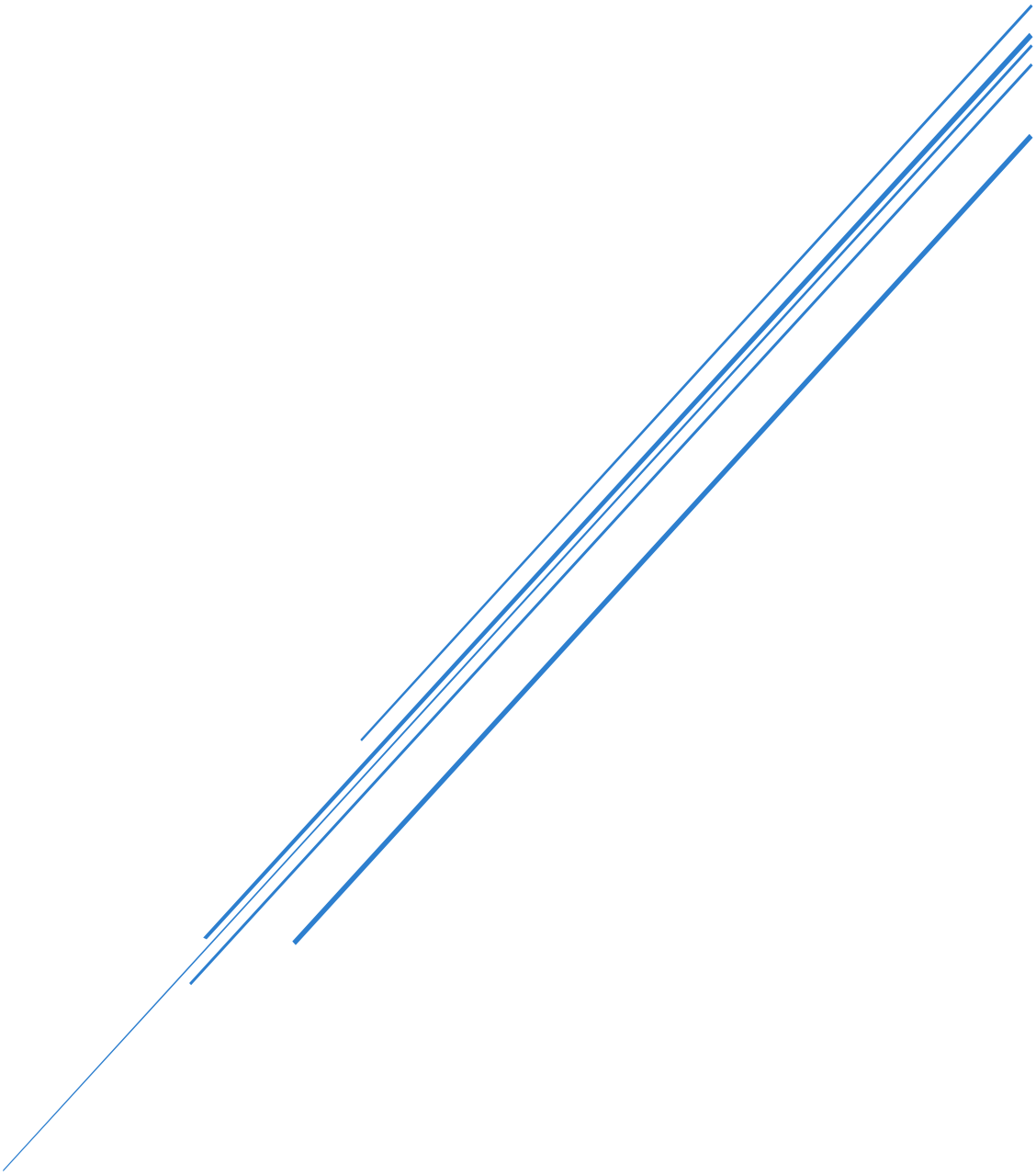
LOAD BALANCING CONFIGURATION

MikroTik



SUBASH SUBEDI
INFO@SUBASHSUBEDI0.COM.NP

Abstract

This project focuses on designing and implementing a dual-WAN load balancing system using MikroTik RouterOS 7.x with the Per Connection Classifier (PCC) technique. The complete network topology-consisting of two ISP connections, a MikroTik router, and multiple LAN clients-was built and simulated using GNS3 with real MikroTik CHR images. The purpose of this setup is to distribute traffic evenly across two WAN links, increase overall bandwidth efficiency, and provide automatic failover in case of link failure. All configurations were performed, tested, and validated within the GNS3 environment, ensuring the solution is accurate, repeatable, and suitable for real-world deployment.

