MikroTik Training (Basic) Part- II

Organized by
Access Telecom (BD) Ltd

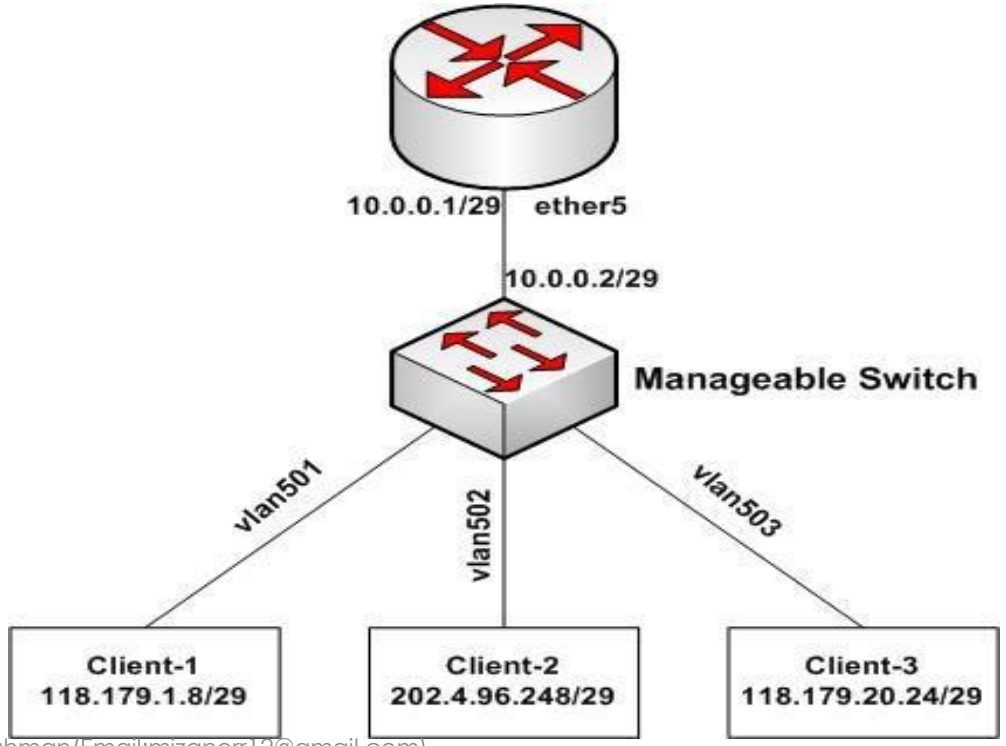
Presented by
Md. mizanor rahman
Executive of FON
Access Telecom BD Ltd.
Alap Communication Ltd.

Objectives:

- **VLAN**
- **Bridge**
- **IP-IP Tunnelling (L3 VPN)**
- **GRE Tunnelling (L3 VPN)**
- **Failover (Auto Redundancy)**
- **Administrative User Password and it's Security**
- **Backup Recovery**
- **Q&A Session**

VLAN

- A Virtual Local Area Network **(VLAN)** is a logical group of Workstations, Servers and Network Devices that appear to be on the same LAN despite their geographical distribution.



Benefits of VLAN

- The key benefits of implementing **VLANs** include:

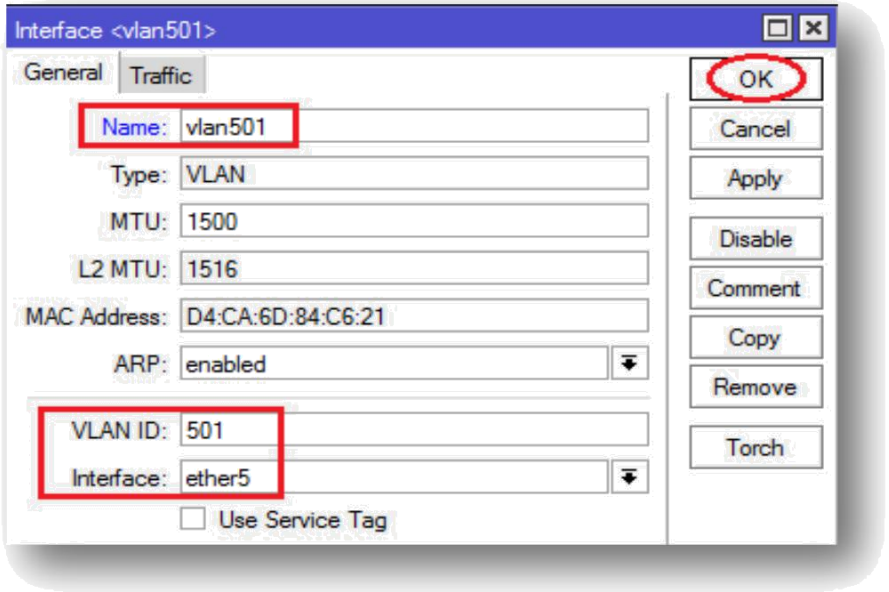
- ✓ Allowing network administrators to apply additional security to network communication.
- ✓ Making expansion and relocation of a network or a network device easier.
- ✓ Providing flexibility because administrators are able to configure in a centralized environment while the devices might be located in different geographical locations.
- ✓ Decreasing the latency and traffic load on the network and the network devices, offering increased performance.

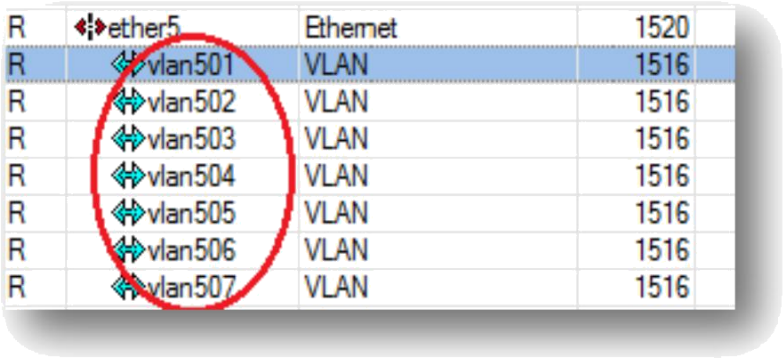
VLAN in MikroTik

How to create VLAN in MikroTik?

- ✓ **VLAN** should be created under any Ethernet Port of **MikroTik.**
- ✓ Here **VLAN** Name is not significant where as **VLAN ID** and **Interface** is significant.

Procedure: **Interface → VLAN** 



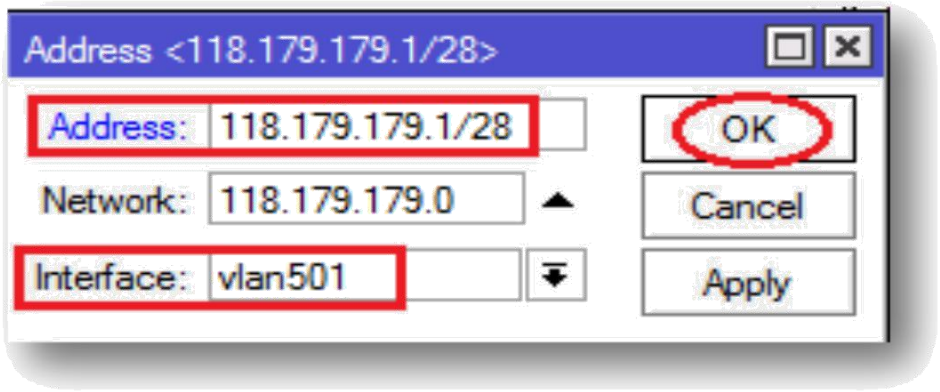


VLAN Implementation

- After creating **VLAN** under any Ethernet we'll add IP Address in **VLAN-Interface** so that it can segregate the Network Traffic

  i.e. one Interface doesn't receive packets of another Interface.

Procedure: **IP → Address** 



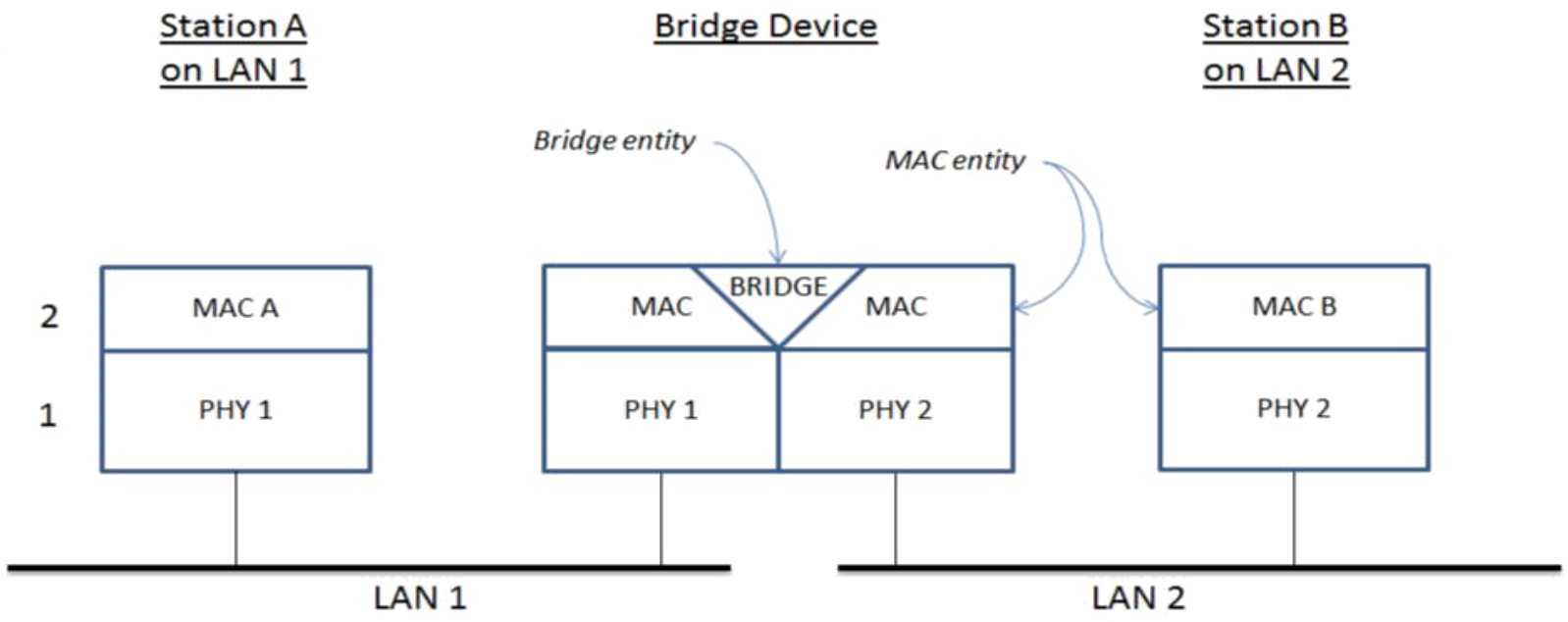
vlan501 doesn't receive any packets of **vlan502** unless its direction goes to **vlan501**

Bridge

- **Network Bridging** is the action taken by network equipment to create an aggregate network from either two or more communication networks, or two or more network segments.
- **Network Bridge** is a network device that connects multiple network segments.

