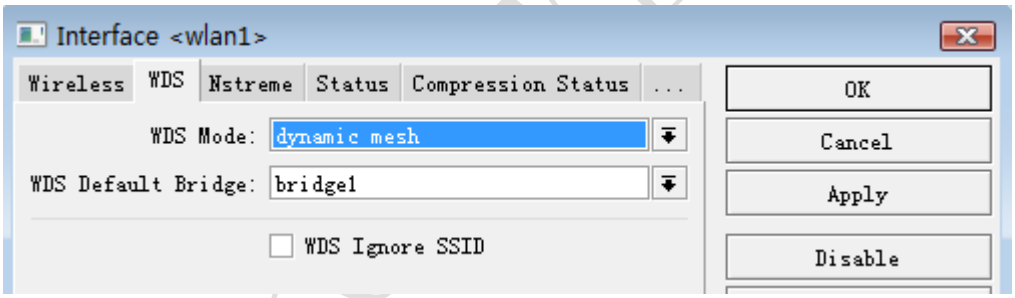
之後我們進入 ip firewall nat 配置 src-nat 的偽裝策略 action=masquerade，用於隱藏內部的私有 IP 位址連接上網，如下圖：



接下來配置 wlan1 的無線模組，我們這裡以 2.4G-bg 為主，配置 mode=ap-bridge、Band=2.4G-B/G、Frequency=2412、SSID=CDNAT、WDS-Mode=dynamic-mesh、wds-default-bridge=bridge1，配置如下圖：



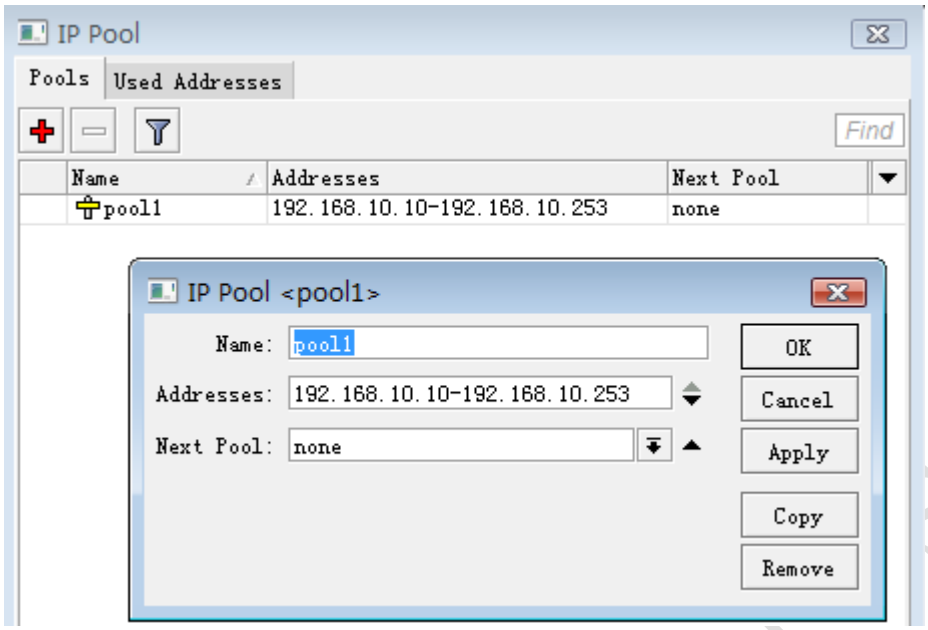
在 wlan1 配置 WDS 選項，設置 WDS-Mode=Dynamic-Mesh，比將其添加到 bridge1 中：



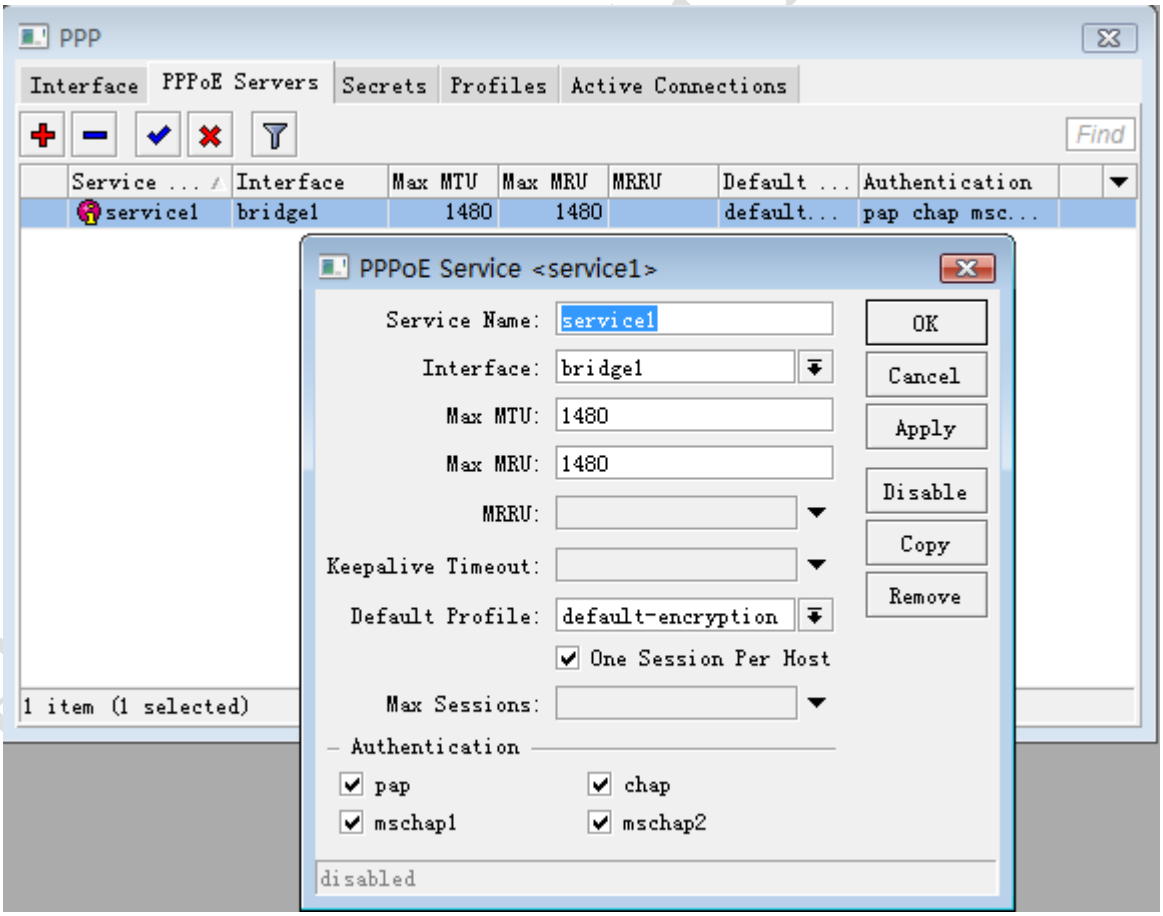
當 WLAN 無線配置完後，配置 PPPoE 服務，PPPoE 伺服器參數如下：

1、 用戶分配地址段：192.168.10.10-192.168.10.253
2、 用戶閘道地址：192.168.10.1
3、 DNS 伺服器：61.139.2.69
4、 用戶頻寬為：下行：2Mbps，上行：1Mbps

首先我們需要進入 ip pool 中添加 PPPoE 分配給用戶的位址集區：



步驟 2：配置完基本參數後，現在進入 ppp 的 PPPoE-server 選項，啟用 PPPoE 伺服器，將伺服器的 interface 設置到 bridge1 上，Default-Profile 選擇 default-encryption，並將 one-session-per-host 打上勾，驗證方式 Authentication 都選擇上：



步驟 3：進入 profile 中配置 profiles 規則，這個是配置使用者組規則，即不同的使用者類型分配一個 profile 類型，這裡我們使用預設的 default-encryption 規則

1、 Local-address 為本地路由器的閘道地址，
2、 Remote-address 為給用戶端分配的 IP 地址，

3、 DNS-server 填寫相應的 DNS 伺服器

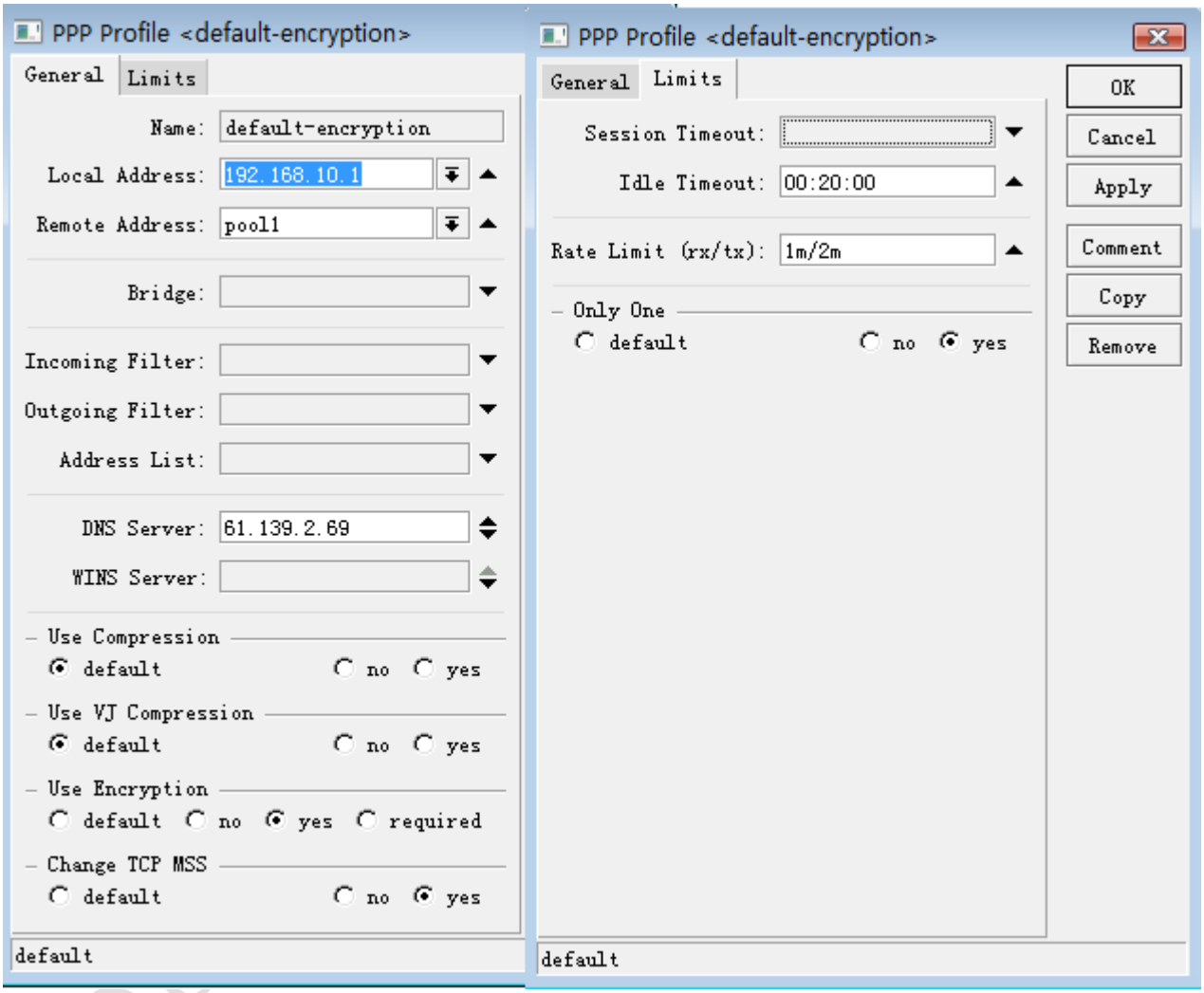
4、 User-Encryption 使用加密方式，在 windows 下預設是要求 PPPoE 撥號是加密的

5、 Idle-Timeout：空閒超時時間，即當用戶端在一段時間內沒有任何資料流程量後，就斷開連接。

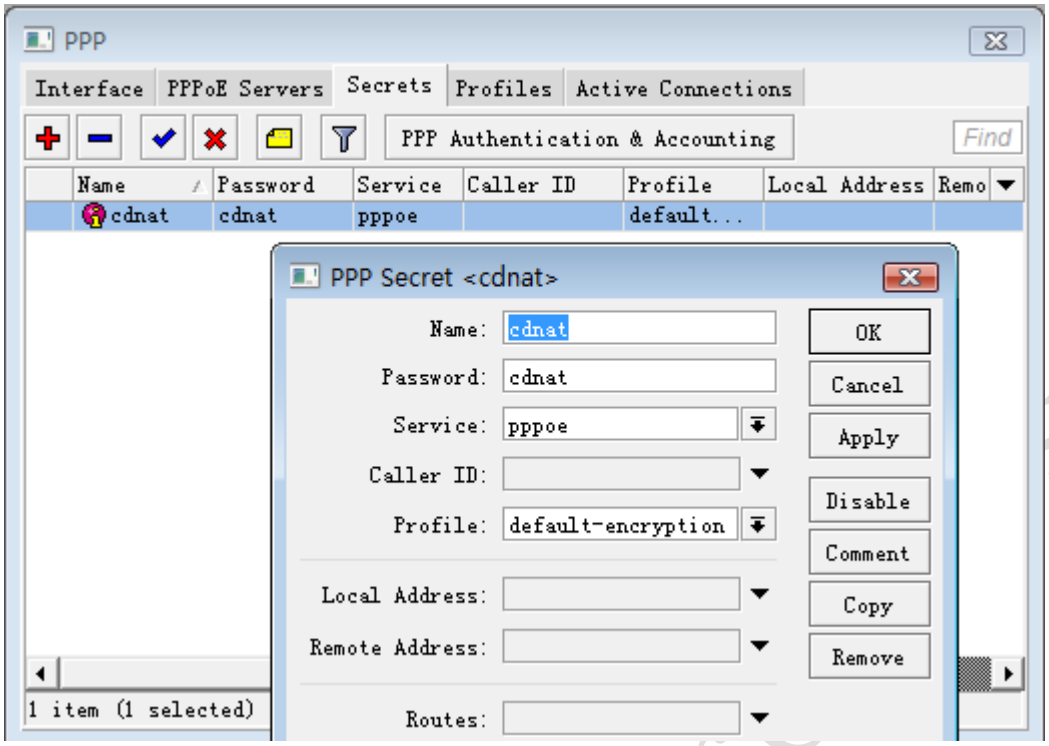
6、 Rate-Limit：用戶頻寬，rx/tx（上行/下行）

7、 Only-one：即用戶帳號是否唯一

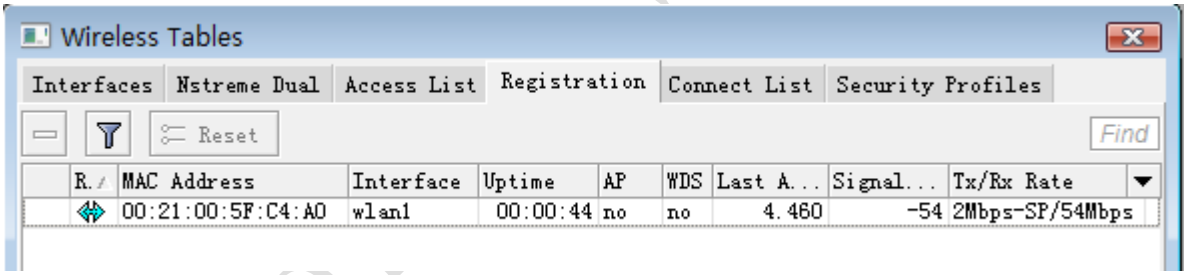
我們在配置 profile 時，設置 Local-address=192.168.10.1，Remote-address=pool1，DNS-Server=61.139.2.69，Idle-Timeout=20 分鐘，Rate-limit=1m/2m，only-one=yes



