

How to

MIKROTIK

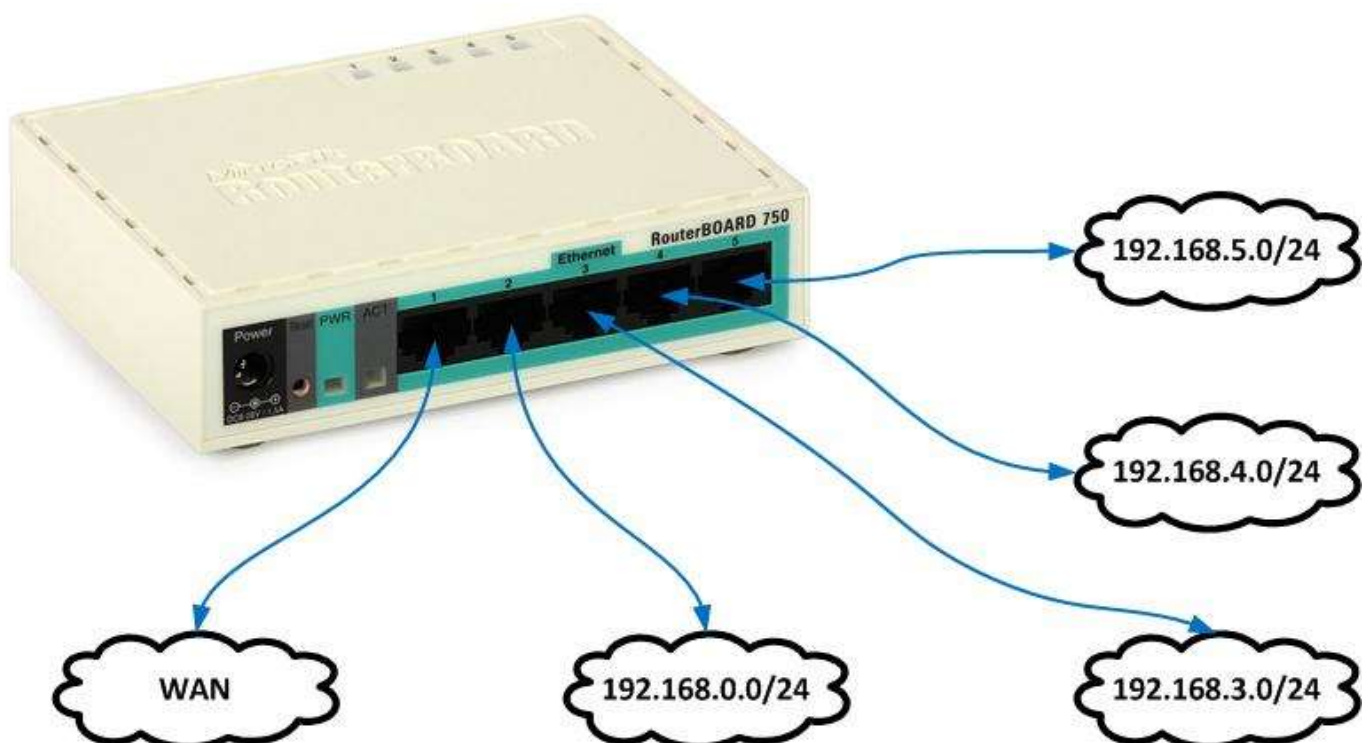
HOW TO RUN MULTIPLE NETWORKS FROM A MIKROTIK

06/04/2017 | THE PUNISHER

5 Port Router?

Since a Mikrotik (example being a 750GL) is a true 5 port router and not just a consumer-grade router (which is actually a router with 4 port switch), you can set them up to run multiple networks, use multiple ISPs for WAN failover, and more. Why you would want to do any of these things is beyond the scope of this post, but this will show you how to do it.

Let's take an RB750GL and have it run 4 internal networks instead of one. Like this:

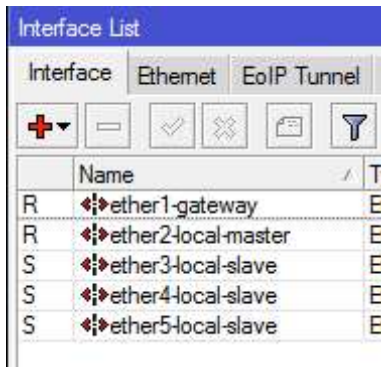


Master Port

Out of the box, the 750 is set up for ether1 to be the WAN port and ether2, ether3, ether4, and