Bridge Architecture

A bridge connecting two LAN segments



Bridge in MikroTik

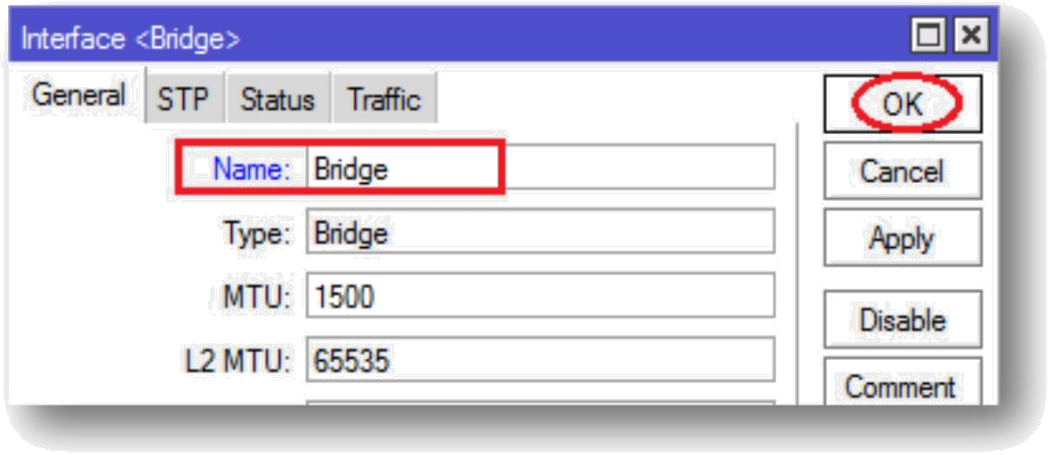
- To create a Bridge Network the steps are:

1. Create a Bridge
2. Aggregate member ports under the Bridge
3. IP Addressing for Bridge Network
4. Test

Creating a Bridge

- To create a Bridge we've to go **Bridge** from Main Menu then add a Bridge named **"Bridge".**

Procedure: **Bridge → Bridge** 

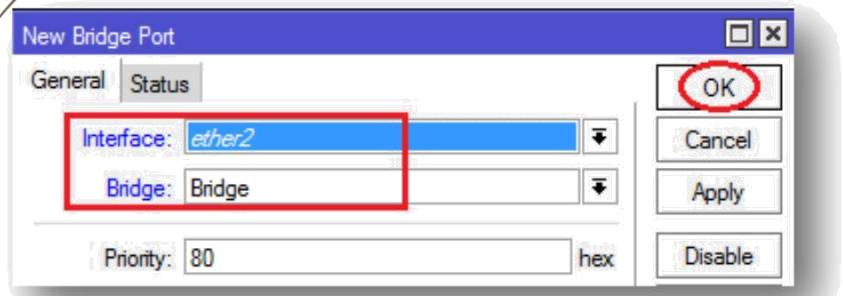


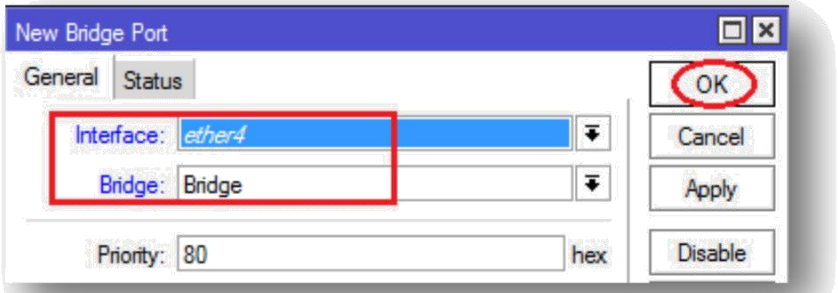
This is the **Bridge** that will aggregate two or more ports we assigned

Port Aggregation

- Now we've to aggregate the member ports under the created Bridge.

Procedure: **Bridge → Ports** 



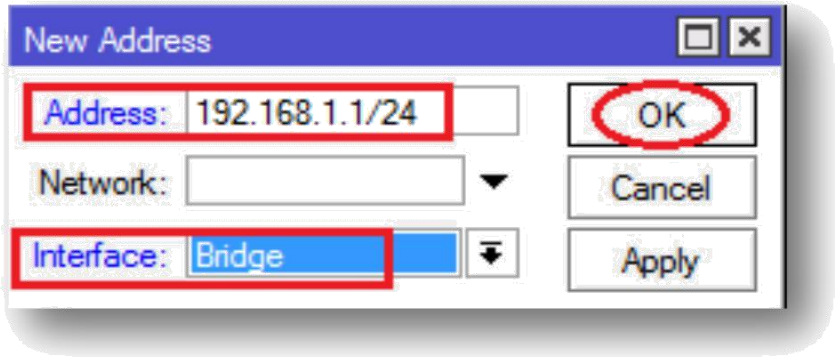


Here, we've merged **ether2** and **ether4** logically with this **Bridge** so that we can use both simultaneously or anyone as alternate

IP Addressing

- Then we've assign an IP Address in Bridge Interface so that we can test from all the member ports.

Procedure: **IP → Address** 



We'll assign an IP Address in Bridge Interface whose member ports are **ether1** and **ether2**.

Test

Test your configuration:

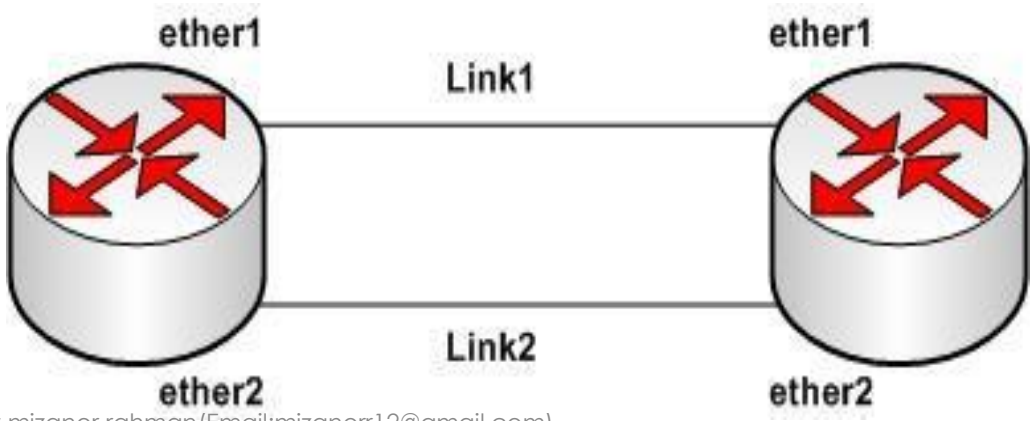
1. Set IP Address 192.168.1.10 with Mask 255.255.255.0 and Gateway 192.168.1.1 in your Laptop.
2. Connect your Laptop with ether2 of MikroTik
3. Ping your Gateway (192.168.1.1)
4. Is it responding ? [Y/N]
5. Now reconnect your Laptop with ether4 of MikroTik
6. Ping your Gateway (192.168.1.1) again
7. Is it responding ? [Y/N]

Spanning Tree Protocol (STP)

- The **Spanning Tree Protocol** (**STP**) is a network protocol that ensures a loop-free topology for any bridged Ethernet local area network.
- The basic function of **STP** is to prevent bridge loops.
- **Spanning Tree** also allows a network design to include spare (redundant) links to provide automatic backup paths if an active link fails, without the danger of bridge loops, or the need for manual enabling/disabling of these backup links.

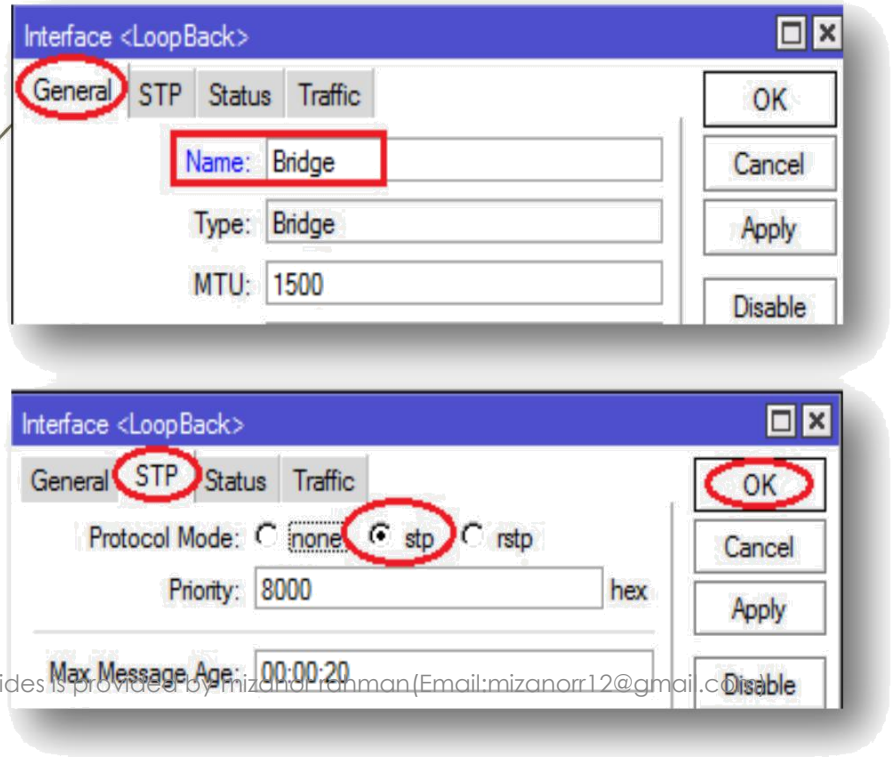
STP Topology

- This is our LAB Topology where we'll implement Link Redundancy by applying **STP**.
- In our Topology, we have two Links (**R1:ether1 to R2:ether1** and **R1:ether2 to R2:ether2**).
- Now we'll create **Bridge** in both Routers according to our LAB Topology.



STP in MikroTik

- We'll create a Bridge named **"Bridge"** and enable **STP** on that Bridge.



Applying **STP** on Bridge on port will be in active mode **(root port)** and another port will be in inactive mode **(alternate port)** which will active when root port **fail**

IP-IP Tunnelling (L3 VPN)