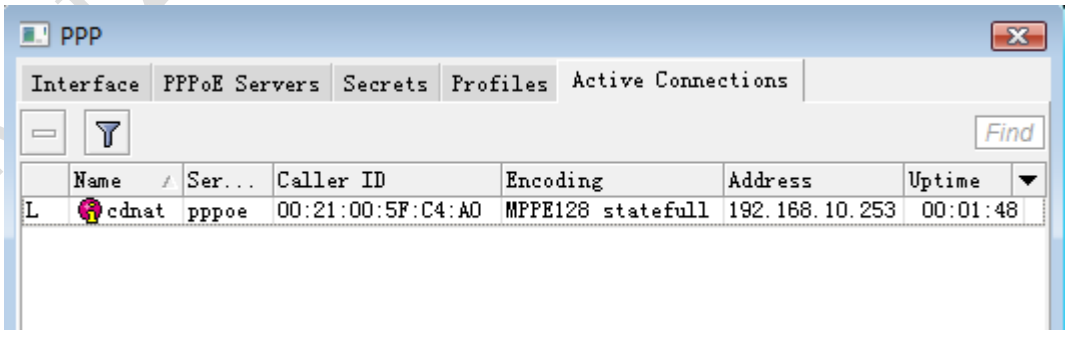
進入 secrets 配置帳號，我們新添加一個帳號為 cdnat，密碼為 cdnat。選擇服務類型為 pppoe，並選擇 Profile 類型為 default-encryption，如下圖：



步驟 4：這樣 PPPoE 伺服器建立完成，我們可以通過撥號連線測試一下，首先我們通過筆記本的無線網路卡連接到 MikroTik 的 AP 設備上，連接後，我們可以在 wireless 中的 Registration 中查看連接情況和信號強度：



在 windows 上配置完 PPPoE 撥號連線後，我們通過撥號連線到 PPPoE 服務，我們可以在 PPP 中的 Active 中看到連接成功的 PPPoE 帳號：



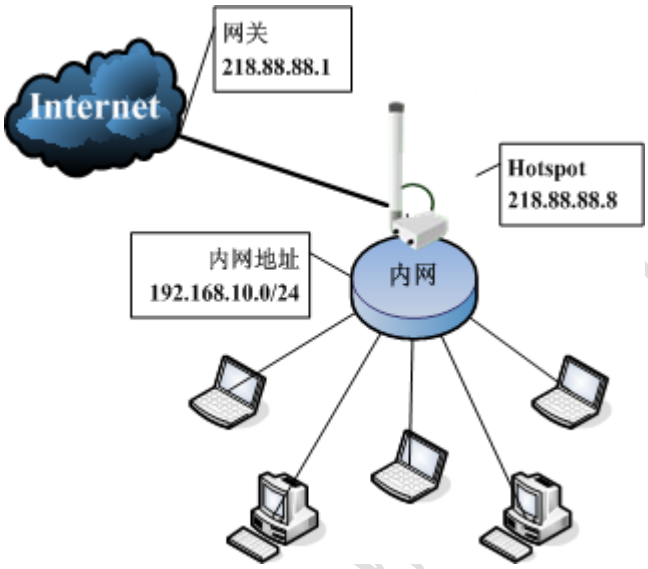
11.2 基於 Hotspot 的 WLAN 認證

Hotspot 的 WLAN 認證和 PPPoE 不同在於，PPPoE 是基於二層鏈路的認證，只能在二層網路傳輸如果有三層設備（路由器）就不無法穿透。而 Hotspot 是基於三層的 IP 網路驗證，能被二層資料透傳，Hotspot

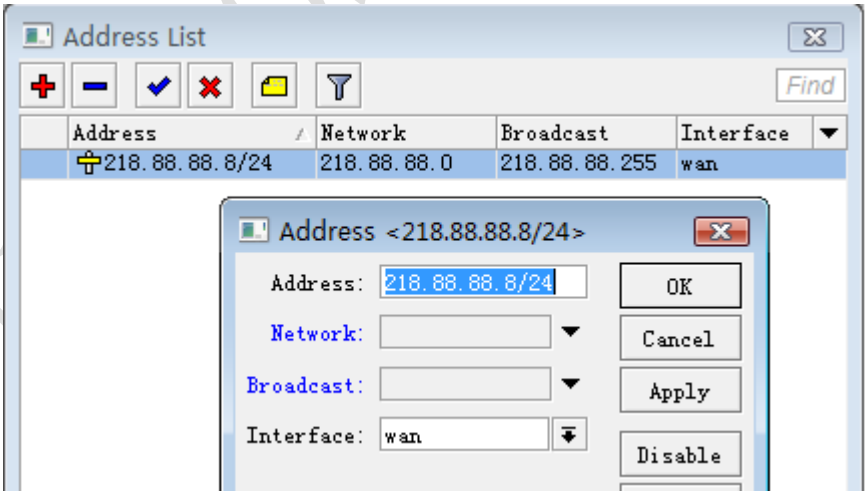
支援二層和三層的 IP 位址認證，所以能在二層和三層中傳輸，而 Hotspot 能基於 MAC 的 UPNP 認證，這種認證方式只能在二層中傳輸。我們來看看 Hotspot 的兩種驗證方式：

基於 IP 的驗證：根據用戶端配置或者獲取正確的 IP 位址和閘道，對客戶的帳號和密碼進行認證上網。

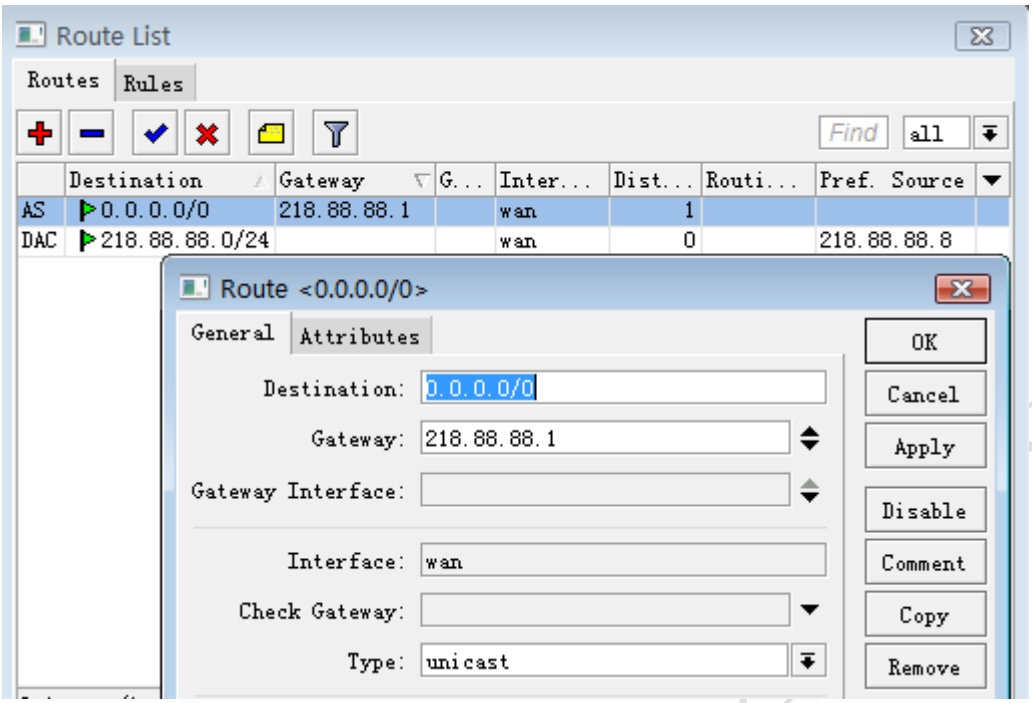
基於 MAC 的 UPNP 驗證：UPNP 隨插即用連接，Hotspot 伺服器會在同一局域網內發送 ARP 廣播，告訴局域網內的所有主機自己的閘道設備，並為線上的主機分配一個虛擬的 IP 位址，這樣客戶主機在沒有配置正確的 IP 位址情況下也能連接到 Hotspot 閘道伺服器，並認證上網。



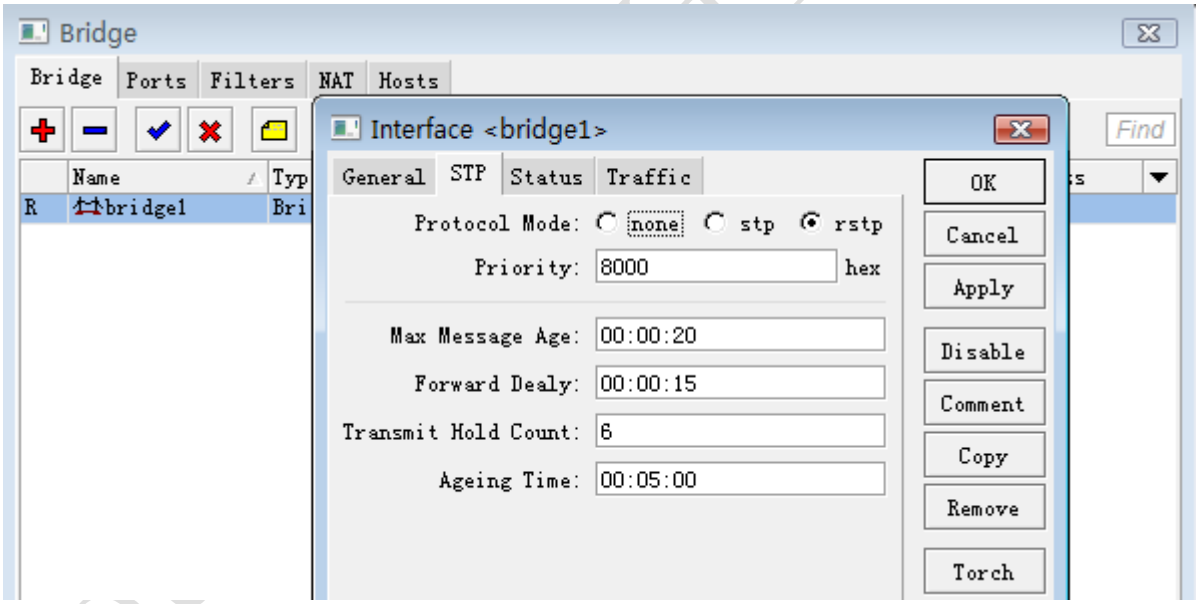
wan 口的外網連接 IP 位址是 218.88.88.8/24，內網 IP 段為 192.168.10.0/24，我們先添加 wan 口的外網 IP 地址：



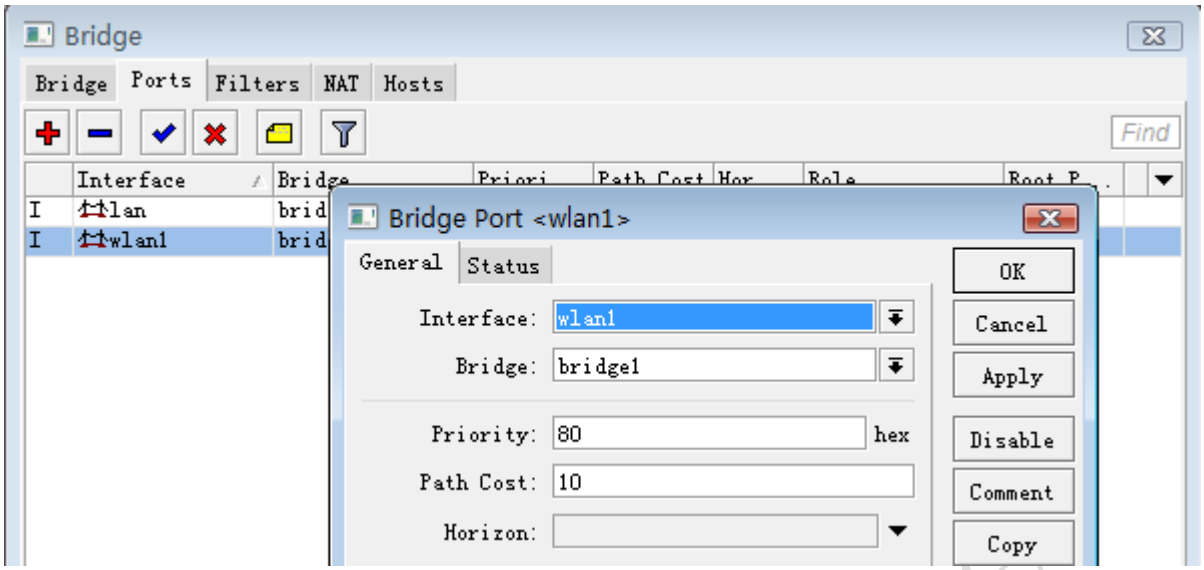
接著我們進入 ip route 配置路由，閘道為 218.88.88.1



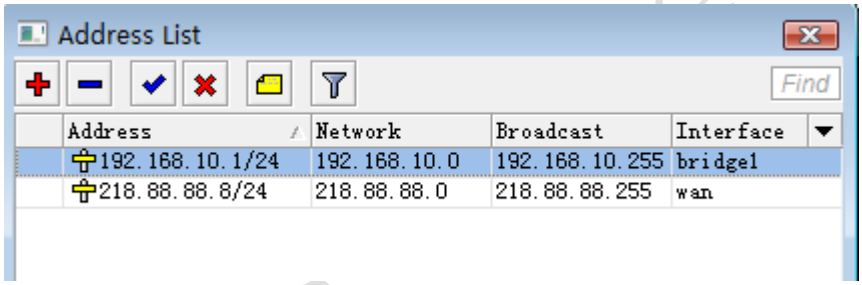
進入 bridge 在我們添加橋接，並設置 rstp 的參數，將 lan 口和 wlan1 設置到 bridge1 中：