Table of Contents

1. Objective

Implement reliable dual-WAN load balancing with automatic failover using the PCC (Per Connection Classifier) method. This configuration provides:

Approximately 50/50 traffic distribution (or weighed if required)

Full bandwidth utilization of both ISP links

Connection persistence (same connection always uses the same WAN)

Automatic failover and failback

2. Network Topology

Interface	Role	IP Assignment	Example IP	Gateway
Ether1	WAN1	ISP 1	192.168.1.89/24	192.168.1.254
Ether2	WAN2	ISP 2	192.168.174.133/24	192.168.174.2
Bridge1	LAN	Static LAN	192.168.88.1/24	192.168.88.1/24

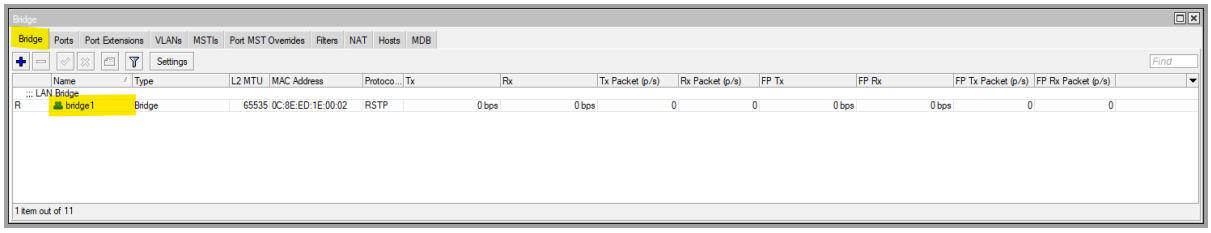
3. Configuration Steps

3.1. Create LAN Bridge

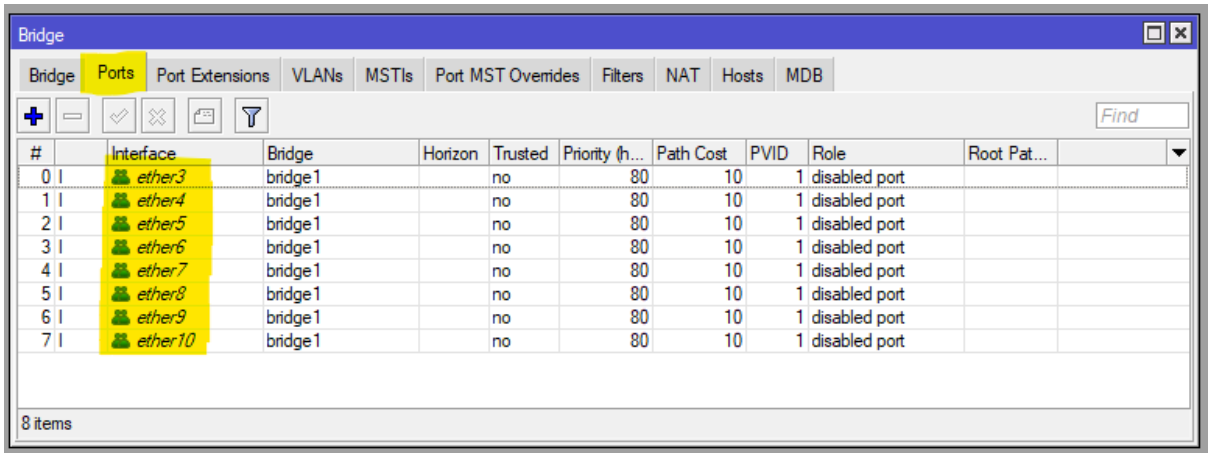
We create bridge1 and add ether3–ether10 as bridge ports. This makes all LAN interfaces behave as one unified switch, simplifying management and ensuring all local devices share the same network.

```
/interface bridge
add name=bridge1 comment="LAN Bridge"

/interface bridge port
add bridge=bridge1 interface=ether3
add bridge=bridge1 interface=ether4
add bridge=bridge1 interface=ether5
add bridge=bridge1 interface=ether6
```

```
add bridge=bridge1 interface=ether7
add bridge=bridge1 interface=ether8
add bridge=bridge1 interface=ether9
add bridge=bridge1 interface=ether10
```




```
[admin@MikroTik] > /interface bridge add name=bridge1 comment="LAN Bridge"
[admin@MikroTik] >
```