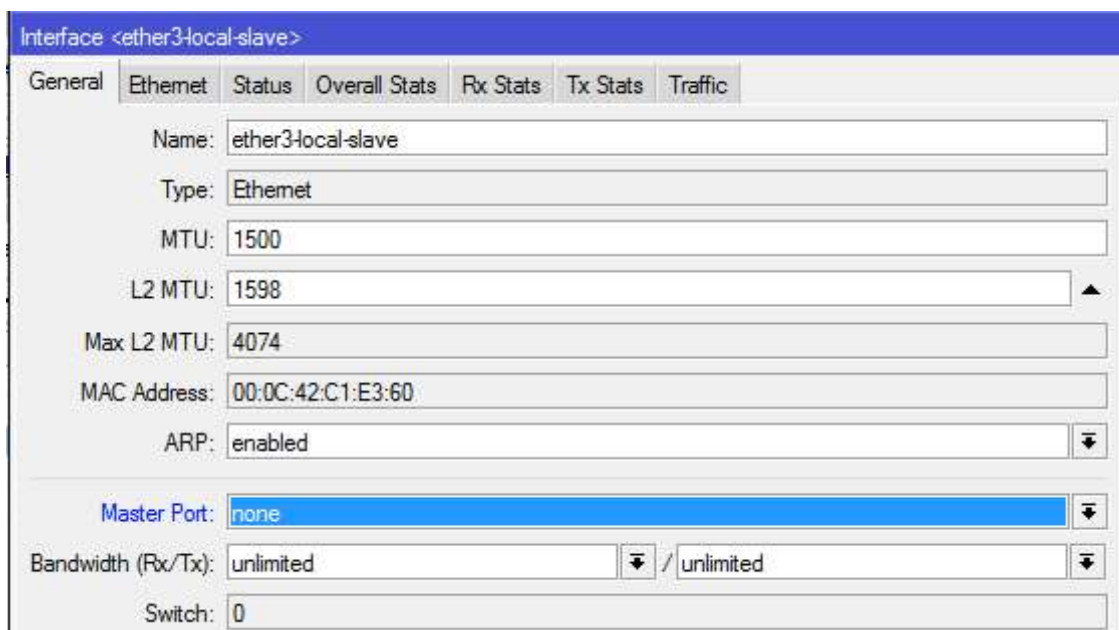
ether3, 4, and 5 have their **Master Port** set to ether2. The ports are even “name” as such, and there is an “S” in the left column showing which ports are “slaves”:



	Name	Type	Status
R	ether1-gateway	E	E
R	ether2-local-master	E	E
S	ether3-local-slave	E	S
S	ether4-local-slave	E	S
S	ether5-local-slave	E	S

If we want ether3-5 to run separate networks, we need to set them to have **NO Master Port**. It's important to note that the “Name” of the interface will not change based on your settings. If you change ether3 from a Slave to a Master, the name will **not** update on its own. It's just a name. You could call it “Port 3” or “Fred” or “I have my NAS plugged in here.” It doesn't really matter to the functionality of the router... Don't get confused by the port names when you start making changes.

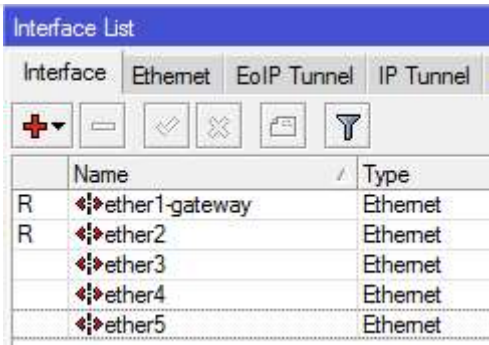
Change the Master Port on ether3, 4, and 5 like this (and change the interface's name if you like):



General	Ethernet	Status	Overall Stats	Rx Stats	Tx Stats	Traffic
Name:	ether3-local-slave					
Type:	Ethernet					
MTU:	1500					
L2 MTU:	1598					
Max L2 MTU:	4074					
MAC Address:	00:0C:42:C1:E3:60					
ARP:	enabled					
Master Port:	none					
Bandwidth (Rx/Tx):	unlimited		/ unlimited			
Switch:	0					

When you've set up ether3-5 to have no Master Port, none of the ports should show an “S” in the left column.