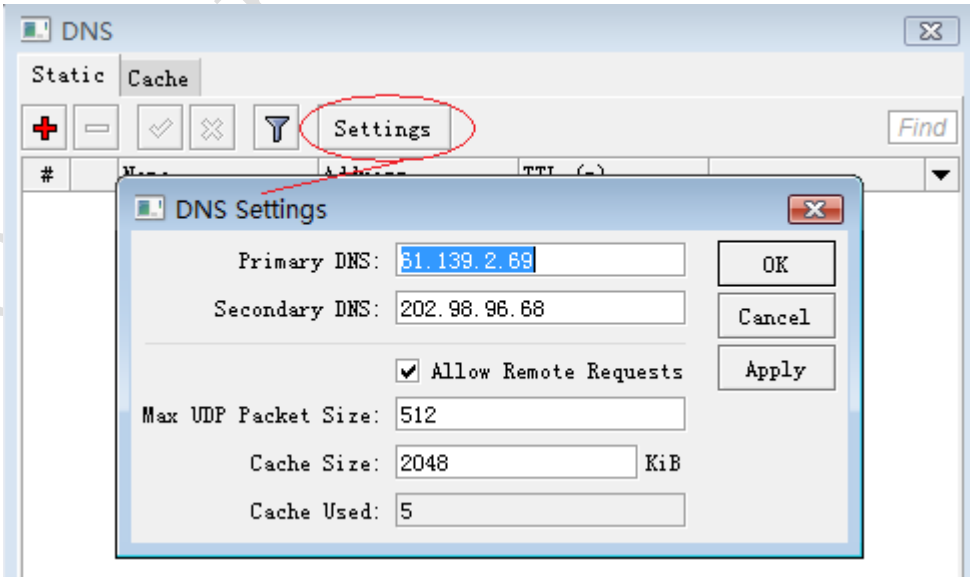
將 lan 和 wlan1 添加如 bridge1 中：



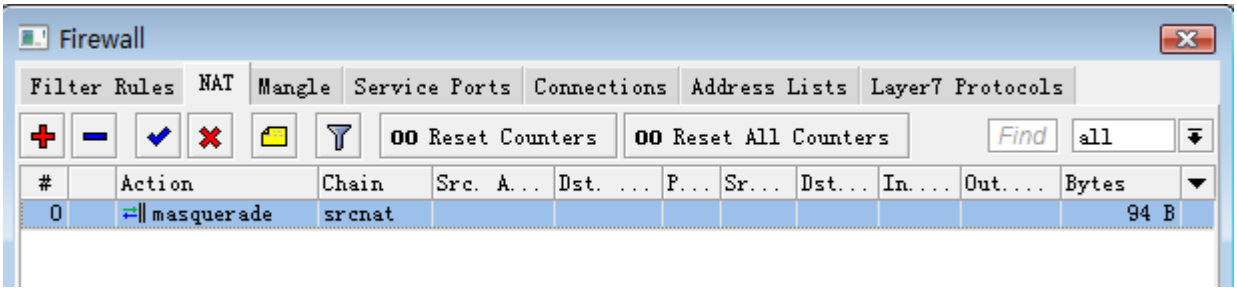
我們分配內網 IP 地址 192.168.10.1/24 設置到 bridge1 上



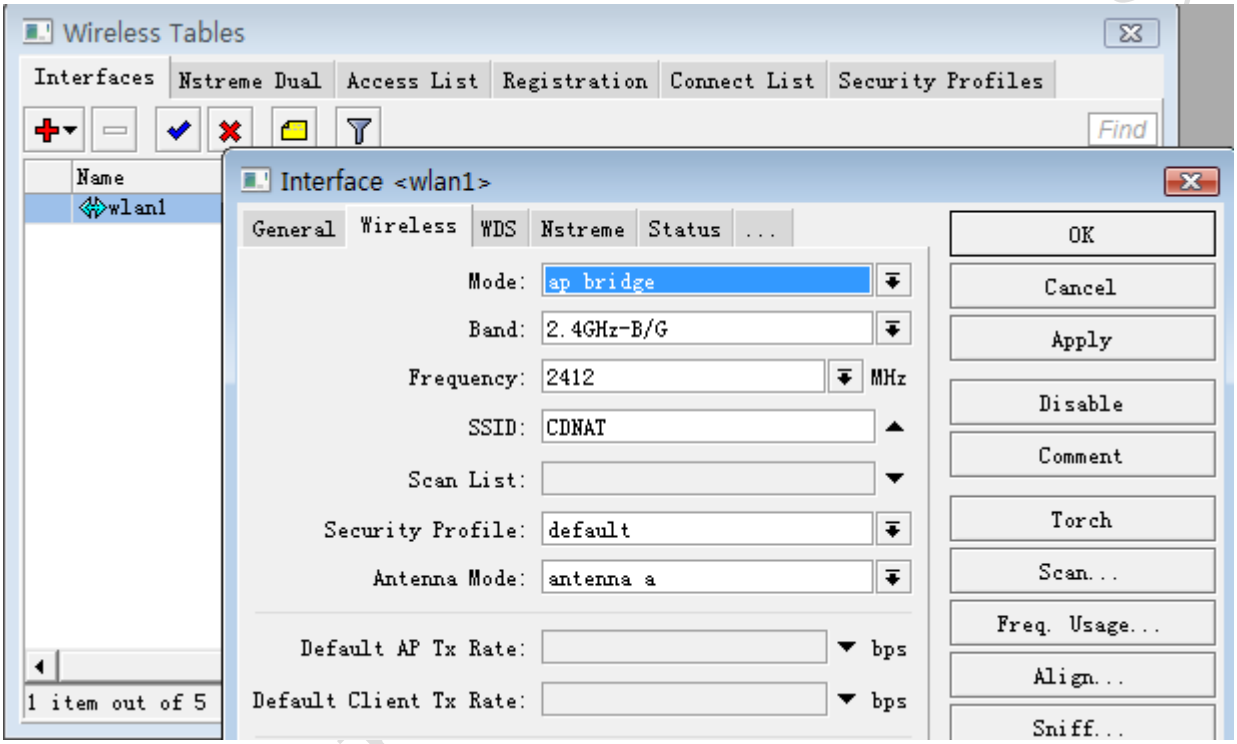
進入 ip dns 配置 DNS 服務，因為 Hotspot 要求通過本機的 DNS 解析後才能跳轉到認證頁面，這裡我們配置 DNS 服務分別為 61.139.2.69 和 202.98.96.98，將 allow-remote-request 參數選擇上，該功能是啟用 DNS 緩存：



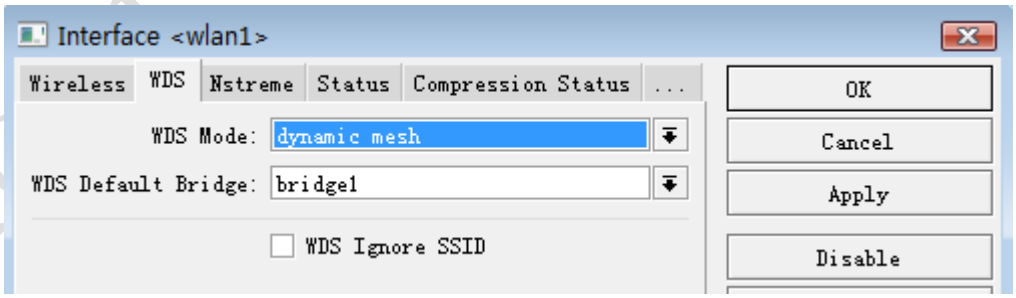
之後我們進入 ip firewall nat 配置 src-nat 的偽裝策略 action=masquerade，用於隱藏內部的私有 IP 位址連接上網，如下圖：



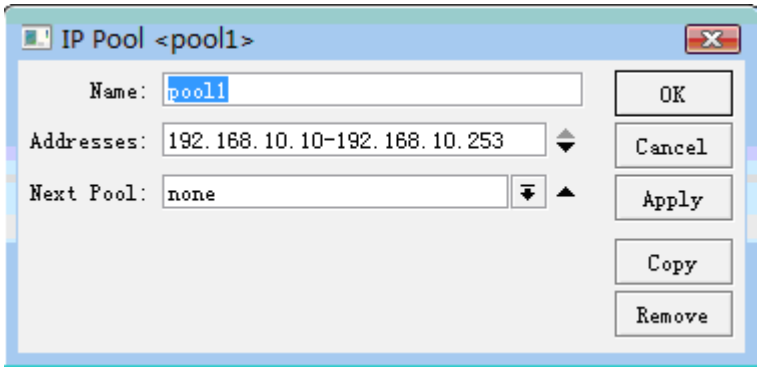
接下來配置 wlan1 的無線模組，我們這裡以 2.4G-bg 為主，配置 mode=ap-bridge、Band=2.4G-B/G、Frequency=2412、SSID=CDNAT、WDS-Mode=dynamic-mesh、wds-default-bridge=bridge1，配置如下圖：



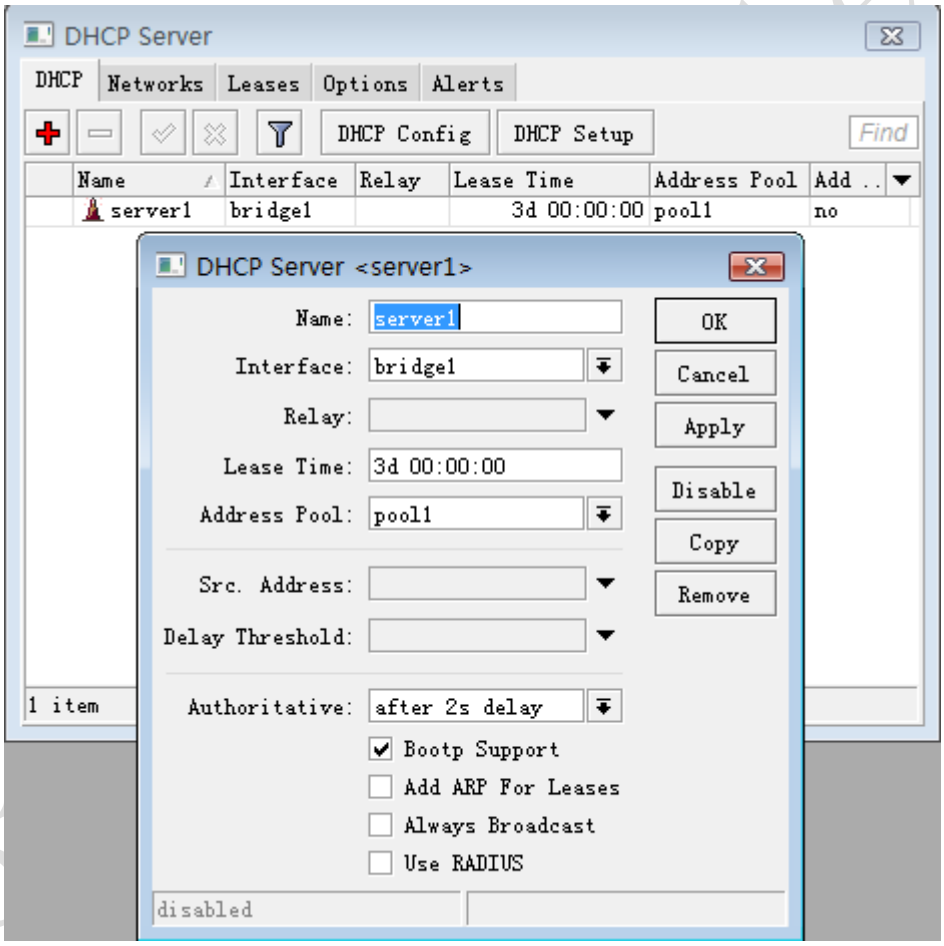
在 wlan1 配置 WDS 選項，設置 WDS-Mode=Dynamic-Mesh，比將其添加到 bridge1 中：



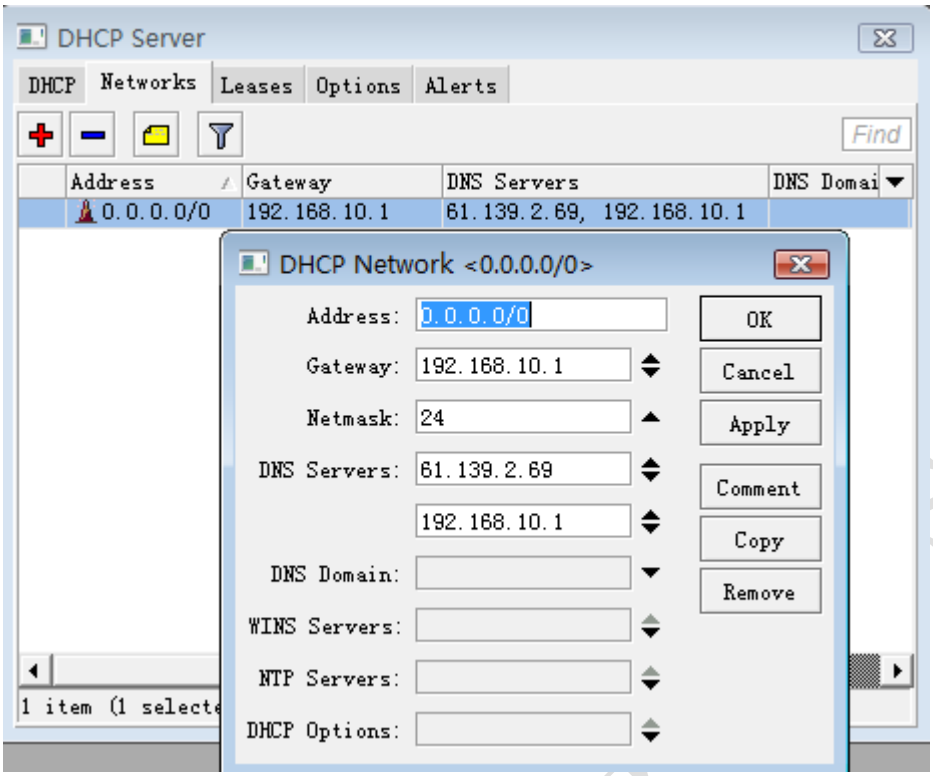
配置完基本參數後，首先我們添加位址集區用於 Hotspot 的 IP 地址分配：



我們配置好位址集區後，我們需要建立 DHCP 伺服器，當用戶沒有設置 IP 位址的情況下，可以通過 Hotspot 伺服器獲取到動態的 IP 位址，我們進入 ip dhcp-server，選擇 Interface=bridge1，給用戶分配的位址集區 Address-pool=pool1



之後進入 DHCP-Server 的 Networks 配置分配給客戶的閘道和 DNS 服務，分配閘道位址是 192.168.10.1，DNS 服務設置為 61.139.2.69 和 192.168.10.1（在 ip dns 中啟用了 dns 緩存功能）：