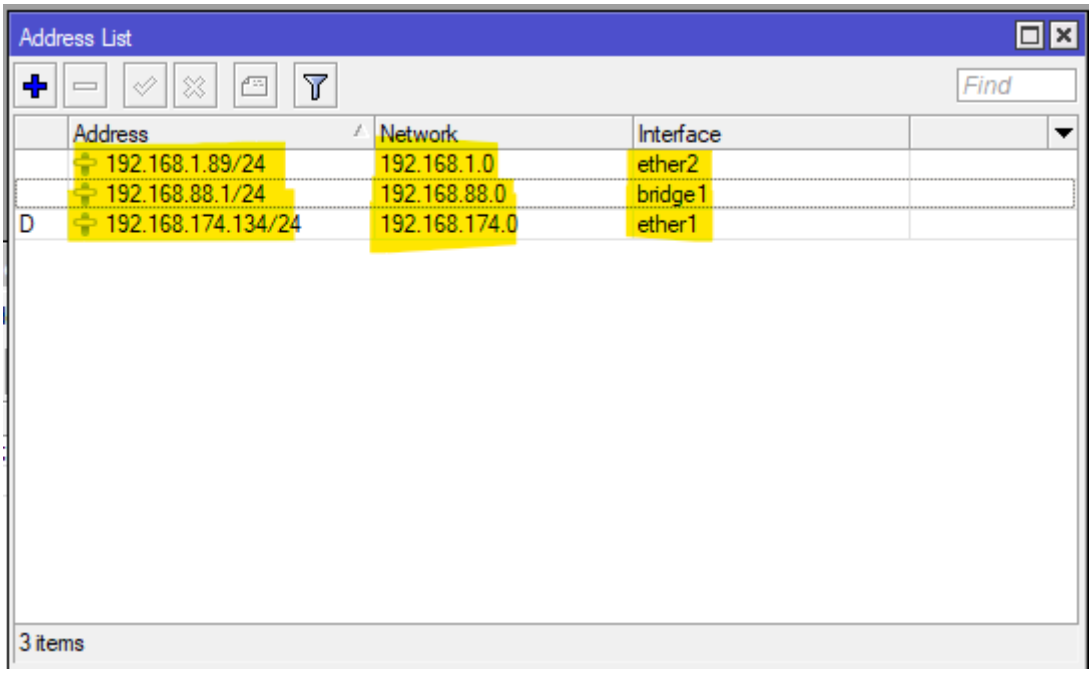

```
[admin@MikroTik] > /interface bridge port
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether3
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether4
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether5
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether6
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether7
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether8
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether9
[admin@MikroTik] /interface/bridge/port> add bridge=bridge1 interface=ether10
[admin@MikroTik] /interface/bridge/port> /
[admin@MikroTik] >
```

3.2. Assign LAN & WAN IP Address

We assign the bridge a fixed IP address (192.168.88.1/24). This becomes the default gateway for all LAN devices and the management IP for the router.

```
/ip address
add address=192.168.88.1/24 interface=bridge1 network=192.168.88.0
add address=192.168.174.134/24 interface=ether1 network=192.168.174.0
add address=192.168.1.89/24 interface=ether2 network=192.168.1.0
```

```
[admin@MikroTik] >
[admin@MikroTik] > /ip address add address=192.168.1.89/24 interface=ether2 network=192.168.1.0
[admin@MikroTik] > /ip address add address=192.168.174.134/24 interface=ether1 network=192.168.174.0
[admin@MikroTik] > /ip address add address=192.168.88.1/24 interface=bridge1 network=192.168.88.0
[admin@MikroTik] >
[admin@MikroTik] >
```