	Name	Type
R	ether1-gateway	Ethernet
R	ether2	Ethernet
	ether3	Ethernet
	ether4	Ethernet
	ether5	Ethernet

Now, for a port to run its own network, it needs a few things:

- IP Address
- DHCP Server
- Route

IP Addresses

To give each port its own IP address, go into **IP**, then **Addresses**. Click the +, type in the **Address** you want to give the port, type in the **Network** you want to assign to the port, and select the port from the **Interface** drop-down menu. To set up the 192.168.3.xyz subnet on ether3, it would look like this:



Address <192.168.3.1/24>

Address: 192.168.3.1/24

Network: 192.168.3.0

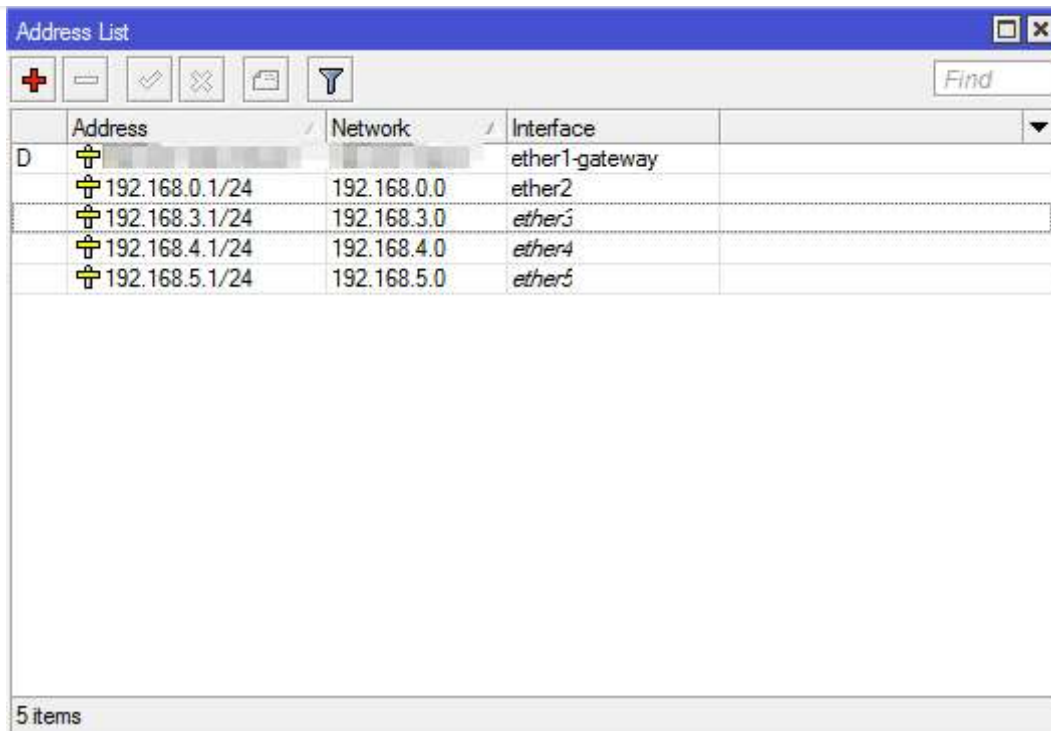
Interface: ether3

enabled

Buttons: OK, Cancel, Apply, Disable, Comment, Copy, Remove

When all 4 subnets are set up, your **Address List** screen should look similar to this (ether3, 4, and 5 are in italics because there is nothing plugged into them):





	Address	Network	Interface
D	+		ether1-gateway
	+	192.168.0.1/24	192.168.0.0 ether2
	+	192.168.3.1/24	192.168.3.0 ether3
	+	192.168.4.1/24	192.168.4.0 ether4
	+	192.168.5.1/24	192.168.5.0 ether5

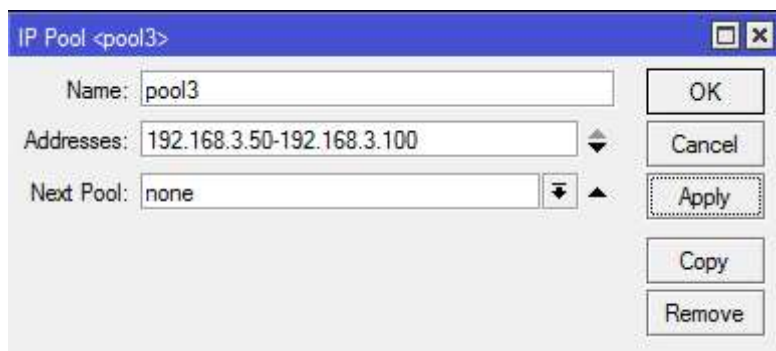
5 items

DHCP Servers & IP Pools

Next is to set up a DHCP server for each port. It's the same concept as [changing the default DHCP range](#), but you're just adding additional DHCP servers and IP Pools for your network to use.