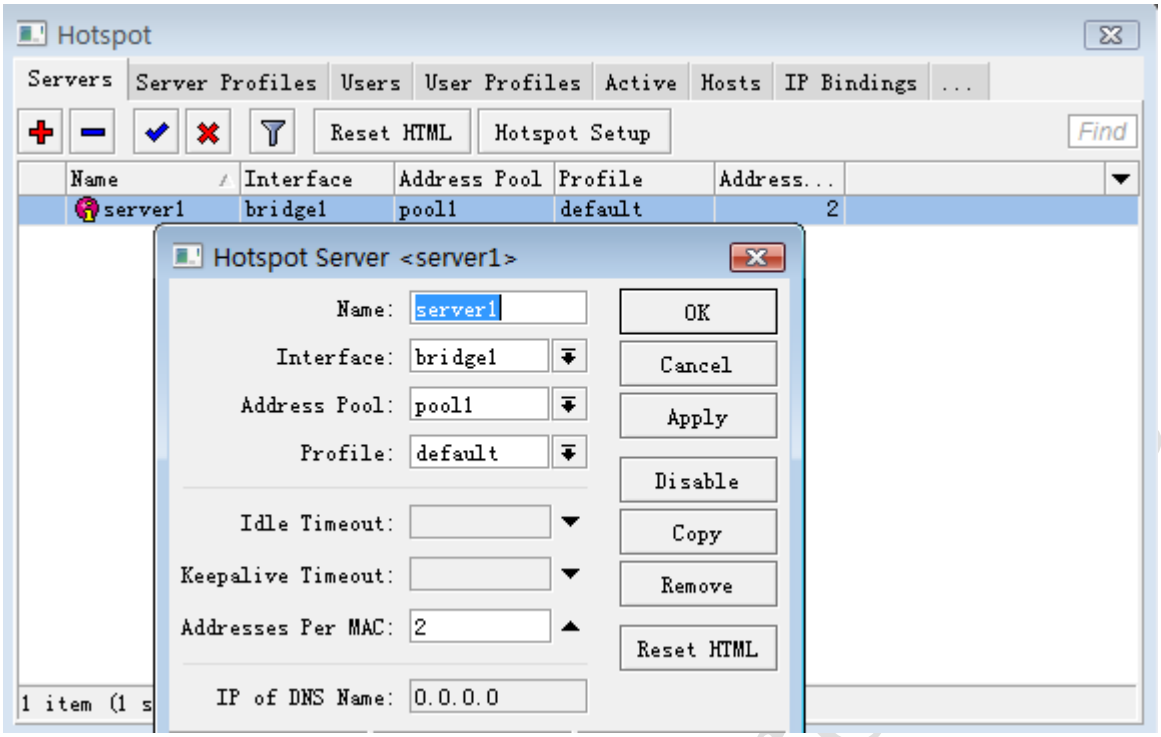


配置完 DHCP 伺服器後，我們進入 ip hotspot 目錄下配置 Hotspot 服務，配置 Hotspot 伺服器步驟如下：

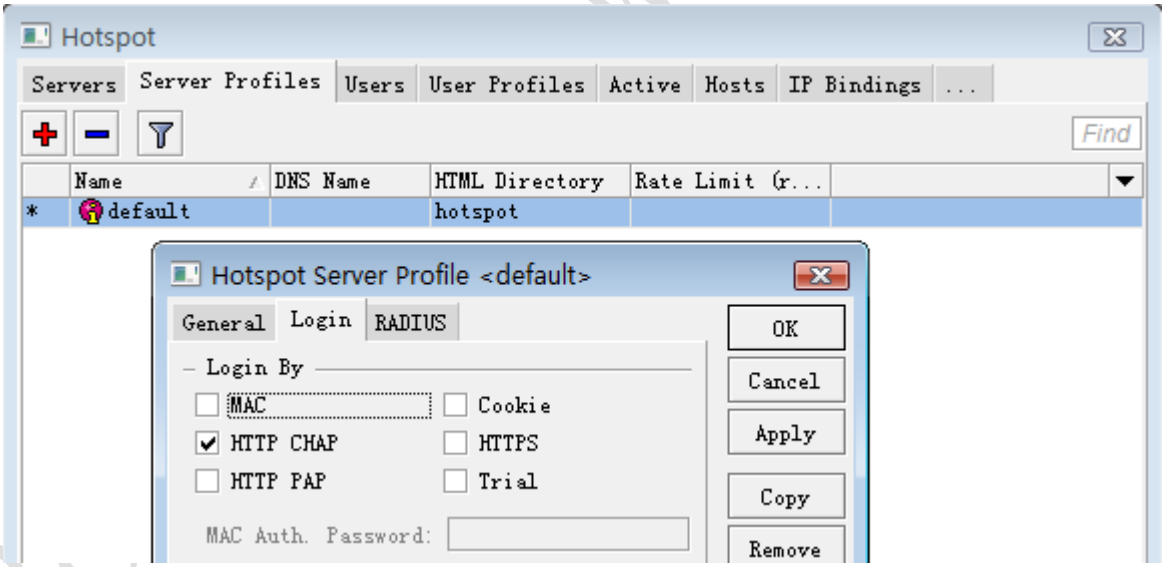
1、 進入 Server 添加 Hotspot 伺服器，在 Server-Profile 中配置伺服器組規則；
2、 進入 User-Profile 目錄下配置使用者組規則，包括共用使用者數、閒置時間和頻寬；
3、 在 User 目錄下添加使用者帳號和密碼；
4、 通過 IE 流覽器登陸 Hotspot 認證頁面，並認證登陸。

步驟 1：配置 Hotspot 伺服器

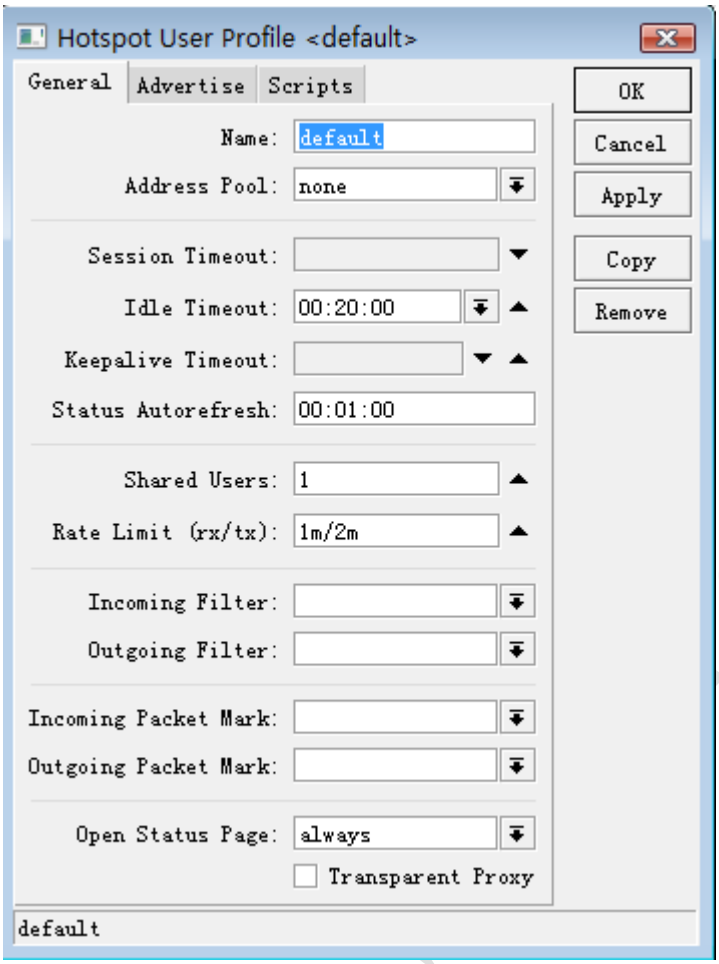
我們進入 ip hotspot server 目錄下，添加伺服器 Server1，將認證介面定義到 bridge1 上，並配置 Pool 位址集區用於 UPNP 使用者的 IP 位址分配，設置 Address-per-mac=2（允許 1 個 IP 位址邦定 MAC 位址的數量，降低 IP 位址模擬多個 MAC 攻擊的可能性）。



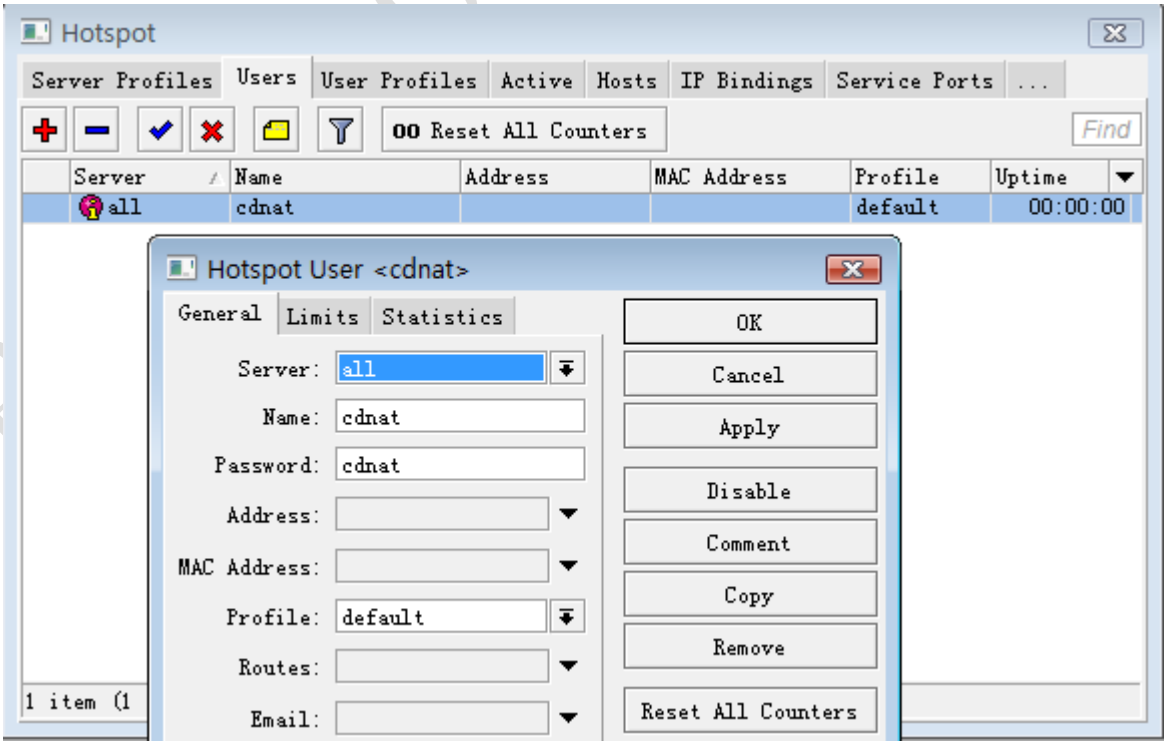
步驟 2：進入 server profiles 配置，伺服器組規則，我們選用預設的 default 規則，我們將 Login 方式修改為僅有 HTTP-CHAP 協議：



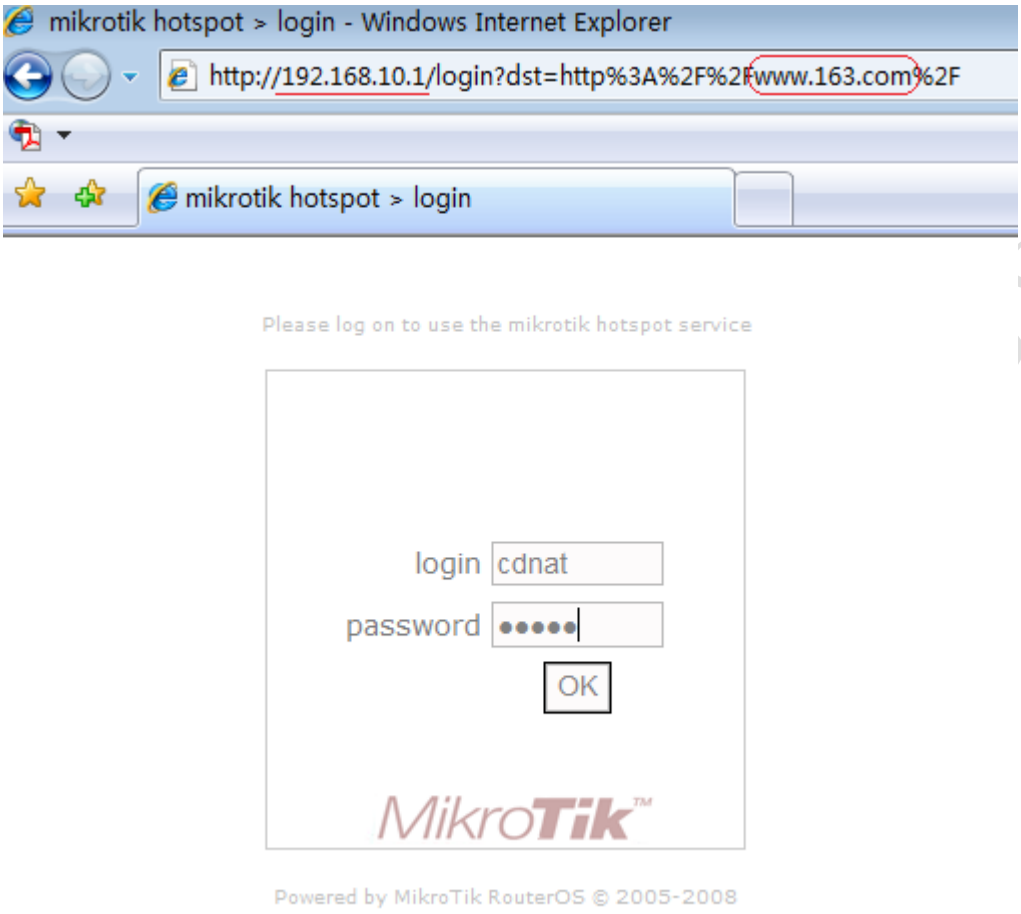
然後進入 User-profile 目錄下，配置使用者組規則，設置 Idle-Timeout=00:20:00（20 分鐘），Shared-User=1（帳號只被 1 個用戶使用），Rate-limit=1m/2m（上行 1Mbps，下行 2Mbps），配置如下：



步驟 3：進入 Users 目錄下，添加使用者帳號，帳號 cdnat，密碼 cdnat，使用者分組的 Profile 選擇 default：

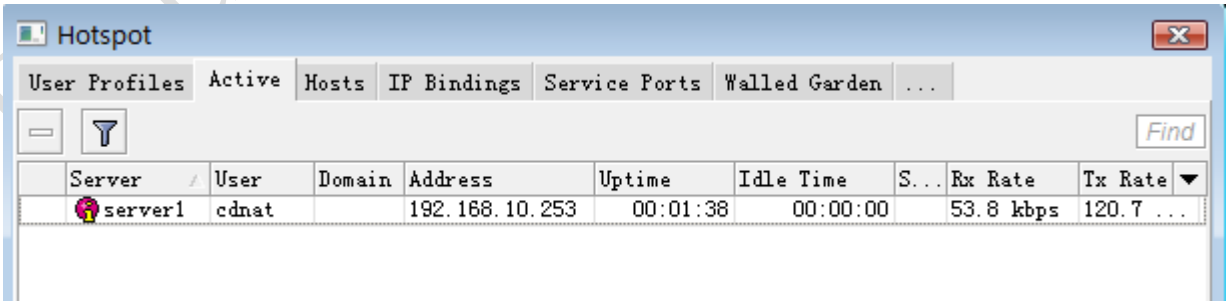


步驟 4：Hotspot 配置基本完成，現在我們可以通過 IE 流覽器登陸到 Hotspot 的認證頁面，我們在 IE 流覽器中輸入 www.163.com 的網址，當 Hotspot 伺服器正確解析道 163 網址的位址，會自動跳轉到認證頁面，輸入我們的用戶名和密碼認證登陸：



上面的位址欄中，我們可以看到，在輸入 www.163.com 被 hotspot 強行跳轉到了 192.168.10.1 的認證閘道上。該認證頁面上可以修改，具體操作請參閱**《RouterOS 中文網路教程》**。