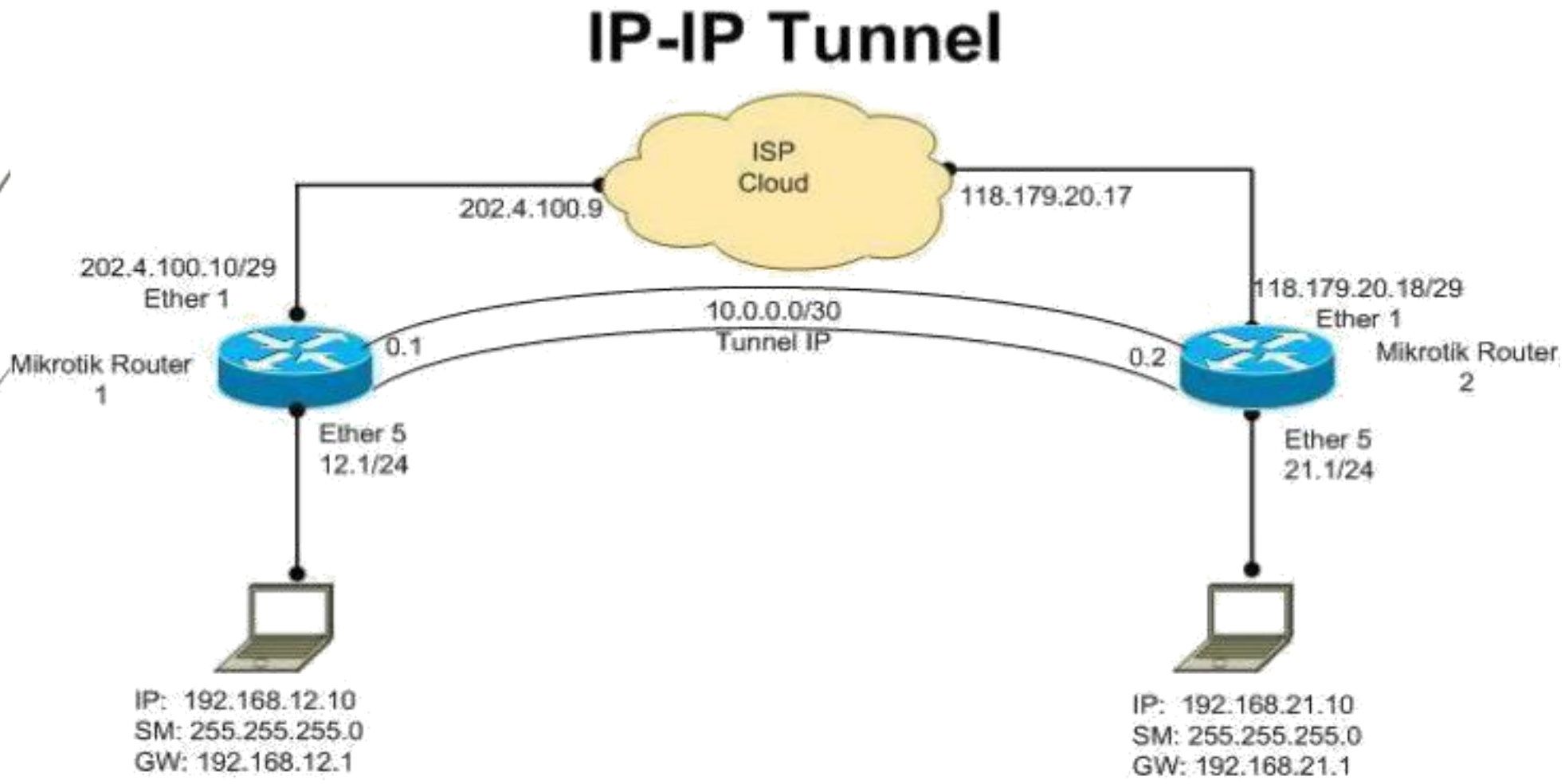
IPIP Tunnel:

- Layer 3 routed type traffic
- Inter operability with other vendors
- Based on RFC

"An **IP tunnel** is an Internet Protocol (**IP**) network communications channel between two networks. It is used to transport another network protocol by **encapsulation** of its packets."

Precondition: Before configuring IP-IP Tunnel between Two Locations, make sure that there is reach ability from Local Router to Remote Router.
The general scenario of an IP-IP Tunnel where we will create IP-IP Tunnel between **Dhaka** and **Tangail** is shown in below:

LAB Diagram for IP-IP Tunnel

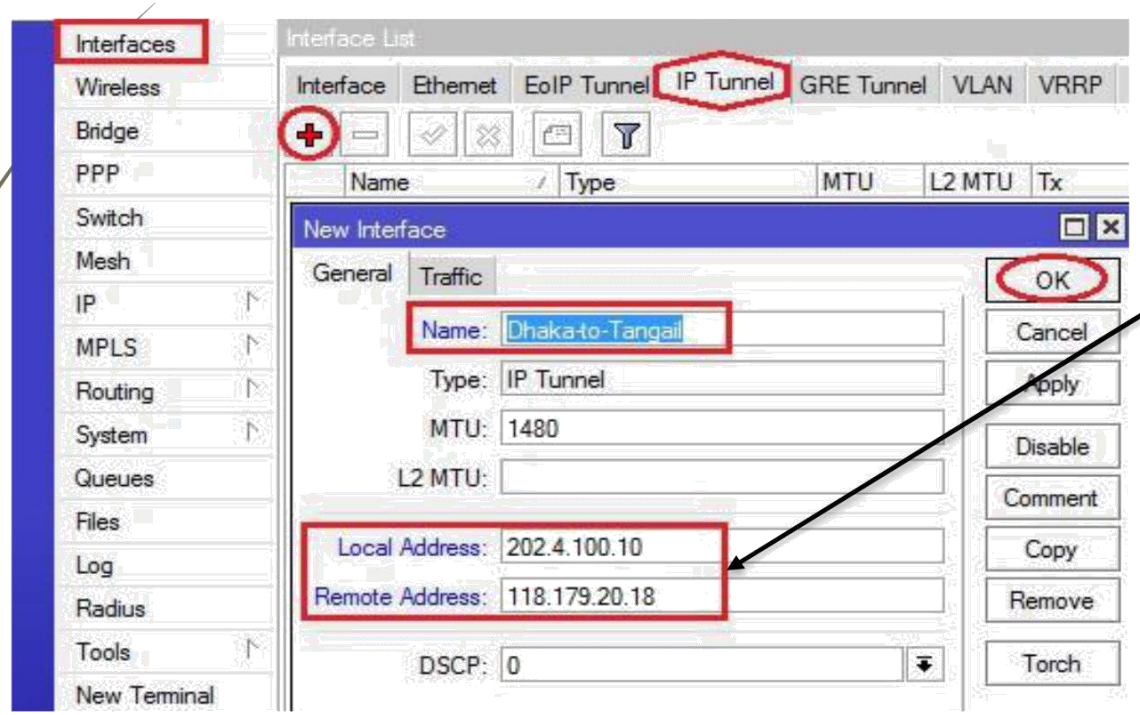


IP-IP Tunnelling (Cont.)

#Configurations of Dhaka Router

Creating a Tunnel Interface

Here we have to create a Tunnel Interface from Interface List as follows: Go to **Interface IP Tunnel**

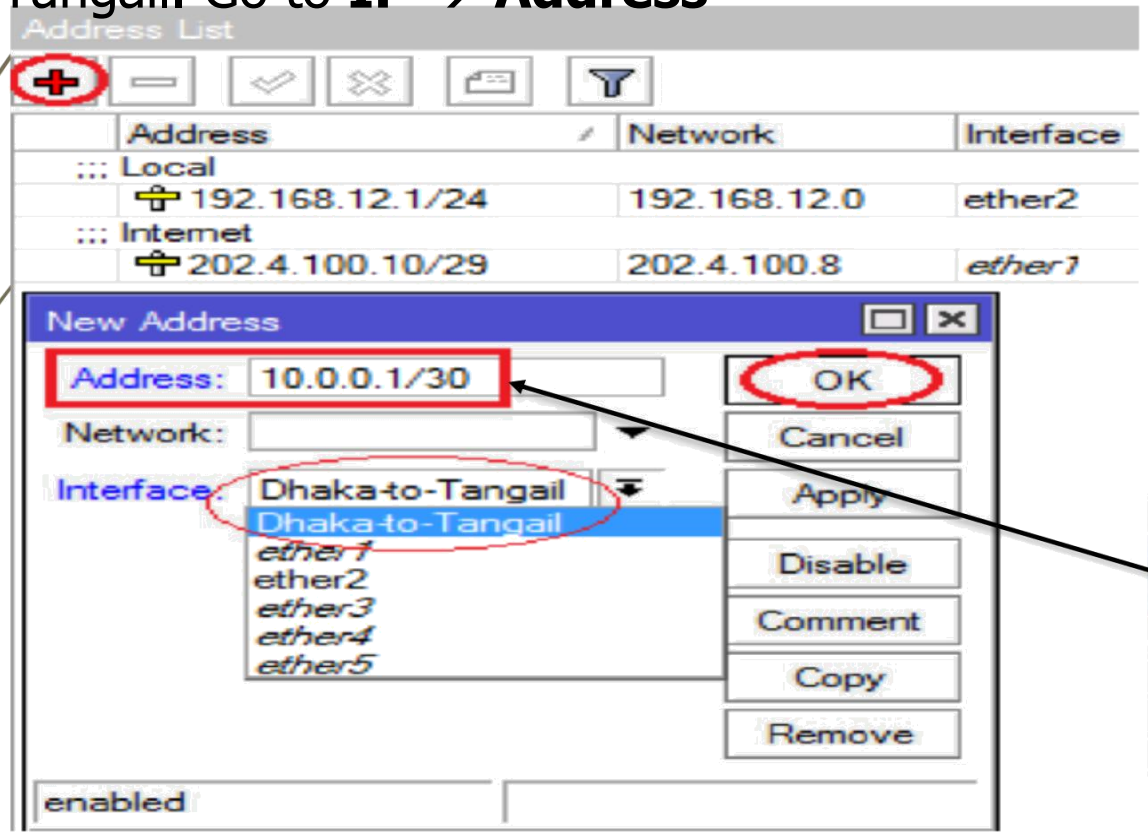


Where,
Local Address = WAN IP of Dhaka
Remote Address = WAN IP of Tangail

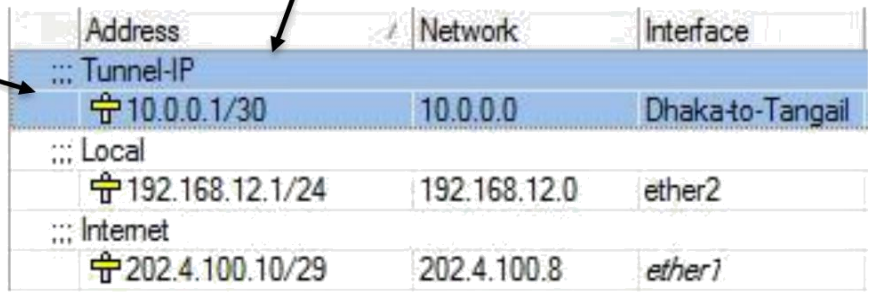
IP-IP Tunnelling (Cont.)

#Add a Tunnel IP (P2P) in Tunnel Interface

Then add a Tunnel IP (P2P) for ensuring a secure VPN between Dhaka to Tangail. Go to **IP → Address**



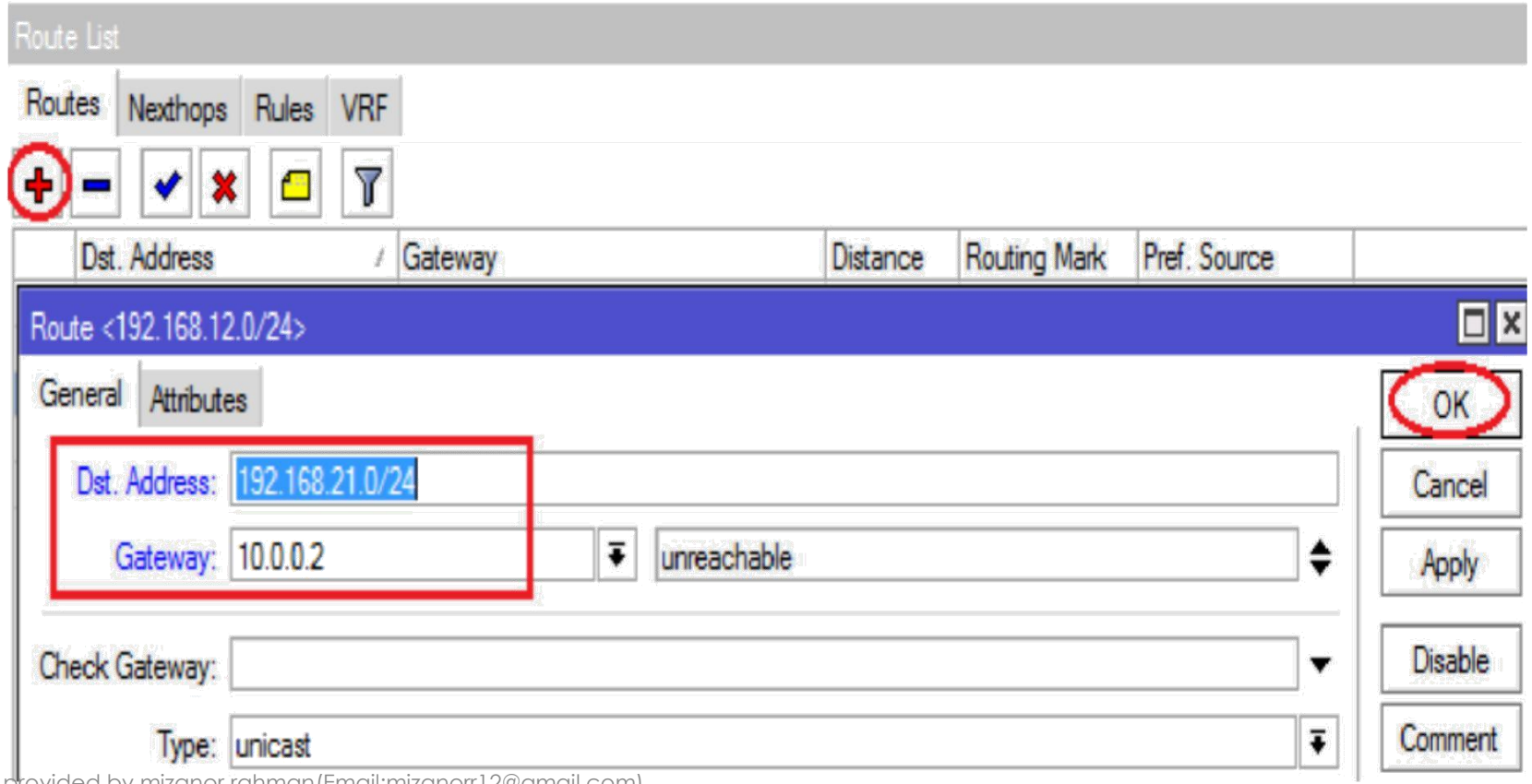
Here is shown our IP Address List after adding Tunnel IP