First set up the additional IP Pools you're going to use for each server. You'll want to do this first so that when you're setting up the new DHCP Server you can just select the new IP Pool from the drop-down instead of having to close it, go back and create the new Pool, then go back to creating the new DHCP server.

Go to **IP / Pool** and click the +. Name the Pool whatever you want (like pool1 for the .1 subnet, pool2 for the .2 subnet, or whatever you find easy to remember). Then enter the **Addresses** as the DHCP range you want to use for that port – so for the .3 subnet you may want something like **192.168.3.50-192.168.3.100**.



IP Pool <pool3>

Name: pool3

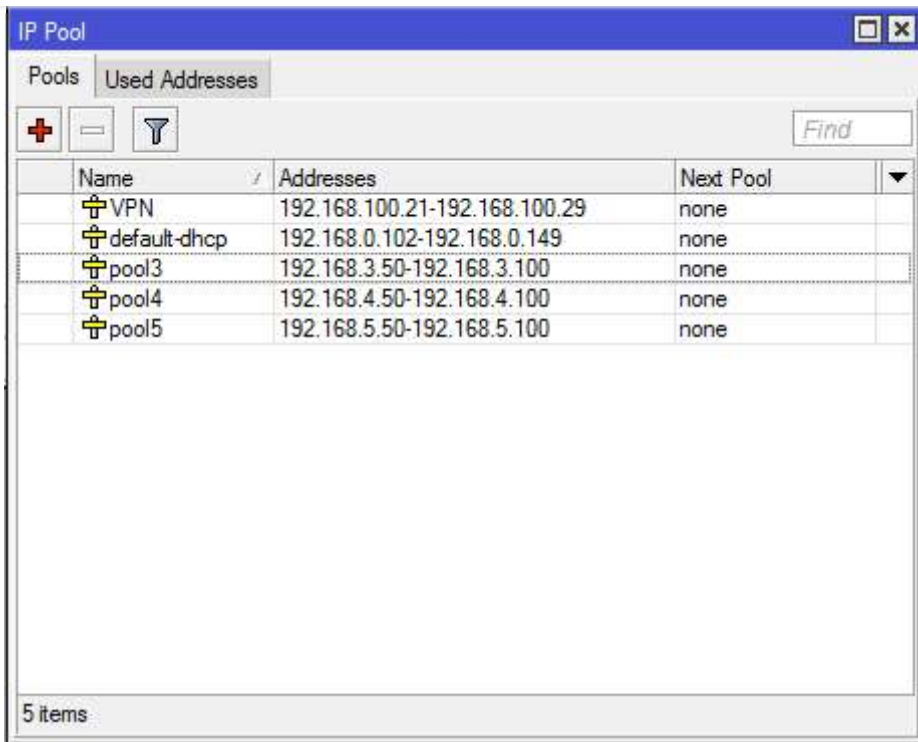
Addresses: 192.168.3.50-192.168.3.100

Next Pool: none

OK Cancel Apply Copy Remove

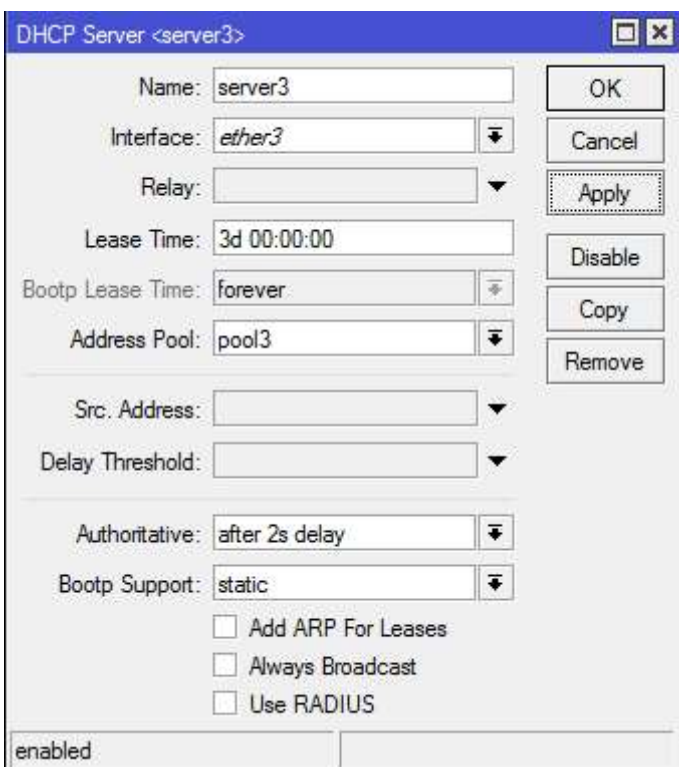
Do this for each port and your **Pool** screen should look something like this (I have a [VPN](#) pool set up as well, and my default-dhcp was already set up and I didn't want to change it just for these screenshots):





At this point those IP Pools aren't being used by anything. You have to set up new DHCP Servers to use them.