登陸後，我們可以在 active 中看到，當前登陸的使用者帳號狀態：



注：在隨插即用情況下登陸，是在使用者沒有正確配置當前網路下的靜態 IP 位址和閘道，通過 Hotspot 伺服器的 Pool（位址集區）分配一個虛擬的 IP 位址給客戶主機，但要求客戶注意必須配置靜態的 IP 和閘道，如果只配置了靜態的 IP 位址，則隨插即用不會生效。

第十二章 其他無線應用

12.1 無線管理 VLAN 與業務 VLAN

RouterOS 的橋可以透傳二層的 VLAN 資料，通過在接入端設備配置 VLAN 介面，分別連接到各個接入點的 VLAN 交換機，這樣讓每個區域的使用者的資料能夠通過 VLAN，直接到達我們的接入設備，而不會受到其他二層資料廣播的影響，同時也減小了二層的廣播風暴，隔離每個區域的資料。在一些運營商會將管理 VLAN 和業務 VLAN 劃分開，這樣有助於對設備的管理。

關於 RouterOS VLAN 的介紹可以參考 RouterOS 網路教程中的 VLAN 一章，由於大多 RouterBOARD 都基本 Switch 交換功能，但 WLAN 網卡晶片是獨立於 Switch 交換邏輯晶片，所以只能通過 bridge 來實現 VLAN 透傳

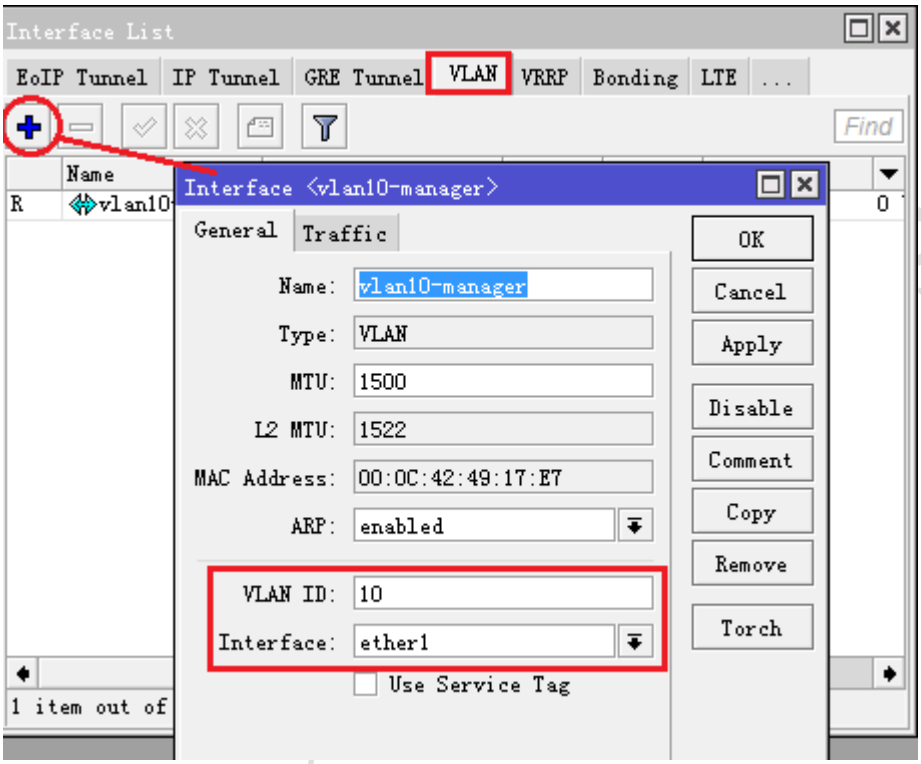
WLAN 網路可以建立 ap-bridge to station-wds 模式的橋接器連接，基於透明橋接的方式，我們可以把網路資料透傳到遠端的節點，ap-bridge 可以建立點對點或者點對多點的模式，根據網路方案的需要可以靈活調整網路結構。

VLAN 配置實例

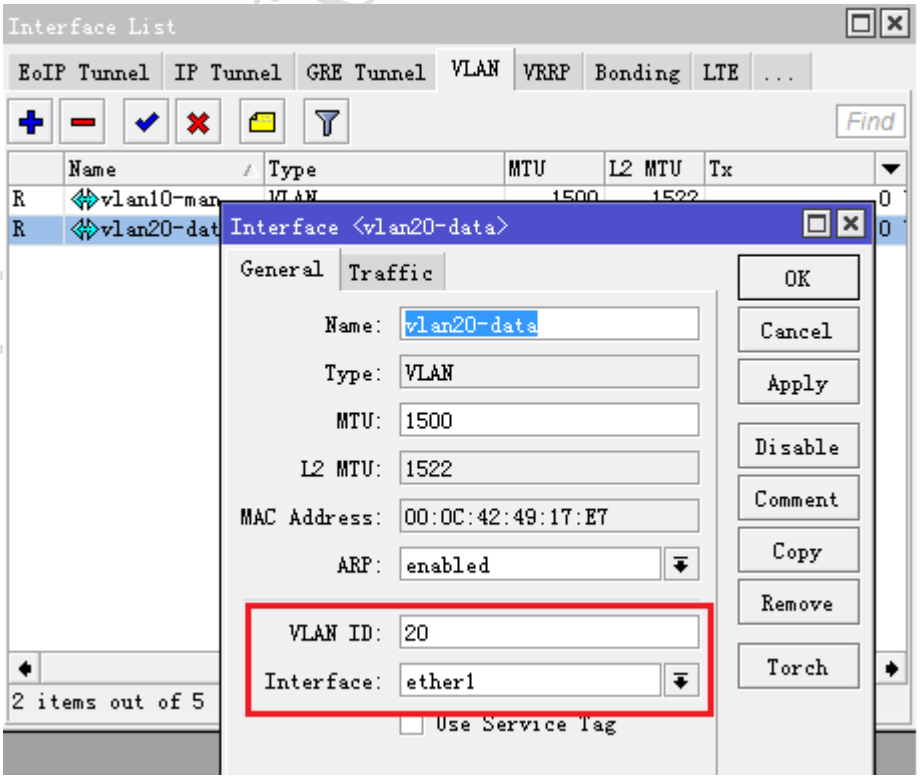
這種方式常見於運營商的 WLAN 網路，即組網時，就將管理網路和業務網路劃分開，避免之間相互影響。假設我們有以下網路，vlan 10 為管理 VLAN 負責設備管理，vlan 20 為業務 VLAN 負責用戶 PPPoE 或熱點認證上網。而上聯的三層交換機通過 trunk 模式將兩個 vlan 透傳給 RouterBOARD 設備。如下圖：

RouterBOARD 通過 ether1 的以太網介面連接到三層交換機，這樣 ether1 介面需要配置兩個 vlan，vlan10 和 vlan20

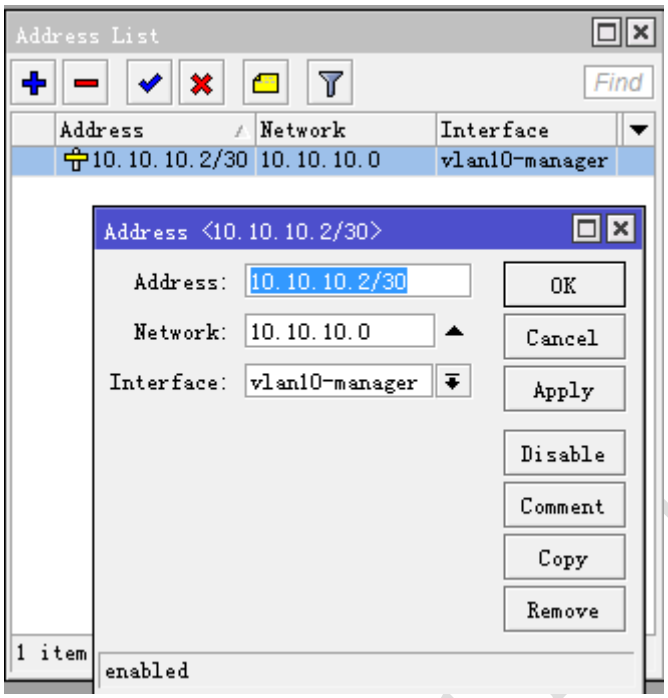
首先進入 interface 功能表，打開 VLAN 設置，添加管理 vlan，取名為 vlan10-manager，設置 VLAN ID 為 10，綁定到 ether1 介面



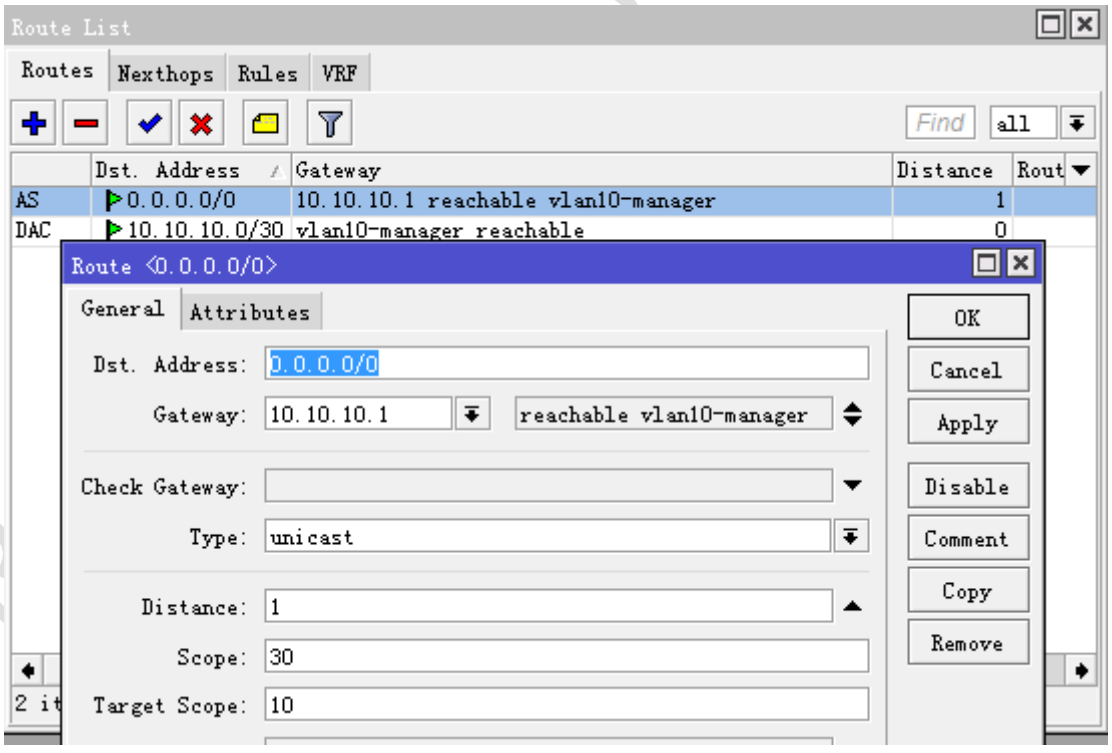
用同樣的方法添加業務 VLAN，取名 vlan20-data，設置 VLAN ID 為 20，同樣綁定到 ether1



這樣的配置類似于其他路由器配置 VLAN 子介面，下面配置管理 VLAN 的 IP 位址，進入 ip address 添加 IP 位址，並配置到 vlan10-manager 上