IP-IP Tunnelling (Cont.)

Add a Static Route on Next Tunnel Interface

We have to add a Route to Next Tunnel IP (10.0.0.2) to reach specific destination (Tangail, Prefix 192.168.21.0/24). Go to **IP Routes**

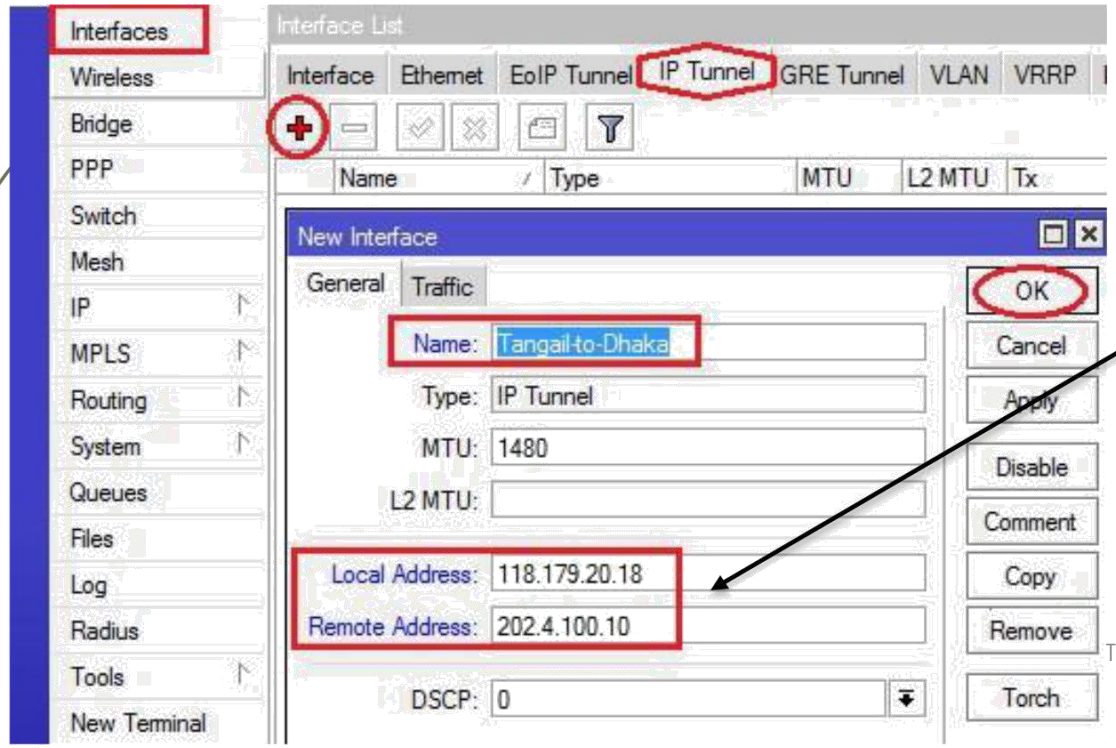


IP-IP Tunnelling (Cont.)

#Configurations of Tangail Router

Creating a Tunnel Interface

Here we have to create a Tunnel Interface from Interface List as follows: Go to **Interface □ IP Tunnel**

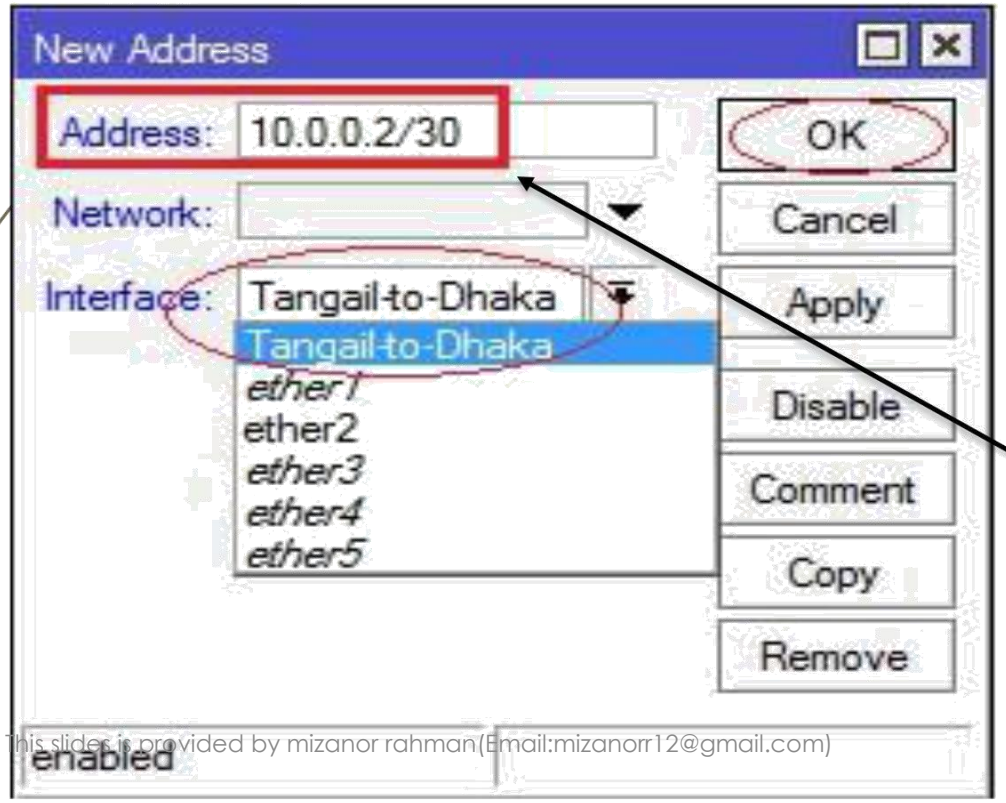


Where,
Local Address = WAN IP of Tangail
Remote Address = WAN IP of Dhaka

IP-IP Tunnelling (Cont.)

Add a Tunnel IP (P2P) in Tunnel Interface

Then add a Tunnel IP (P2P) for ensuring a secure VPN between Tangail to Dhaka. Go to **IP Address**



Here is shown our IP Address List after adding Tunnel IP

	Address	Network	Interface
;;; Tunnel-IP			
	10.0.0.2/30	10.0.0.0	Tangail-to-Dhaka
;;; Internet			
	118.179.20.18/29	118.179.20.16	*ether1*
;;; Local			
	192.168.21.1/24	192.168.21.0	ether2

IP-IP Tunnelling Testing…

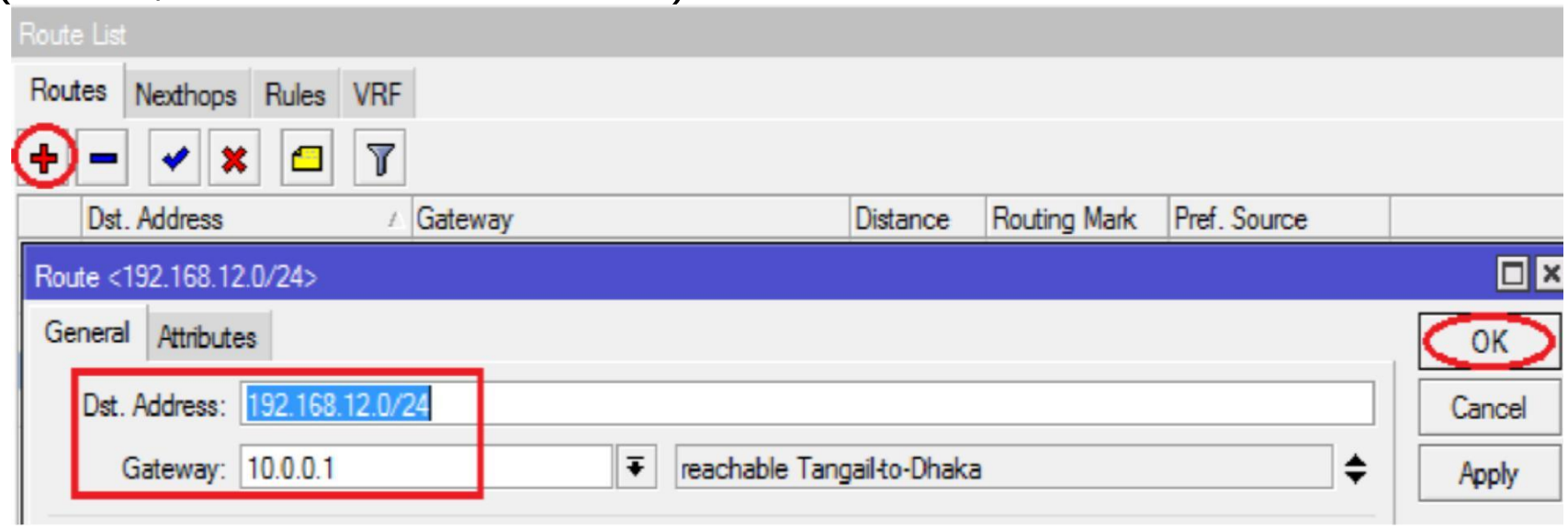
Tunnel Test by pinging Remote Tunnel IP:


```
[admin@MikroTik] > ping 10.0.0.1
HOST SIZE TTL TIME STATUS
10.0.0.1 56 64 2ms
10.0.0.1 56 64 0ms
sent=2 received=2 packet-loss=0% min-rtt=0ms avg-rtt=1ms max-rtt=3ms
```

IP-IP Tunnelling (Cont.)