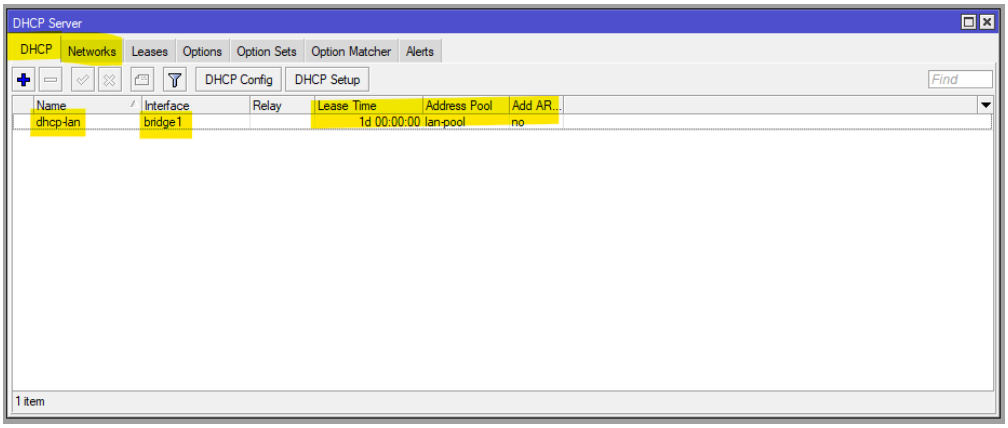
3.3. Configure DHCP Server for LAN

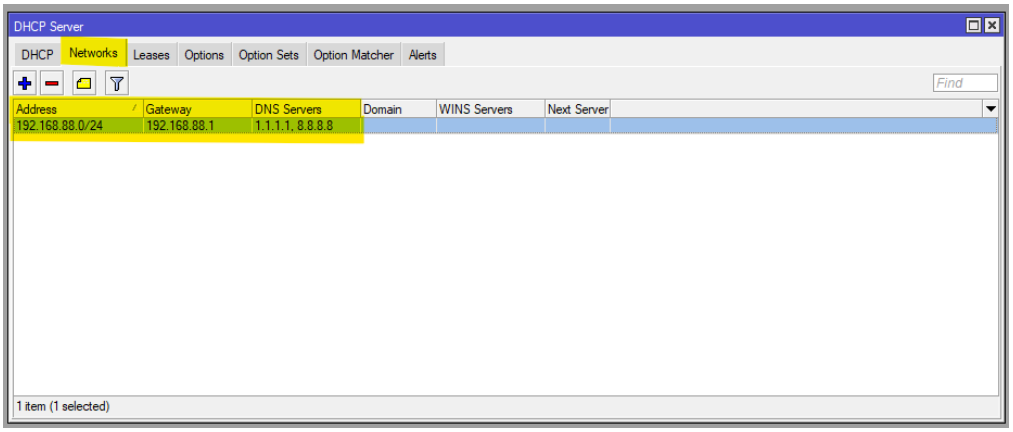
The DHCP server automatically provides IP addresses, gateway, and DNS information to client devices. This allows laptops, desktops, and mobile devices to connect without manual configuration.

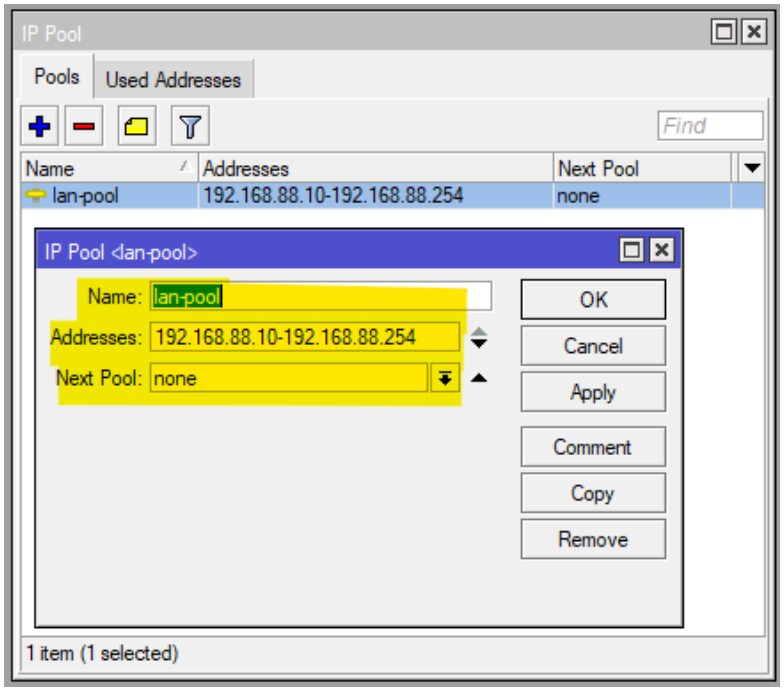
```
/ip pool
add name=lan-pool ranges=192.168.88.10-192.168.88.254

/ip dhcp-server
add name=dhcp-lan interface=bridge1 address-pool=lan-pool lease-time=1d

/ip dhcp-server network
```

add address=192.168.88.0/24 gateway=192.168.88.1 dns-server=1.1.1.1,8.8.8.8

```
[admin@MikroTik] >
[admin@MikroTik] > /ip pool add name=lan-pool ranges=192.168.88.10-192.168.88.254
[admin@MikroTik] >
[admin@MikroTik] > /ip dhcp-server add name=dhcp-lan interface=bridge1 address-pool=lan-pool lease-time=1d
[admin@MikroTik] >
[admin@MikroTik] > /ip dhcp-server network add address=192.168.88.0/24 gateway=192.168.88.1 dns-server=1.1.1.1,8.8.8.8
[admin@MikroTik] >
[admin@MikroTik] >
```