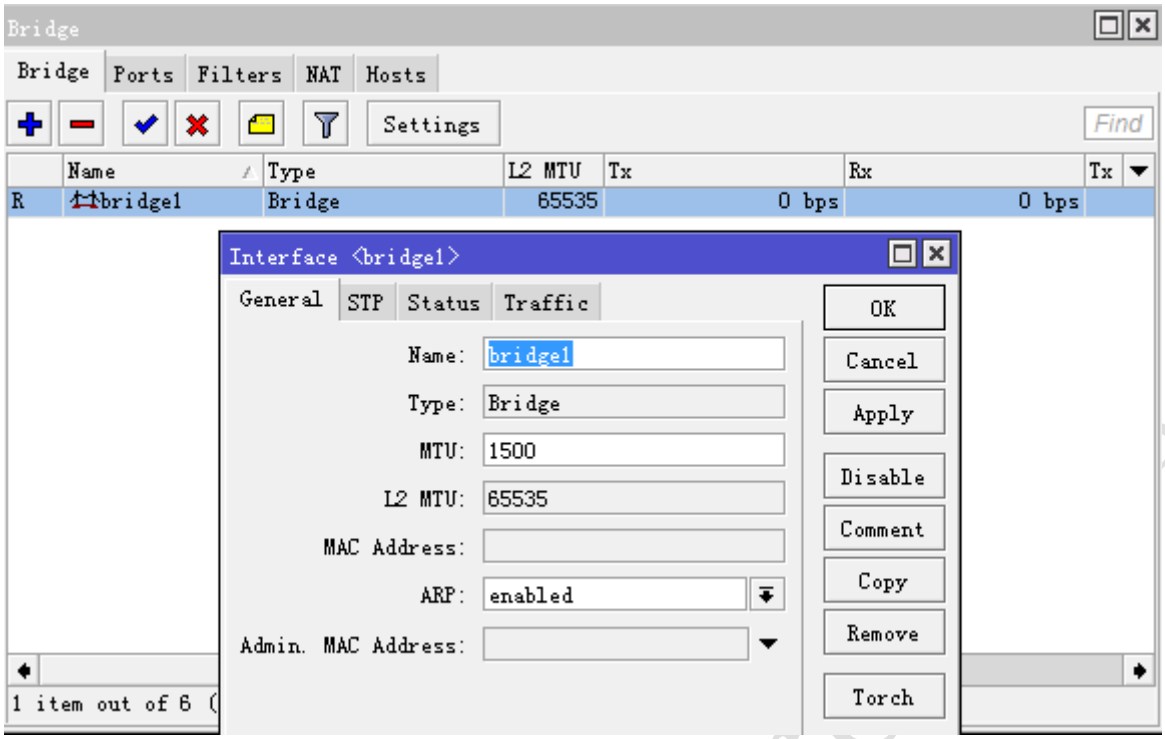


添加預設閘道器 10.10.10.1

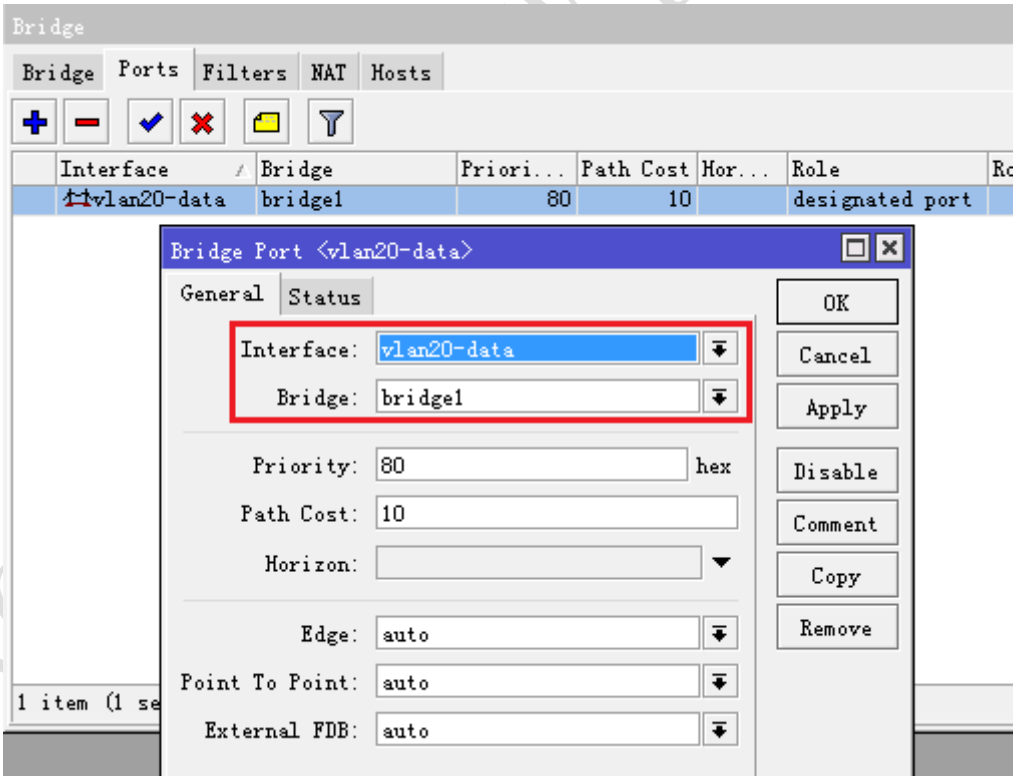


這樣管理 VLAN 配置完成，接下來配置業務介面的無線網路透傳，

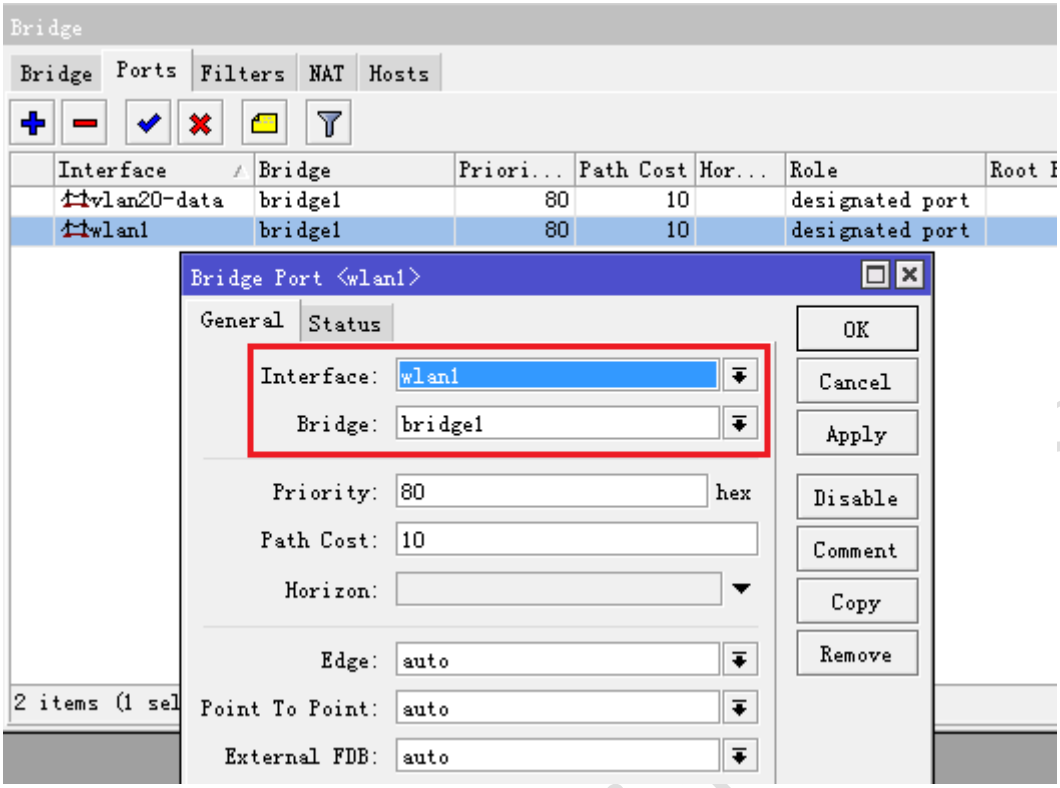
業務 VLAN 實質就是要將 vlan 20 和 wlan1 網卡做透明橋接，這裡我們需要添加一個 bridge，進入 bridge 功能表，添加一個 bridge1



進入 ports 將 vlan20-data 和 wlan1 網卡做橋接，下面是添加 vlan20-data 到 bridge1



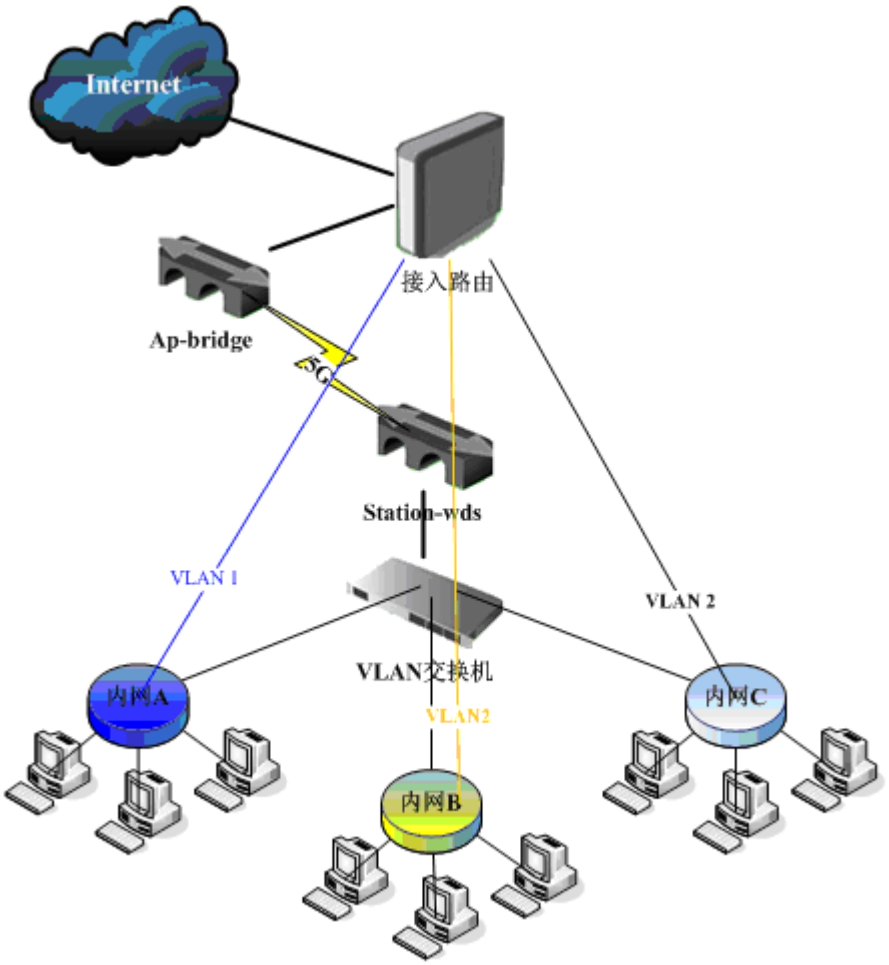
添加 wlan1 到 bridge1



這樣配置，就完成了 vlan 10 為管理 VLAN，vlan 20 為業務 VLAN 的配置

VLAN 方案一

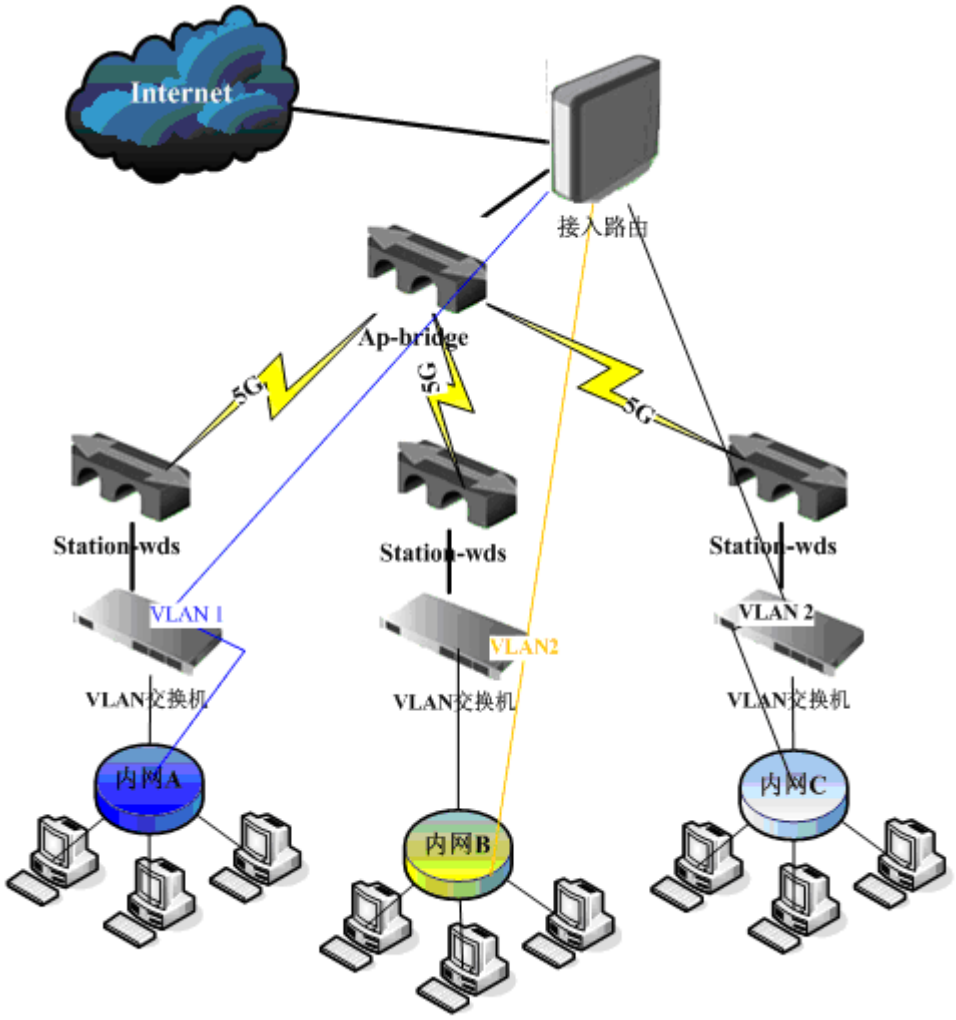
假設我們的網路通過 Internet 連接到接入路由，然後分別通過 2 個點對點的橋接設備連接到遠端的 3 個網路，在點對點和 3 個網路中間有一個 VLAN 交換機連接，我們通過接入路由配置 VLAN，分別連接到遠端的 3 個不同 VLAN 網路，如下圖：



通過這樣的網路結構，我們可以為每一個 VLAN 劃分一個子網，便於網路的管理。由於建立的是 VLAN 隧道，3 個網路可以直接（VLAN 隧道建立了虛擬連結，通過 ID 區分每個鏈路）和接入路由通訊，這樣的 VLAN 劃分我們不僅可以劃分不同的子網，也可以建立 3 個獨立的 PPPoE 服務或者 Hotspot 服務。

VLAN 方案二

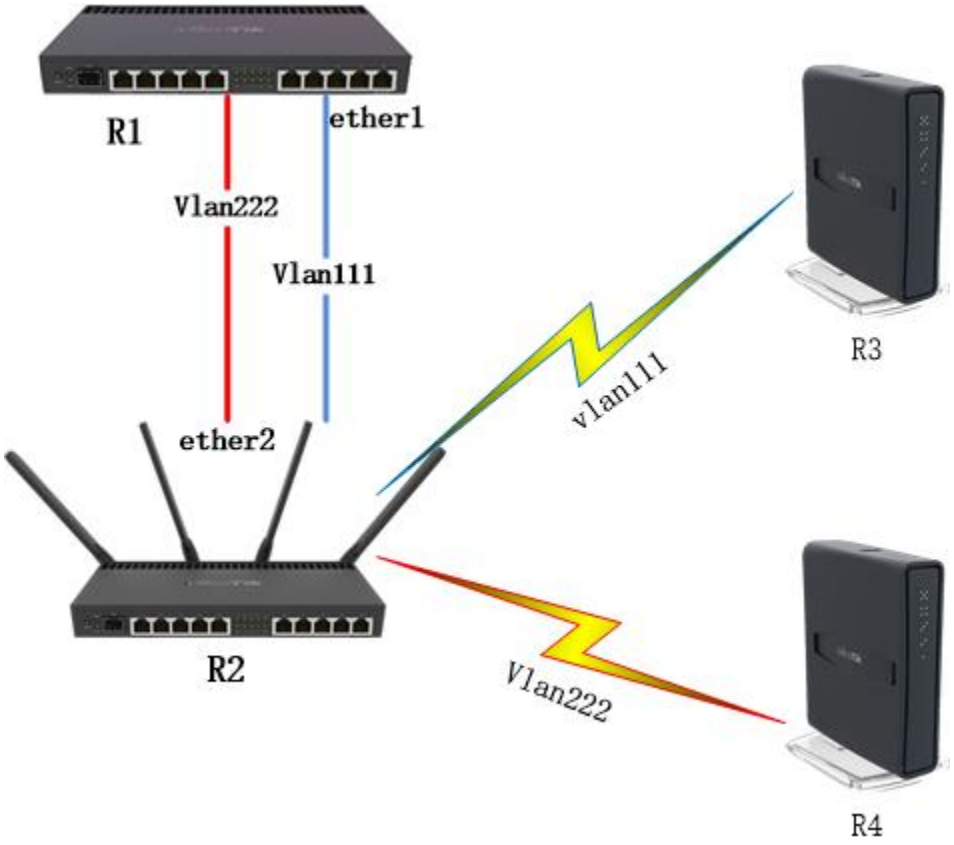
假設我們的無線網路不是點對點的方式，而是點對多點的橋連接，構建與其點對點模式基本相同，同樣是通過接入路由建立多 VLAN，分別連接到各網路內的 VLAN 交換機：



以上是對 VLAN 在 WLAN 網路中的一些應用參考。

12.2 基於 vlan-filtering 和 wireless vlan-mode（推薦）

RouterOS 支援基於無線鏈路的 VLAN 透傳。 例如，當我們在一個辦公網路中，需要使用一台 RouterOS 設備分隔辦公 WiFi 和來訪 WiFi 時，可以使用 VLAN 方式做二層隔離，使用 Wlan1 做來訪 WiFi，在 Wlan1 上創建虛擬 AP（VirtualAP Wlan2）作為辦公 WiFi。 在之前的版本中有相應的解決方案，但 v6.41 後在 bridge 增加了 VLAN Filtering 功能，通過 vlan filtering 能更好實現無線中 vlan 透傳，也是官方推薦方法。



從 RouterOS v6.41 開始，Birdge 增加了 VLAN Filtering 功能支援二層 VLAN 轉發和 VLAN 標記處理，因此在 v6.41 後建議使用以下配置方案。