Add a Static Route on Next Tunnel Interface

We have to add a Route to Next Tunnel IP (10.0.0.1) to reach specific destination (Dhaka, Prefix 192.168.12.0/24). Go to **IP Routes**

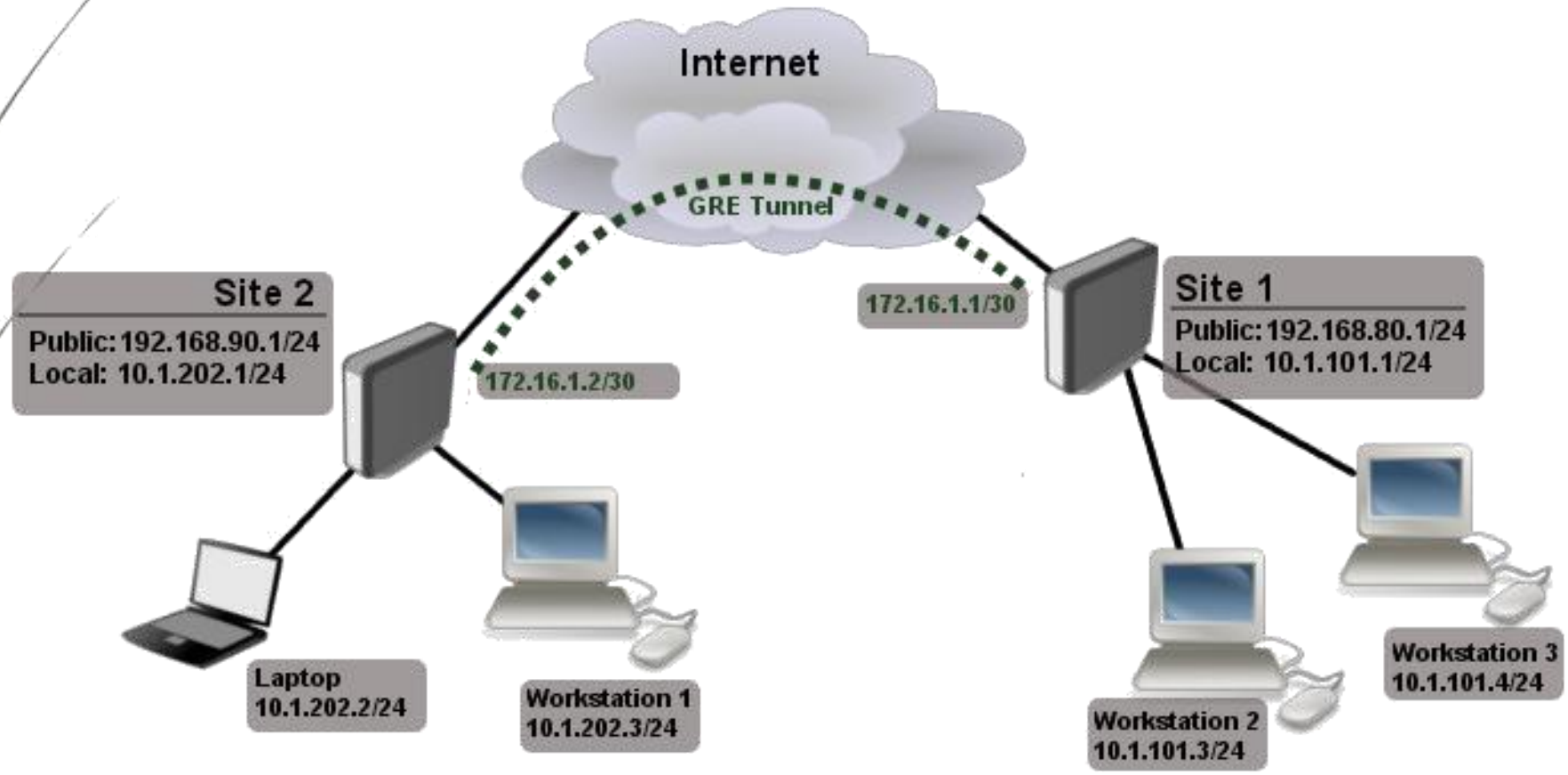


Test your Tunnel:

You can check your Tunnel is working or not by pinging both sides Local PC's and make sure that Firewall is Disabled in those PC's. If it works then will get all the facilities you have in your Local Network (File Transfer, Remote Desktop, VNC, Chatting Software etc.).

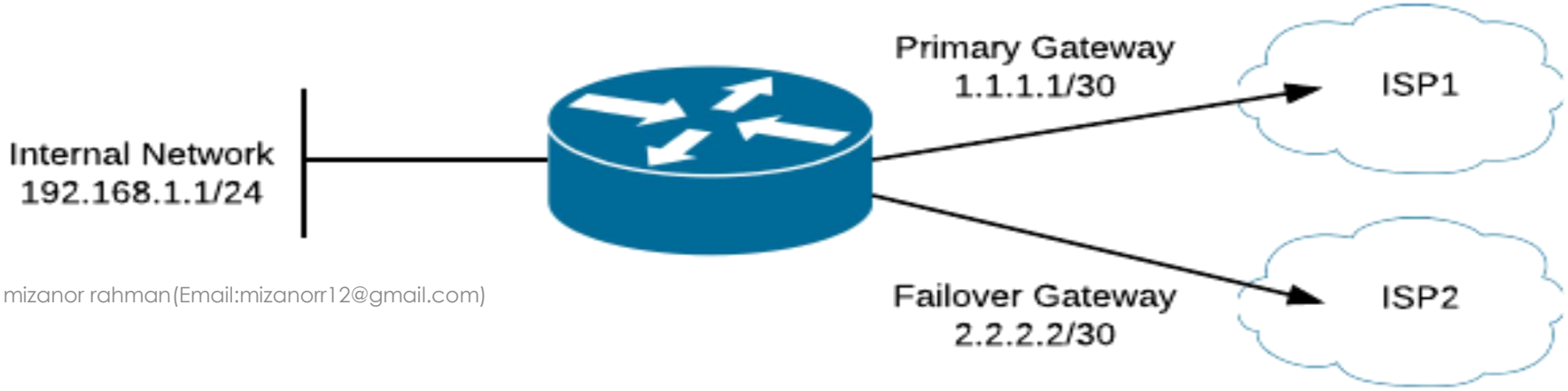
GRE Tunnelling (L3 VPN)

Almost Same with IP-IP Tunnel



Failover (Auto Redundancy)

Failover is a backup operational mode in which the functions of a system component (such as a processor, **server**, network, or database, for example) are assumed by secondary system components when the primary component becomes unavailable through either failure or scheduled down time.



Failover (Auto Redundancy)

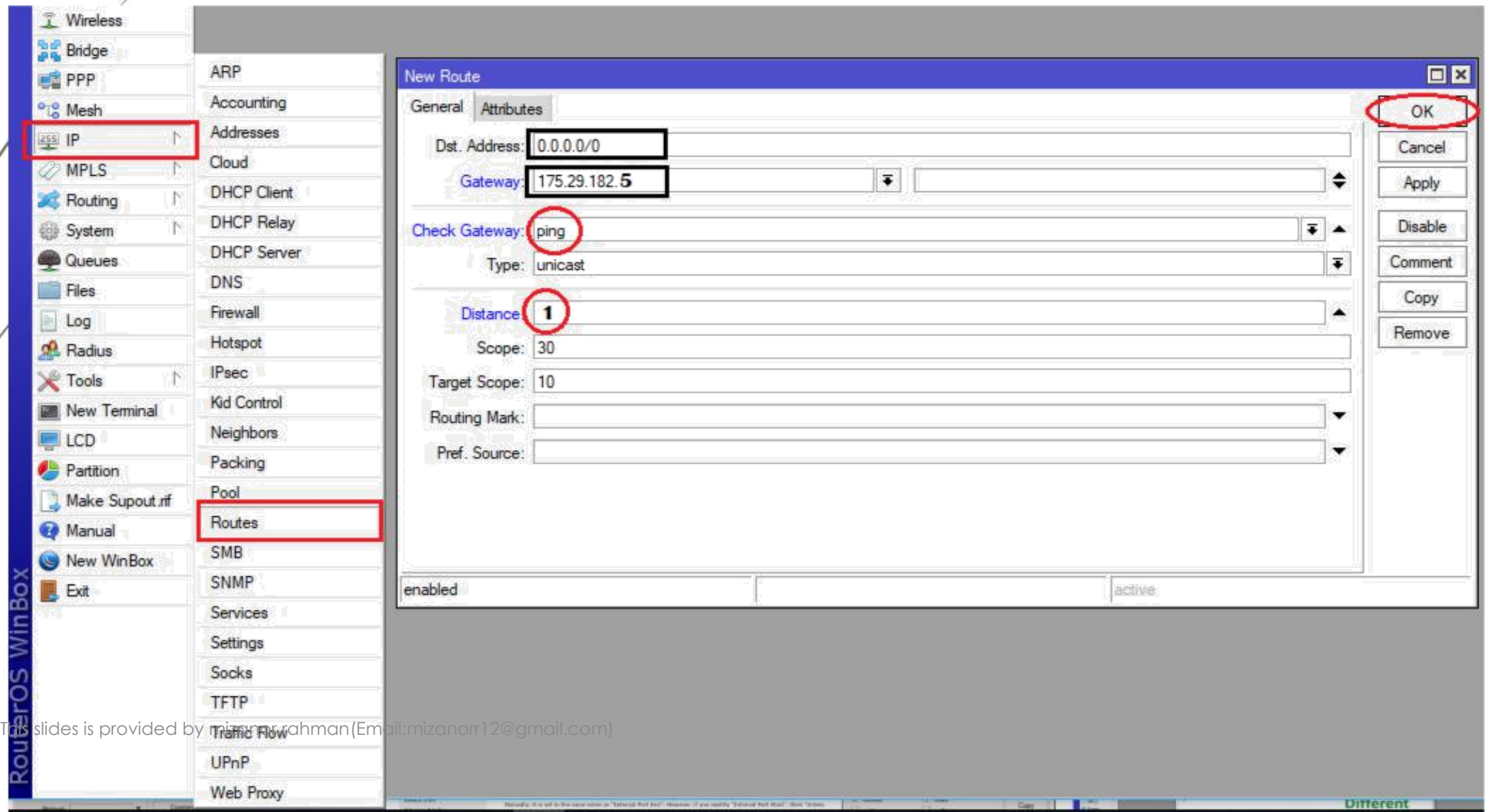
This example explains how to use multiple gateways with one taking over when first fails. It begins with adding the gateways. Set bigger **distance** value for the secondary one, and **check-gateway** for the first one:

/ip route add gateway=192.168.1.1
/ip route add gateway=192.168.2.1 distance=2

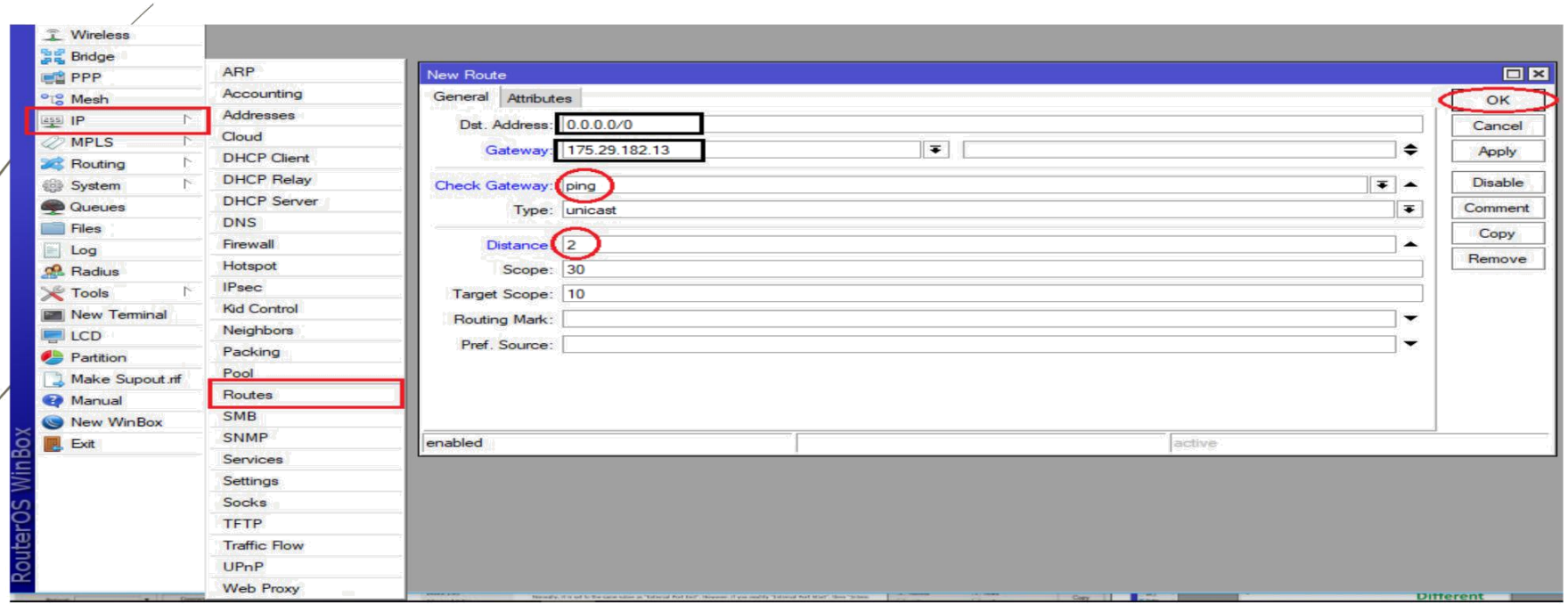
That's all, there are no more steps!