Go to **IP / DHCP Server / DHCP** and click the +. Name the new server whatever you like, set the **Interface** to the port you want to have use this server, and set the Address Pool you want this Server to draw from. In other words...



Do this for each port and your **DHCP Server** screen should look something like this:



DHCP Server						
<div> <div>DHCP</div> <div>Networks</div> <div>Leases</div> <div>Options</div> <div>Alerts</div> </div> <div> <div>+</div> <div>-</div> <div>↕</div> <div>✕</div> <div>⌵</div> <div>DHCP Config</div> <div>DHCP Setup</div> </div>						
Name	Interface	Relay	Lease Time	Address Pool	Add AR...	
default	ether2		3d 00:00:00	default-dhcp	no	
server3	ether3		3d 00:00:00	pool3	no	
server4	ether4		3d 00:00:00	pool4	no	
server5	ether5		3d 00:00:00	pool5	no	

Next you have to set up your DHCP Networks, so that each DHCP Client will receive the correct DHCP information like what its Gateway and DNS servers are. For example:

DHCP Network <192.168.5.0/24>

Address: 192.168.5.0/24

Gateway: 192.168.5.1

Netmask:

DNS Servers: 192.168.5.1

Domain:

WINS Servers:

NTP Servers:

Next Server:

Boot File Name:

DHCP Options:

OK

Cancel

Apply

Comment

Copy

Remove

After you do this for all 3 new DHCP servers (not counting the one that was already set up on ether2), your **DHCP Server / Networks** screen should look like this:

DHCP Server						
<div> <div>DHCP</div> <div>Networks</div> <div>Leases</div> <div>Options</div> <div>Alerts</div> </div> <div> <div>+</div> <div>-</div> <div>↕</div> <div>⌵</div> </div>						
Address	Gateway	DNS Servers	Domain	WINS Servers	Next Server	
192.168.0.0/24	192.168.0.1	192.168.0.1				
192.168.3.0/24	192.168.3.1	192.168.3.1				
192.168.4.0/24	192.168.4.1	192.168.4.1				
192.168.5.0/24	192.168.5.1	192.168.5.1				

At this point, if you plug your laptop into ether3, it will grab an IP address in the 192.168.3.xyz range. If you unplug it and plug it into ether5, it will grab an IP address in the 192.168.5.xyz range.

Routes



Last step, which isn't really a 'step' but you need to know about it, is what Routes are set up for these new networks you've set up. Look at this screenshot.

	Dst. Address	Gateway	Distance	Routing Mark	Pref. Source	
DAS	0.0.0.0/0	! reachable ether1-gateway	1			
DAC	192.168.0.0/24	ether1-gateway reachable	0		98.233.145.219	
DAC	192.168.3.0/24	ether2 reachable	0		192.168.0.1	
DAC	192.168.4.0/24	ether3 unreachable	0		192.168.3.1	
DAC	192.168.5.0/24	ether4 unreachable	0		192.168.4.1	
DAC	192.168.5.0/24	ether5 unreachable	0		192.168.5.1	

I did NOT enter any of these myself. The "D" in the left column means that each Route was added Dynamically. When you set the IP Addresses for each port, as soon as you added a new Address, the router added a dynamically created Route for that network for you. In this simple scenario you just need to be aware of this, you don't need to do anything with it. *Note – the screenshot shows "unreachable" on the ports that don't have anything plugged into them.*

You now have 4 LAN networks running on your Mikrotik. Since they are all on different subnets, you will not get any "broadcast" traffic between them. You can, however, reach from one subnet to another by going to a specific IP. For example, with Control4, when you open their programming software it picks up a broadcast that the Control4 processor sends out. If you are on a different subnet you will not see it and the processor will never populate in the software. But, you can manually add the IP address of the processor and it will work fine, even if it's on a different subnet. (this isn't a suggestion of how to do it, just an example). Same goes for things like Airplay and other streaming protocols. Many of them rely on broadcasting to tell 'everyone' that they are there and waiting for you to send them a music stream. This can get rather complicated when you start trying to segment off different parts of your networks.

source:<http://networkingforintegrators.com>



 13,533 total views, 59 views today

Like 0

Tweet

Save

Share

16

 MIKROTIK  MULTIPLE  NETWORKS