R1 路由器:

在路由器 R1 創建指定的 VLAN 介面在對應的 ether1 有線介面上，即 trunk 模式的配置，並配置 IP 位址到 VLAN 介面，啟用三層 VLAN。

```
[admin@R1] >
/interface vlan
add interface=ether1 name=vlan111 vlan-id=111
add interface=ether1 name=vlan222 vlan-id=222

/ip address
add address=192.168.1.1/24 interface=vlan111
add address=192.168.2.1/24 interface=vlan222
```

R2 路由器:

我們需要創建 wlan2 虛擬機器 AP 介面，並配置 Wlan1 和 Wlan2 的加密規則

```
[admin@R2] >
/interface wireless security-profiles
add authentication-types=wpa-psk,wpa2-psk eap-methods=”” management-protection=allowed
mode=dynamic-keys name=\
vlan111 supplicant-identity=”” wpa-pre-shared-key=yusvlan111 wpa2-pre-shared-key=yusvlan111
```

```
add authentication-types=wpa-psk,wpa2-psk eap-methods="" management-protection=allowed
mode=dynamic-keys name=\
      vlan222 supplicant-identity="" wpa-pre-shared-key=yusvlan222
wpa2-pre-shared-key=yusvlan222
```

在路由器 R2 的 Wlan1 創建一個虛擬 AP 介面取名 Wlan2，並設置 Wlan1 和 Wlan2 不同的無線加密規則，設置 vlan-mode=use-tag

```
/interface wireless
set [ find default-name=wlan1 ] disabled=no mode=ap-bridge security-profile=vlan111 ssid=vlan111
vlan-id=111 vlan-mode=use-tag
add disabled=no master-interface=wlan1 name=wlan2 security-profile=vlan222 ssid=vlan222
vlan-id=222 vlan-mode=use-tag
```

路由器 R2 創建橋接，設置 vlan-filtering=yes，添加相應的橋接端口，並在 bridge vlan 中創建對應端口的 vlan id 標籤

```
[admin@R2] >
/interface bridge
add fast-forward=no name=bridge1 vlan-filtering=yes

/interface bridge port
add bridge=bridge1 interface=ether2
add bridge=bridge1 interface=wlan1
add bridge=bridge1 interface=wlan2
/interface bridge vlan
add bridge=bridge1 tagged=ether2,wlan1 vlan-ids=111
add bridge=bridge1 tagged=ether2,wlan2 vlan-ids=222
```

提示： 一些 RB 和 CRS 設備集成了交換晶片，能實現以太網二層資料交由交換晶片做線速轉發，啟用橋接 VLAN filtering 功能後無法實現交換晶片線速轉發（除 CRS3xx 系列設備），則會交又 CPU 處理。由於無線介面沒有歸屬於交換晶片，因此也無法實現硬體轉發，只能交給 CPU 處理

R3 路由器:

路由器 R3，添加對應 VLAN111 的 IP 位址到 Wlan1 介面，並創建與 VLAN111 相同的加密配置，設置無線網卡 wlan1 的 mode=station

```
[admin@R3] >
/interface wireless security-profiles
add authentication-types=wpa-psk,wpa2-psk eap-methods="" management-protection=allowed
mode=dynamic-keys name=\
vlan111 supplicant-identity="" wpa-pre-shared-key=yusvlan111 wpa2-pre-shared-key=yusvlan111

[admin@R3] >
/ip address
add address=192.168.1.3/24 interface=wlan1

/interface wireless
set [ find default-name=wlan1 ] disabled=no mode=station security-profile=vlan111
```

R4 路由器:

路由器 R4，同樣添加對應 VLAN222 的 IP 位址到 Wlan1 介面上，並創建與 VLAN222 相同的加密配置，設置無線網卡 wlan1 的 mode=station

```
[admin@R4] >
/interface wireless security-profiles
add authentication-types=wpa-psk,wpa2-psk eap-methods=”” management-protection=allowed
mode=dynamic-keys name=\
vlan222 supplicant-identity=”” wpa-pre-shared-key=yusvlan222 wpa2-pre-shared-key=yusvlan222

[admin@R4] >
/ip address
add address=192.168.2.4/24 interface=wlan1

/interface wireless
set [ find default-name=wlan1 ] disabled=no mode=station security-profile=vlan222
```

以上配置的也適用於終端筆記本，手機和平板接入，只需要在 R1 的 vlan111 和 vlan222 創建 DHCP 服務

DHCP 服務配置

R1 路由器創建 DHCP 服務，首先創建位址集區

```
/ip pool
add name=vlan111 ranges=192.168.1.10-192.168.1.50
add name=vlan222 ranges=192.168.2.10-192.168.2.50
```

R1 上配置 DHCP 服務

```
/ip dhcp-server
add address-pool=vlan111 authoritative=after-2sec-delay disabled=no interface=vlan111
name=server1
add address-pool=vlan222 authoritative=after-2sec-delay disabled=no interface=vlan222
name=server2

/ip dhcp-server network
add address=192.168.1.0/24 dns-server=192.168.1.1 gateway=192.168.1.1 netmask=24
add address=192.168.2.0/24 dns-server=192.168.2.1 gateway=192.168.2.1 netmask=24
```

12.3 LED 配置

RouterOS 允許管理員配置每個 led（發光二極體）的活動狀態。例如通過配置 led 顯示無線信號強度，通過閃爍顯示以太網介面傳輸活動狀態，以及其他的選項等。Led 控制主要針對 RouterBOARD 設備設置。

操作路徑： /system leds

下面是 Groove 設備的 led 預設配置：

```
[admin@MikroTik] /system leds> print
Flags: X - disabled
 #    TYPE                      INTERFACE               LEDS
 0    wireless-signal-strength                          led1
                                                             led2
                                                             led3
                                                             led4
                                                             led5
 1    interface-activity        ether1                  user-led
```

RB Groove 使用 5 個 led 燈顯示無線信號強度，一個顯示以太網介面狀態

6.0 支援 LED 無線信號強度顯示的設備包括 RB400 系列、RB911/711 系列、RB SXT 和 Groove/Metal 系列，根據不同信號強度指定 LED 燈：

- 1 LED - 開啟，當無線用戶端連接到 AP 的信號強度 >=-89dBm
- 2 LED - 開啟，當信號強度 >= -82dBm
- 3 LED - 開啟，當信號強度 >= -75dBm
- 4 LED - 開啟，當信號強度 >= -68dBm
- 5 LED -開啟，當信號強度 >= -61dBm

對於 RB751 和 RB951 系列，5 個以太網介面 LED 和 1 個無線狀態 LED

配置屬性

屬性	描述
disabled (*yes* / *no*; 默認: no)	是否選擇禁用
interface (*字元*; 默認:)	介面名稱，根據網路介面類別型知道 led 顯示類型
modem-signal-treshold (整型 *[-113..-51]*; 默認:)	模組的信號強度設置
leds (led 列表; 默認:)	Led 清單中用於狀態顯示，例如：通過 5 個 led 燈顯示無線信號強度
type (*ap-cap* / *flash-access* / *interface-activity* / *interface-receive* / *interface-status* / *interface-transmit* / *modem-signal* / *wireless-signal-strength* / *wireless-status*;)	狀態類型： ap-cap - 在與 CAPsMAN 初始化狀態下 CAP 閃爍，連接完成後穩定 *flash-access* - 當訪問快閃記憶體 led 閃爍、 *interface-activity* - 介面活動 led 閃爍 *interface-receive* - 介面接收資料 led 閃爍 *interface-status* - 介面連接狀態 led 亮起 *interface-transmit* - 介面發送資料 led 閃爍 *modem-signal* - 3G 模組信號 led 閃爍 (USB 或 miniPCIe 介面) *wireless-signal-strength* - 無線信號強度變化 led 燈亮起，需要多 led 支援 *wireless-status* - 無線連接狀態 led 燈亮起

CAP 配置實例

在 RB951 上設置 user-led 燈，顯示當前 CAP 狀態

```
/system leds
add leds=user-led type=ap-cap
```

注意：6.23 開始 led 命令被刪除，會直接使用/system leds 下對 led 燈進行控制，所以你在 6.23 版本下無法找到 led 命令，需要修改你的腳本在 RB 規範的類型使用

```
*) leds - removed 'led' command and added support for 'on', 'off' types under 'system leds';
```

在 type 中增加了兩個屬性 on 和 off，例如控制 user-led

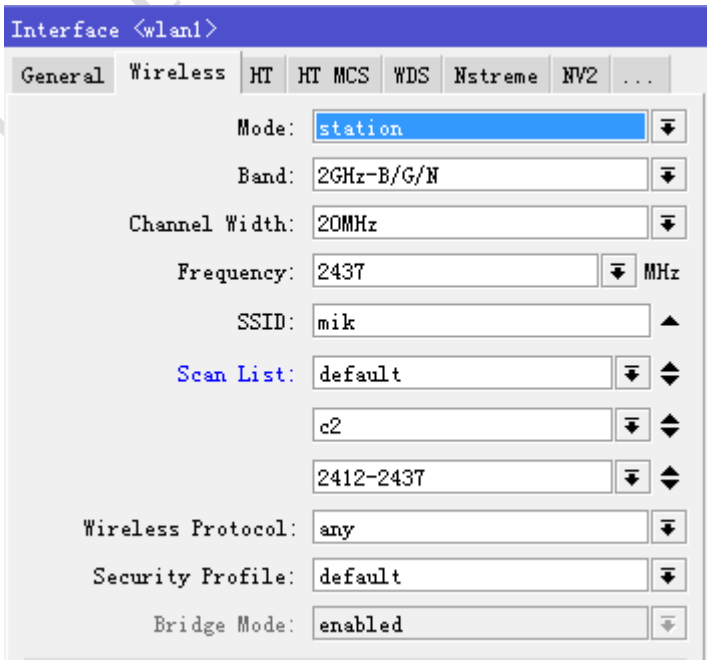
```
/system leds add leds=user-led type=off
```

12.4 Scan-list 搜索列表

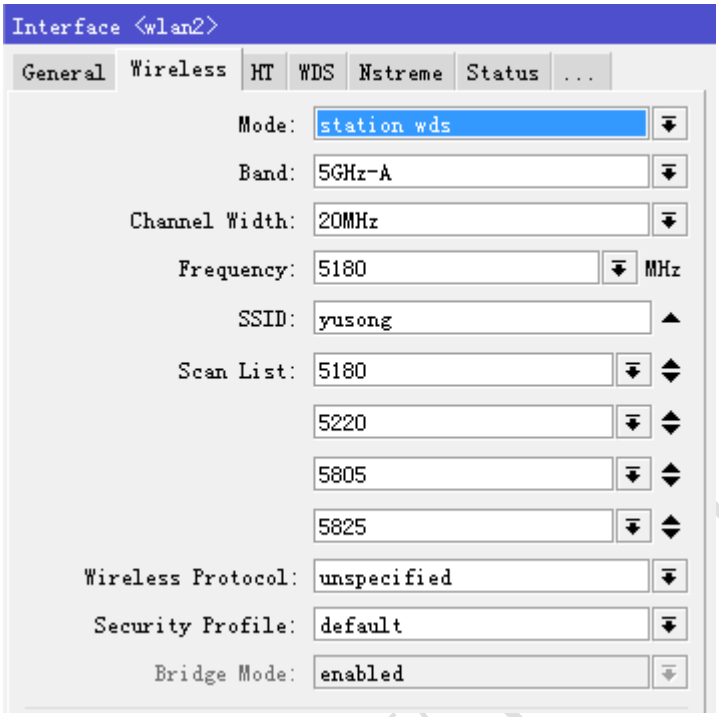
Scan-list 在之前的 Superchannel 介紹時提到，Scan-list 不僅用於指定 superchannel 的頻率範圍，還可以定義 RouterOS 用戶端設備搜索頻率範圍

Scan-list 默認在 5ghz 頻段下，每間隔 20MHz 步進搜索，在 5ghz-turbo 頻段下每間隔 40MHz，2.4G 則間隔 5MHz。如果 scan-list 採用手動設定，所有指定的頻率範圍都會被鎖定搜索（例如：scan-list=default,5200-5245,2412-2427 即會使用默認頻段搜索，並添加從 5200-5245 或 2412-2427 頻率範圍.） 當然如果是 scan-list=2412-2437，即只能在 2412-2437 頻率範圍搜索。

從 RouterOS v6.0 開始對於 winbox 或 webfig 配置 Scan-list 輸入多頻率、頻段範圍，會被分隔為多個 Scan-list 選項。使用逗號間隔幀 v6.0 後的 winbox/webfig 不再支援。如下圖的 winbox 設置：



如在 802.11a 無線網路規定了頻率連接範圍，station-wds 用戶端設備只能連接 5180、5220、5805 和 5825 四個頻率，我們在 v6.0 的網卡配置如下：



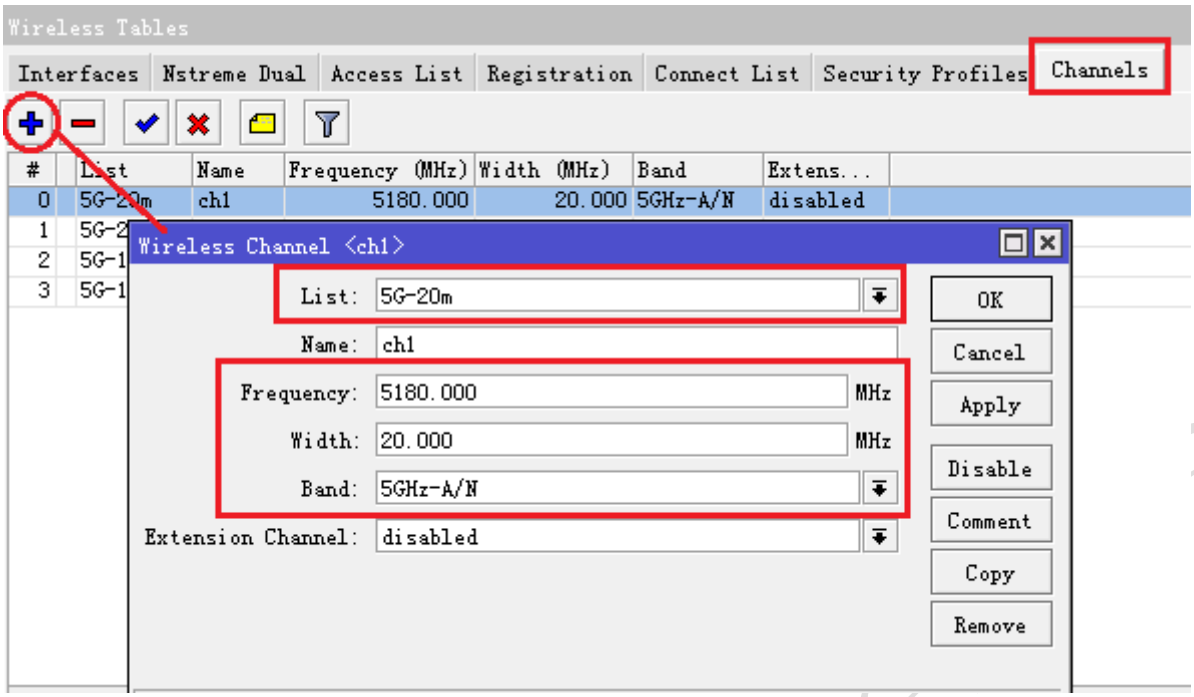
12.5 高級頻率搜索 channels

RouterOS 從 v6 開始提供了一個高級的無線頻率搜索清單配置- channels，僅支援 Atheros AR92xx 晶片，頻率範圍為 2192-2734MHz 和 4800-6100MHz，且支援中心頻率 0.5MHz 步進，能定制頻寬分為從 2.5～30MHz 每 0.5MHz 步進，無線網卡通過 Scan-list 調用該功能，同時 Scan-list 有可以定義搜索範圍。