The first gateway will begin as it's distance is smaller (default 0); the check-gateway will make sure it's up; when the ping will fail, it will disable the first gateway and the second will take over; when first one comes up, it will return to it's function.

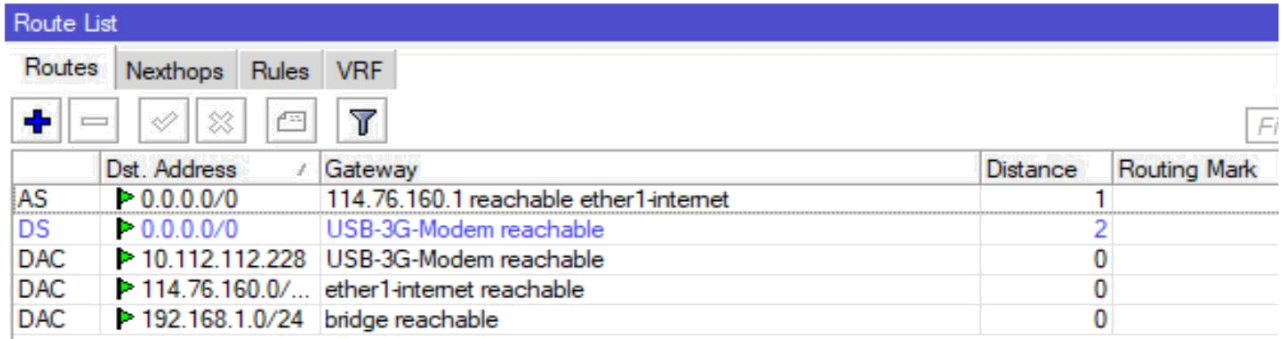
Failover (Auto Redundancy) Route-1



Failover (Auto Redundancy) Route-2



After given route
it looks like this



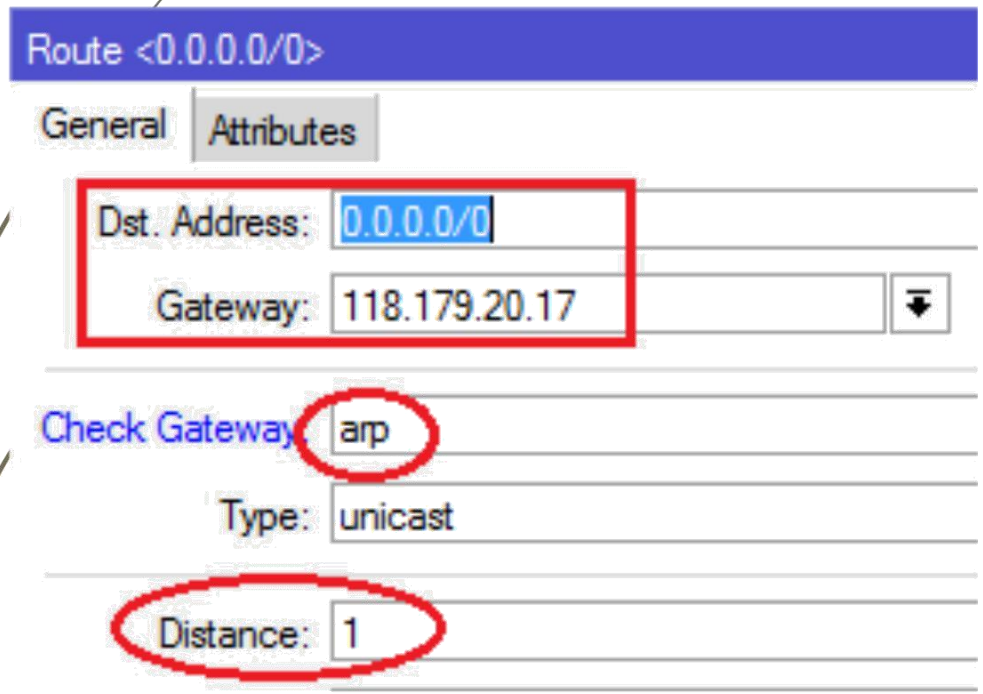
Failover (Auto Redundancy) Another-Way

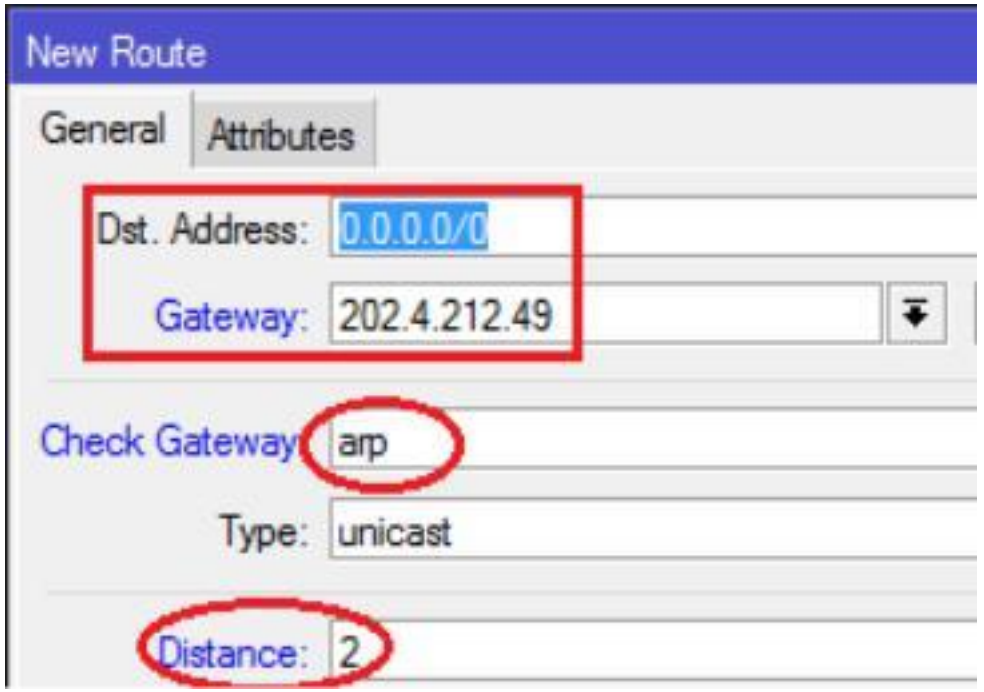
We have two vendor's connectivity or same vendor's connectivity from two separate POP's. We want to configure Failover in our MikroTik then we have to do the followings:

1. Add two IP Addresses in separate Interface (primary = ether1, secondary = ether5)
2. Add Default Route for both IP's differentiate with AD Value (say, primary = 1, secondary = 2)
3. While adding Default Route enable "**Check Gateway = arp**" **(When ICPM not Allowed)**

Note: In this situation, All the Packets will pass through primary link default because its AD value is 1. Router always sends ARP request to its Gateway. If ARP not found from Primary Link then Router learn that it goes to down. Then it sends all the Packets via Secondary Link.

Failover (Auto Redundancy)





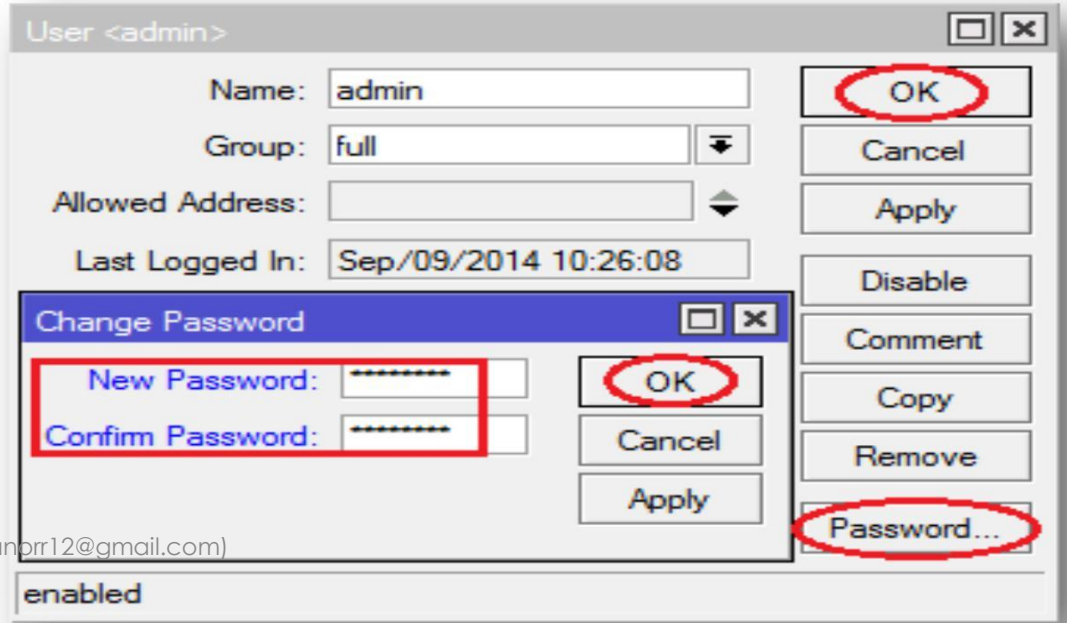
Administrative User Password and it's Security

By default **admin** user who has full permission is created with no password while running MikroTik RouterOS first time. So, after login first time it is your first duty to assign a strong password to admin user**. I always prefer to remove admin user and creating another user with full permission because admin is a known user and a hacker will always try to login to your MikroTik Router with admin user**. So, if you keep this admin user, half of his work is done and he just needs to guess your password to login in your MikroTik router. On the other hand, if you remove admin user, it will be very difficult to guess your full permission user as well as his password.