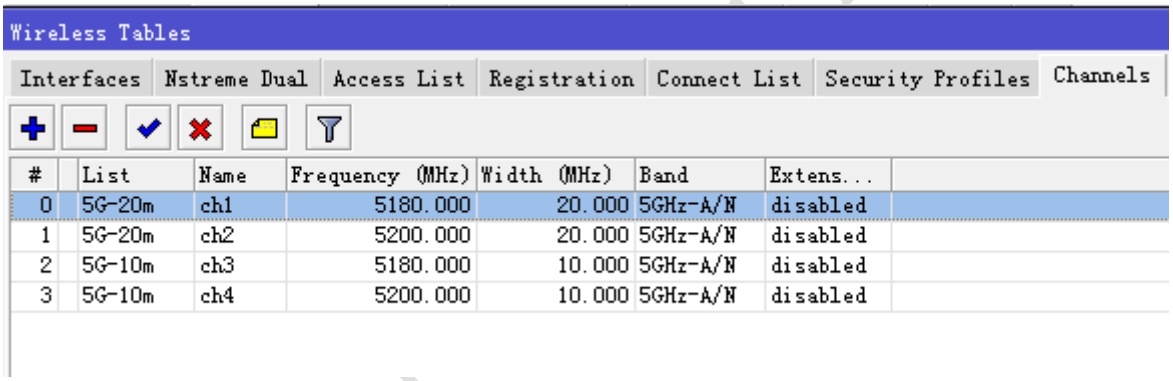
操作路徑： /interface wireless channels

對於特殊的網路環境，我們希望設備採用不通的頻率（frequency）和頻寬（width），我們可以通過 channels 來定義，並配置到 Scan-list 中

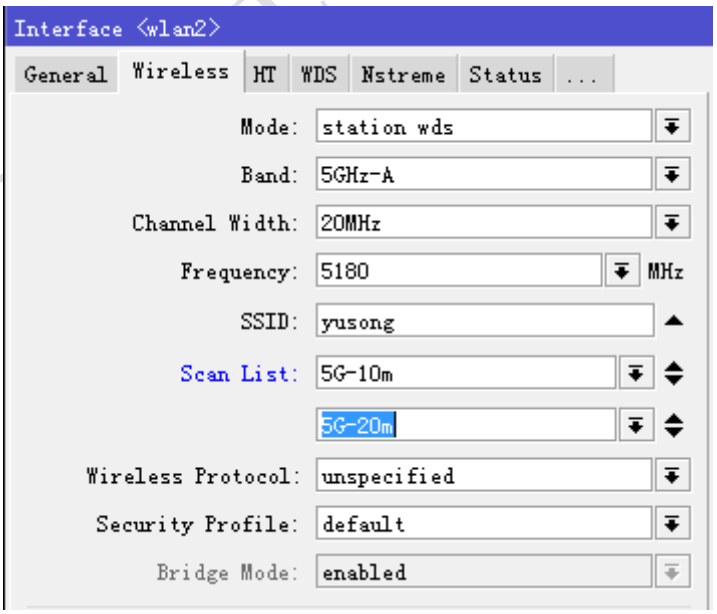
例如下面定義 802.11a 的無線網路環境的搜索範圍和頻寬，頻率 5180 和 5200 分別使用 20Mhz 和 10Mhz 頻寬



依次安装各個頻率和頻寬添加到 channels 下：



添加完成後，我們將 Channels 配置到 Scan-list 中



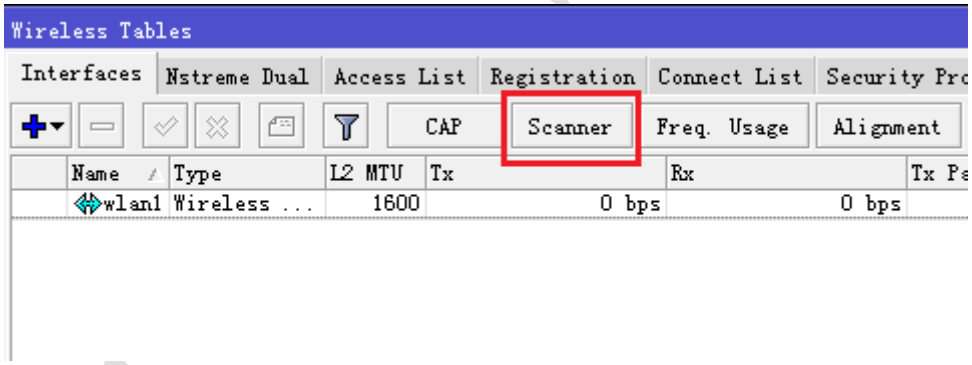
12.6 搜索器 Scaner

Scaner 工具，是用於搜索附近的 AP，搜索 AP 類型取決於無線網卡的型號和設定參數，搜索器便於你找到當前區域中的 AP 節點，並選擇連接。當無線網卡設置為 2.4G 時，只能搜索 2.4G 頻段中的 AP，同樣當設置為 5G 時，只能搜索 5G 頻段中的 AP

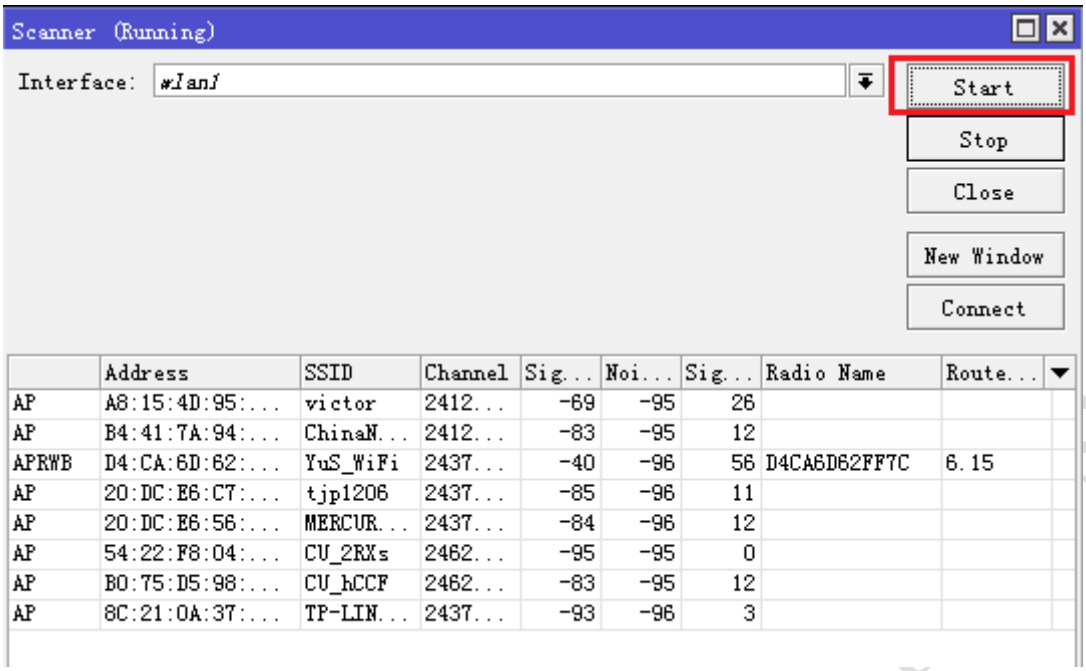
Scaner 搜索器，可以在 interface wireless 中開啟，通過 scan 命令，如下面搜索 wlan1 2.4G 網路 AP

```
[admin@MikroTik] /interface wireless> scan wlan1
Flags: A - active, P - privacy, R - routeros-network, N - nstreme, T - tdma,
W - wds, B - bridge
            ADDRESS           SSID       CHANNEL          SIG   NF  SNR    RADIO-NAME
AP      A8:15:4D:95:65:E0 victor       2412/20-Ce/gn  -68  -95  27
AP      B4:41:7A:94:CB:B1 ChinaNe...  2412/20/gn     -71  -95  24
AP      20:DC:E6:56:A0:50 MERCURY...  2437/20-Ce/gn -85  -96  11
APR WB D4:CA:6D:62:FF:7C YuS_WiFi     2437/20/gn     -45  -96  51     D4CA6D62FF7C
AP      20:DC:E6:C7:C7:E8 tjp1206      2437/20-Ce/gn -85  -96  11
AP      28:2C:B2:63:0B:7C MERCURY...  2412/20-Ce/gn -93  -95  2
AP      9C:21:6A:AC:F3:D6 TP-LINK...  2412/20-Ce/gn -89  -95  6
```

Winbox 中可以在 wireless 功能表下找到



打開後，點擊 start 開始搜索



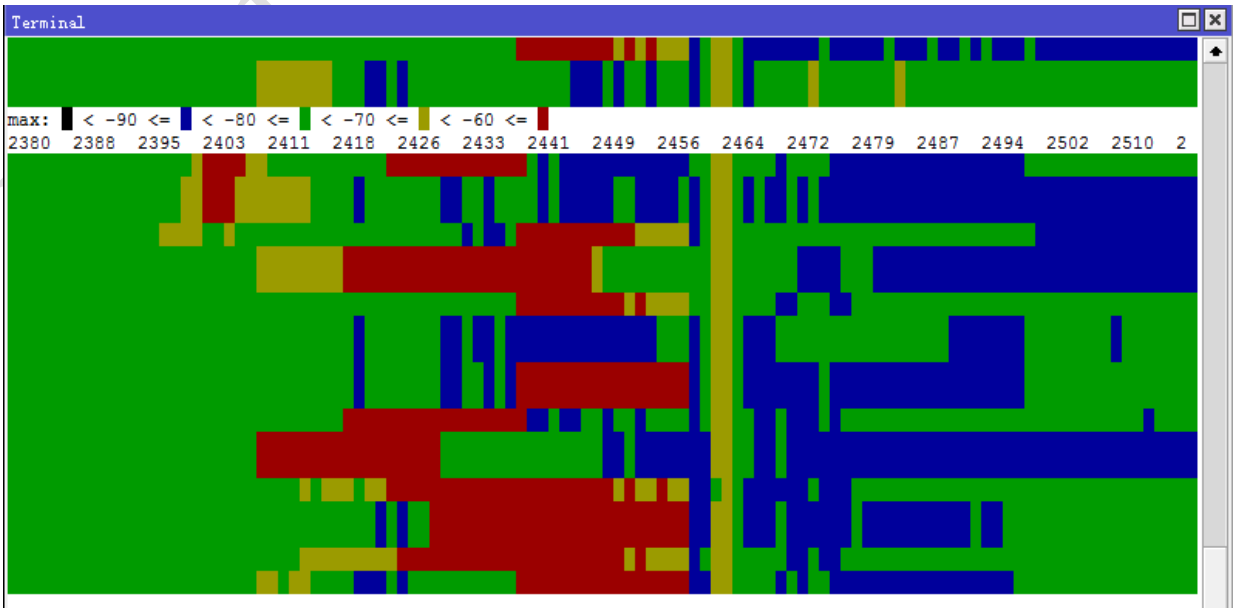
在 Scaner 功能表中，有 Connect 按鈕，當你選擇好的無線 AP 信號後，可以點 Connect 連接，這樣你的無線網卡會自動將 SSID 修改為選定的 AP 的 SSID。

12.7 無線頻譜掃描 spectral scan

spectral scan（頻譜掃描） 能掃描無線網卡支援的所有頻道，並在控制終端繪製出圖片。精確度依賴於無線網卡性能。例如 R52n 支援的頻道包括 4790～6085、 2182～2549。通過繪製頻譜圖可以得到周圍的頻率使用環境，用於分析無線頻道情況，便於選擇空閒的頻道部署自己的網路。

當前支援 Atheros 晶片，如 AR9220, AR9280, AR9223，已經測試通過對無線網卡包括：R52N 和 R2N。

頻譜歷史（Spectral history）



命令列操作如下：

```
/interface wireless spectral-history <wireless interface name>
```

繪製頻譜歷史，圖例和頻率規則每 24 行一組顯示，不同的背景顏色代表當前環境下不同的頻道功率強度，在上圖已經給出了不同顏色的區間值：

- 小於-90 為黑色
- 大於等於-90 小於-80 為藍色
- 大於等於-80 小於-70 為綠色
- 大於等於-70 小於-60 為黃色
- 大於等於-60 為紅色

功率越強代表當前環境下該頻道的干擾越強。

頻譜掃描 Spectral Scan

連續監測頻譜資料，該命令類似'spectral-history'，只是通過顯示每個頻道的具體信號強度。每行顯示一個頻譜頻率，通過字元圖形顯示， 通過”：”顯示平均功率值，平均峰值保持通過”.”.

```
Terminal
FREQ DBM GRAPH
2382 -74 :::::::::::::::::::::::::::::::::.
2388 -73 :::::::::::::::::::::::::::::::::::.
2394 -73 ::::::::::::::::::::::::::::::::::.
2400 -72 :::::::::::::::::::::::::::::::::::.
2406 -75 :::::::::::::::::::::::::::::::::.......
2411 -73 :::::::::::::::::::::::::::::::::::........
2417 -75 :::::::::::::::::::::::::::::::::......
2423 -78 :::::::::::::::::::::::::::::.
2429 -77 :::::::::::::::::::::::::::::.
2435 -80 ::::::::::::::::::::::::::.
2441 -37 :::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::.
2446 -36 :::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::.
2452 -56 ::::::::::::::::::::::::::::::::::::::::::::::::::::::::.
2458 -56 ::::::::::::::::::::::::::::::::::::::::::::::::::::::::.
2464 -70 ::::::::::::::::::::::::::::::::::::::.
2470 -80 ::::::::::::::::::::::::::.
2476 -80 :::::::::::::::::::::::::.
2481 -81 ::::::::::::::::::::::::.
2487 -82 :::::::::::::::::::::::.
2493 -83 :::::::::::::::::::::.
2499 -74 :::::::::::::::::::::::::::::::::.
2505 -73 ::::::::::::::::::::::::::::::::::.
2511 -74 :::::::::::::::::::::::::::::::::.
2516 -75 ::::::::::::::::::::::::::::::::.
-- [Q quit|D dump|C-z pause|down]
```

命令列操作：

```
/interface wireless spectral-scan <wireless interface name>
```

show-interference - 新增一欄目為探測干擾源類型，干擾源類型如下：

- bluetooth-headset
- bluetooth-stereo
- cordless-phone
- microwave-oven
- cwa
- video-bridge
- wifi

參考文獻:

http://wiki.mikrotik.com

http://www.routerboard.com

https://ros.tw/wp/

相關網路技術資料