User Administration

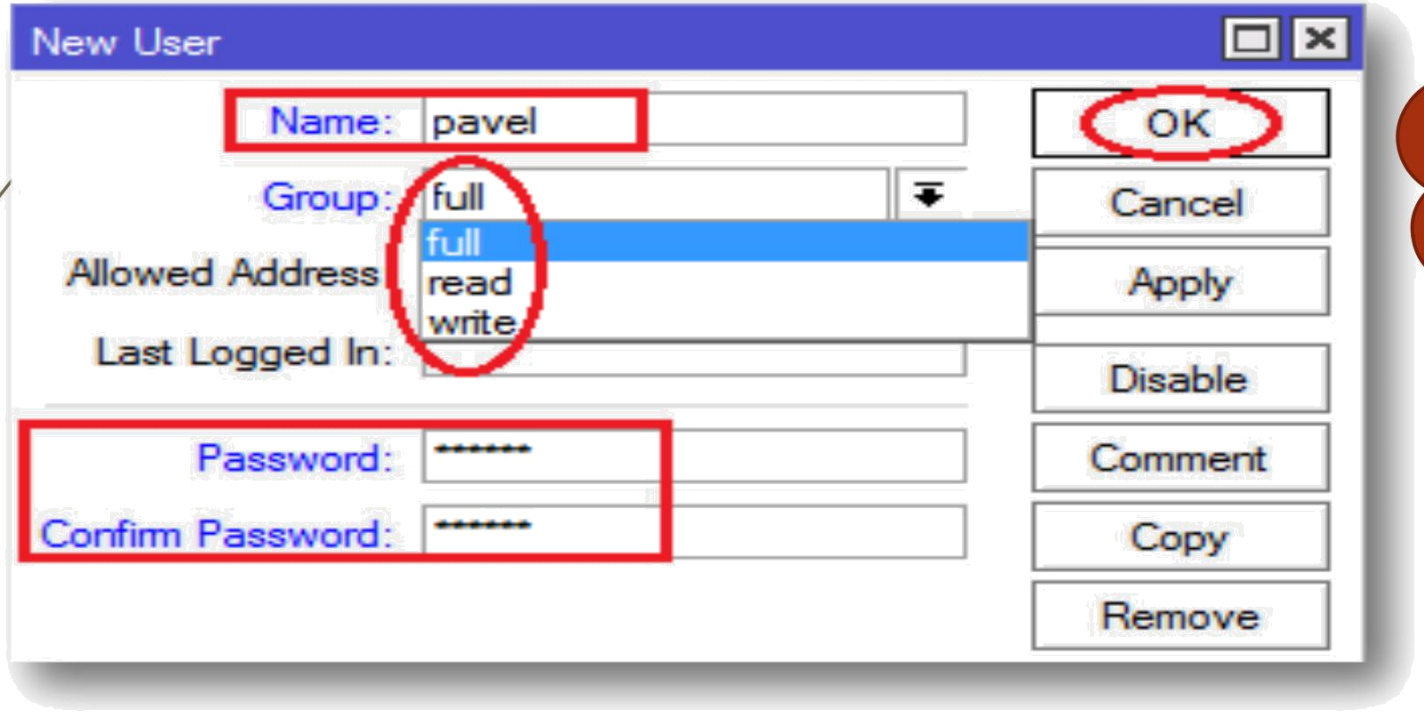
- MikroTik has a default User **"admin"** with **"empty"** Password. We've to set Password for User **"admin"** for security issue.

Procedure: **System → Users → Users** then double click on User "admin"



Create a New User

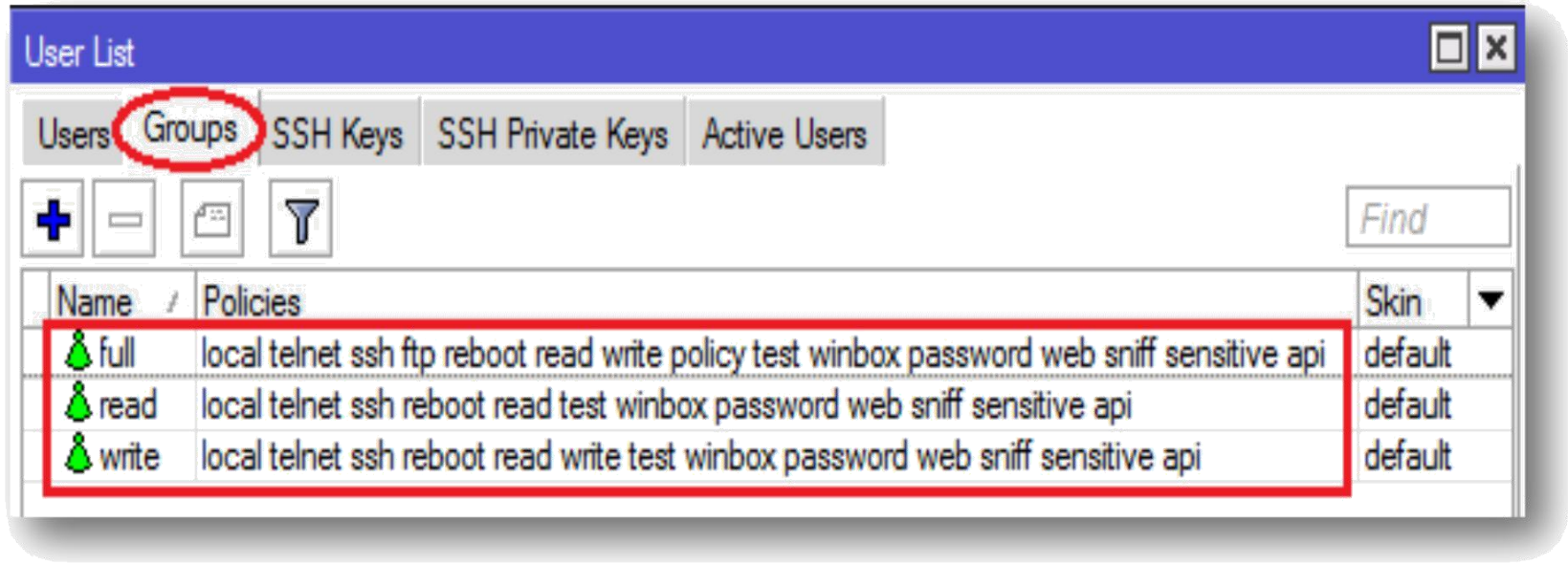
Procedure: **System → Users → Users** 



Then Remove the admin user

User Group

- There are three (03) types of User (full, write and read) with the following permissions:



User Group

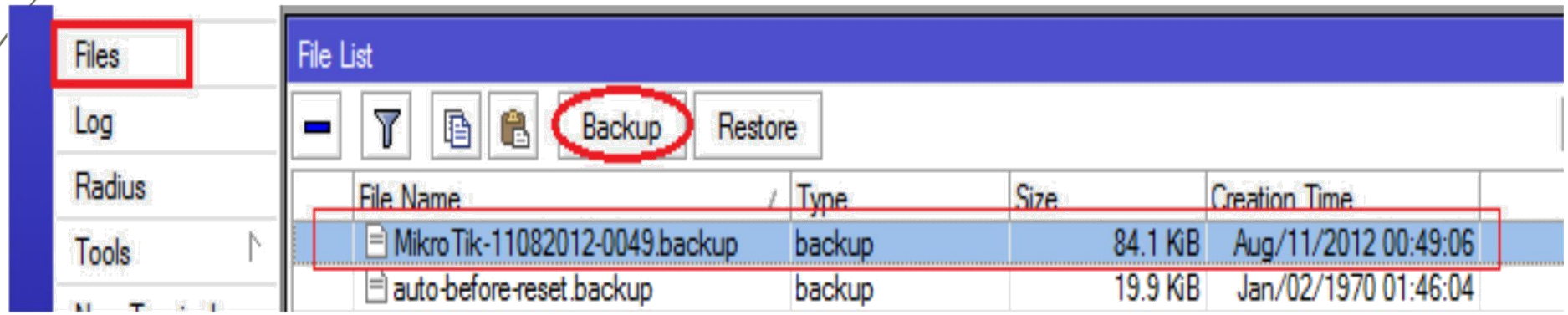
Here, the argument name is the name of the group,
and policy contains the list of policies assigned to the group:

local - User can log on locally via console
telnet - User can log on remotely via telnet
ssh - User can log on remotely via secure shell
ftp - User can log on remotely via ftp and send and retrieve files from the router
reboot - User can reboot the router
read - User can retrieve the configuration
write - User can retrieve and change the configuration
policy - Manage user policies, add and remove user
test - User can run ping, traceroute, bandwidth test **web**
- user can log on remotely via http (Java Console)
ppp - User can log on using ppp connections to the router (PPP, PPTP, PPPoE)

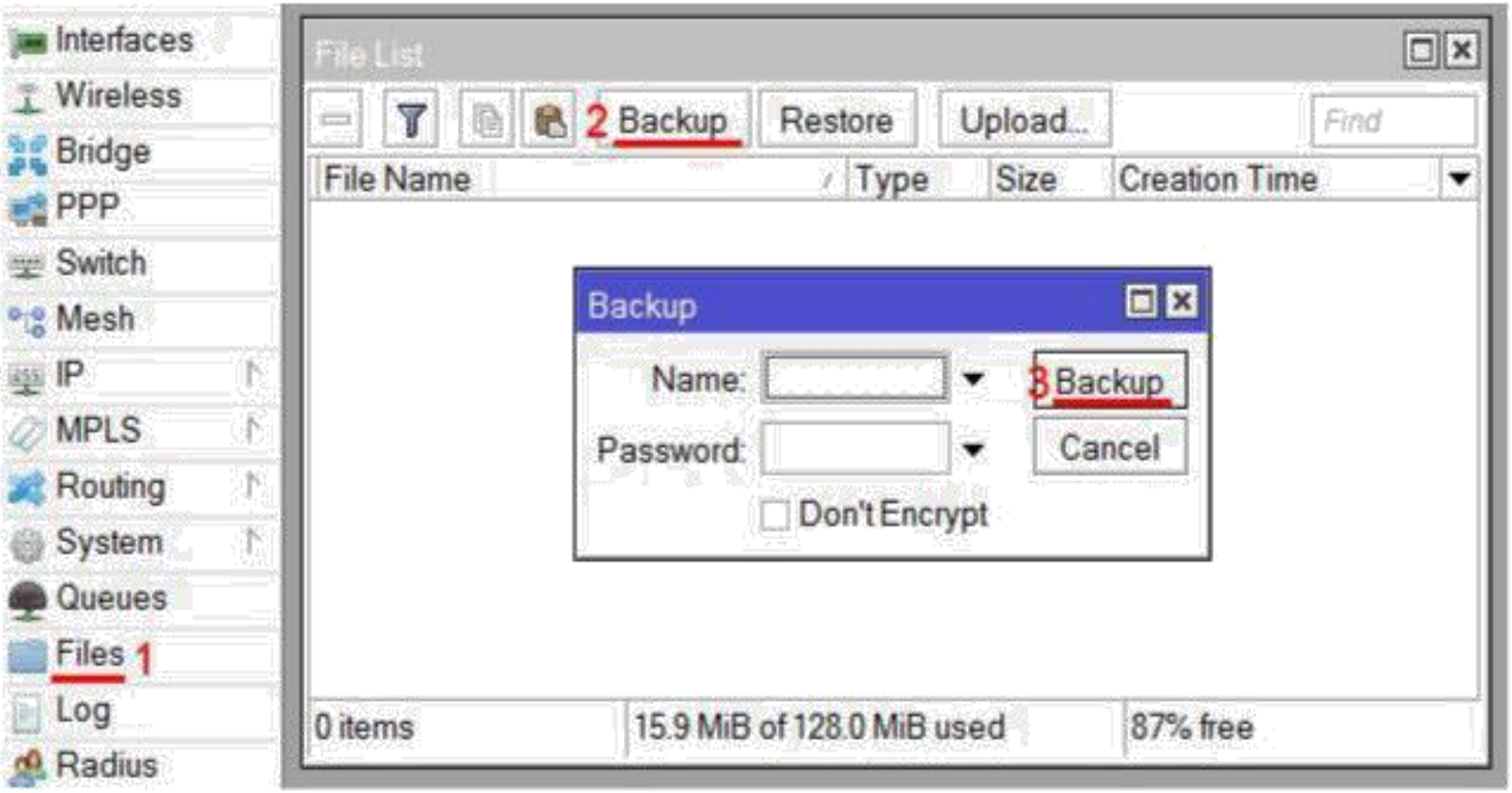
Backup & Recovery

Creating Backup:

1. Go to Files Menu
2. Then Click on Backup Button
3. Backup will create



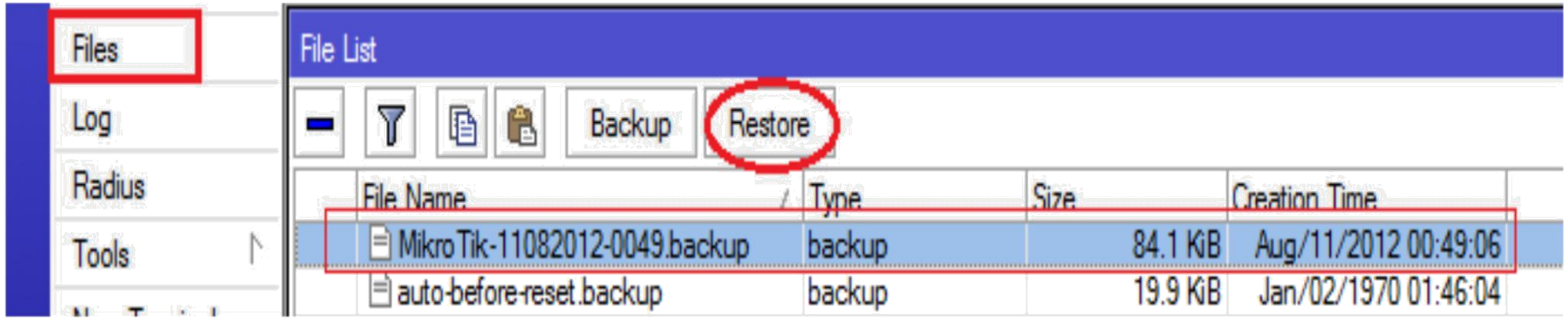
Another Screenshot



Backup Restoration:

Process:

1. Go to Files Menu
2. Paste the Backup File then Select the Backup File
3. Then Click on Restore Button then Backup will Restore



Exporting Configuration

Command name: */export*

The **export** command prints a script that can be used to restore configuration. The command can be invoked at any menu level, and it acts for that menu level and all menu levels below it. The output can be saved into a file, available for download using FTP.

Command Description:

#export file=[filename] - saves the export to a file

Example

Example

```
[admin@MikroTik] > ip address print
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.1.0.172/24      10.1.0.0        10.1.0.255      bridge1
 1   10.5.1.1/24        10.5.1.0        10.5.1.255      ether1
[admin@MikroTik] >
```

To make an export file:

```
[admin@MikroTik] ip address> export file=address
[admin@MikroTik] ip address>
```

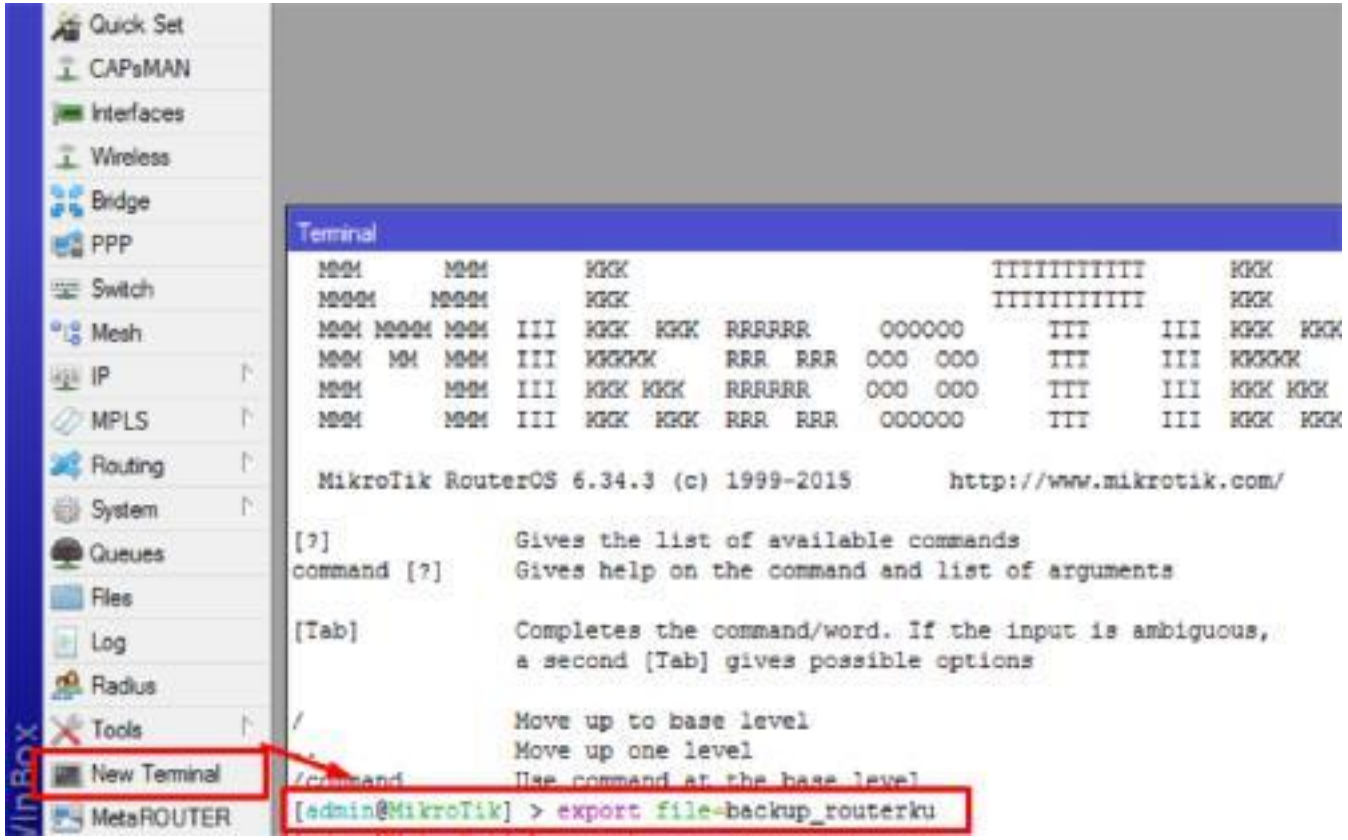
To see the files stored on the router:

```
[admin@MikroTik] > file print
 # NAME                              TYPE          SIZE       CREATION-TIME
0  address.rsc                       script        315        dec/23/2003 13:21:48
[admin@MikroTik] >
```

Exporting Full Configuration

A backup file is created on files with the name of backup_routerku

We can view this file from files tab



Importing Configuration

Command name: */import*

The root level command **/import [file_name]** executes a script stored in the specified file. It will add the configuration from the specified file to an existing configuration. This file may contain any console commands, including scripts. Can be used to restore configuration or parts of it after configuration loss.

Command Description

import file=[filename] - loads the exported configuration from a file to router

Example

Example

To load the saved export file use the following command:

```
[admin@MikroTik] > import address.rsc
Opening script file address.rsc

Script file loaded and executed successfully
[admin@MikroTik] >
```

We can also import from New Terminal. We need to Open .rsc file via notepad then we can copy commends and past on New Terminal.

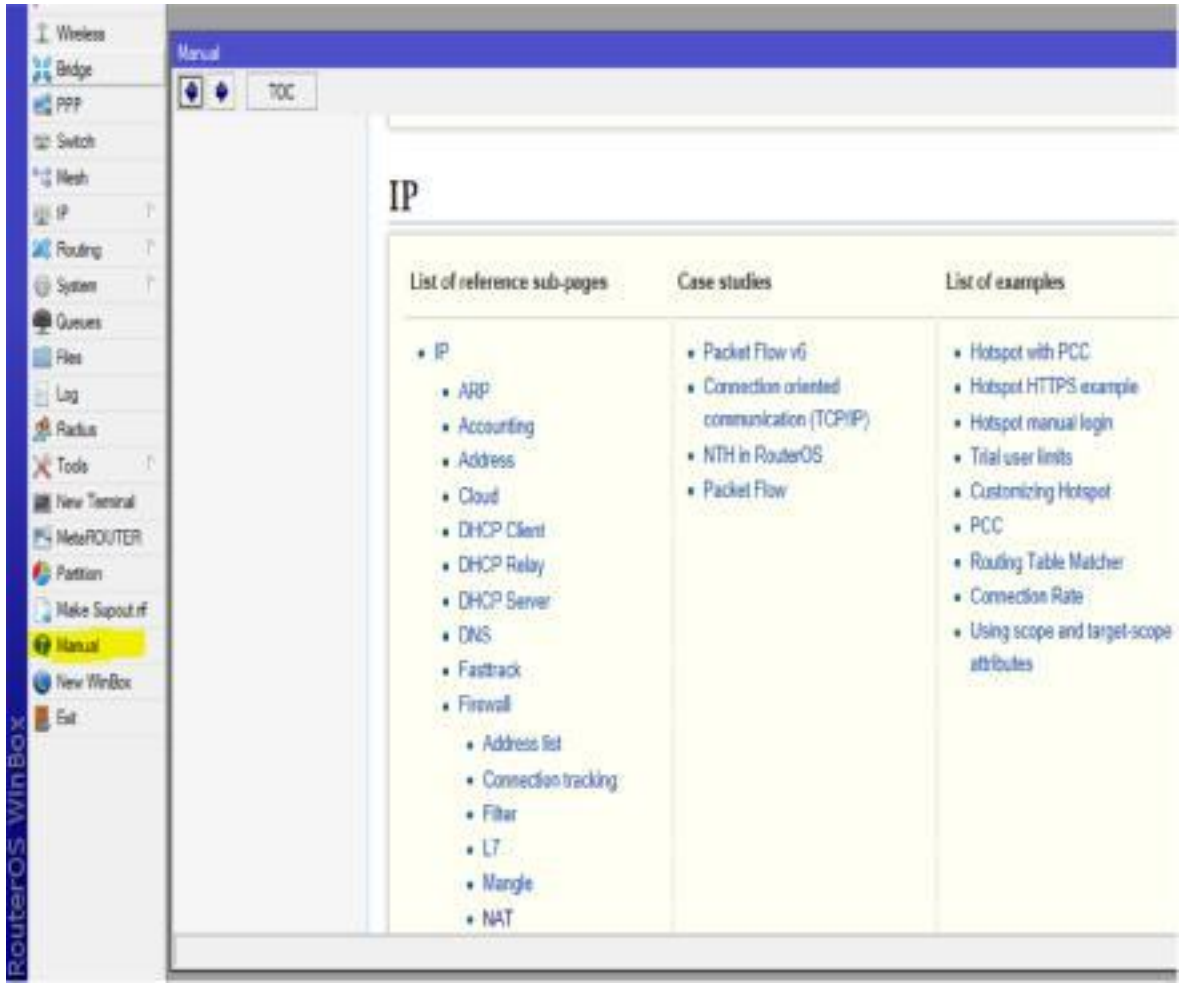
References

For More:
https://wiki.mikrotik.com/wiki/Manual:TOC

https://mum.mikrotik.com/

https://forum.mikrotik.com/

You can also browsing wiki.mikrotik.com from your RouterBoard.




Q
&
A