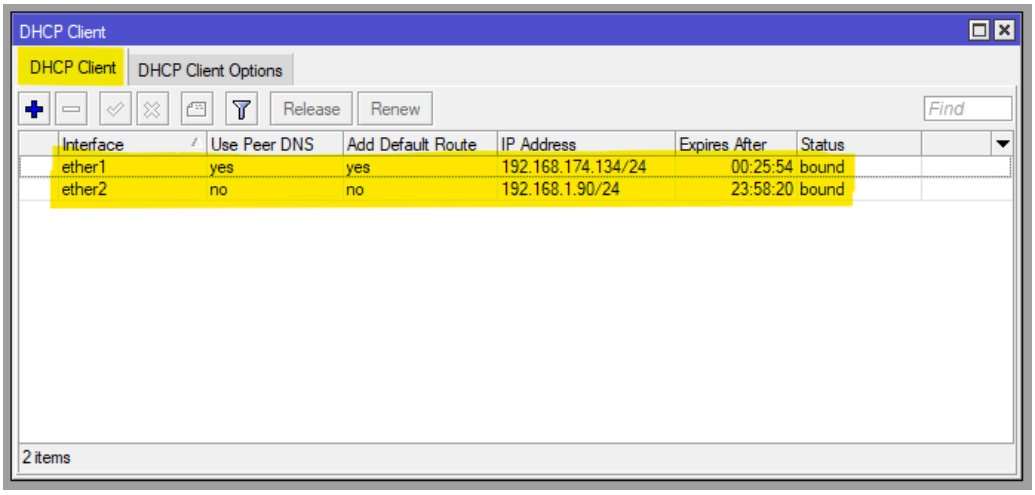


3.4. Configure WAN DHCP Clients (No Auto Routes)

Each WAN interface receives an IP address and gateway from its ISP using DHCP. We disable automatic default routes because PCC will control routing manually to achieve proper load balancing.

```
/ip dhcp-client
add interface=ether1 add-default-route=no use-peer-dns=no use-peer-ntp=no disabled=no
add interface=ether2 add-default-route=no use-peer-dns=no use-peer-ntp=no disabled=no
```

```
[admin@MikroTik] >
[admin@MikroTik] > /ip dhcp-client add interface=ether1 add-default-route=no use-peer-dns=no use-peer-ntp=no disabled=no
failure: dhcp-client on that interface already exists
[admin@MikroTik] >
[admin@MikroTik] > /ip dhcp-client add interface=ether2 add-default-route=no use-peer-dns=no use-peer-ntp=no disabled=no
[admin@MikroTik] >
[admin@MikroTik] >
```

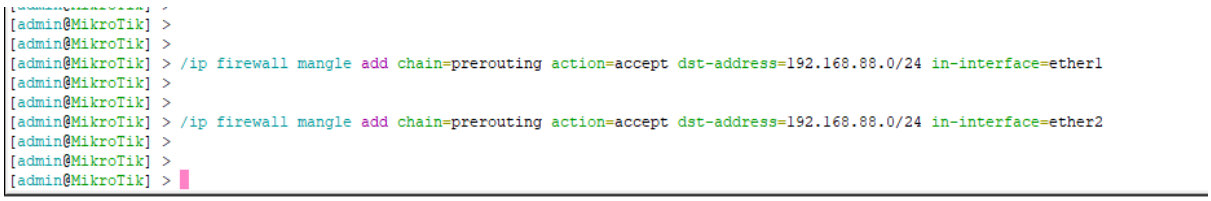


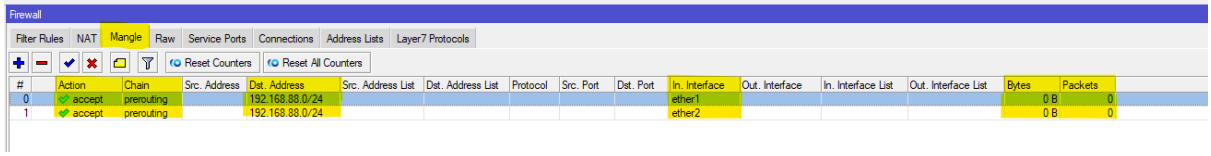
3.5. PCC Mangle Rules

Two mangle rules classify new connections and divide them equally between ISP1 and ISP2. PCC uses source/destination addresses and ports to evenly distribute traffic while keeping each session on one ISP.

3.5.1. Accept Traffic Returning from WAN

```
/ip firewall mangle add chain=prerouting action=accept dst-address=192.168.88.0/24 in-interface=ether1
/ip firewall mangle add chain=prerouting action=accept dst-address=192.168.88.0/24 in-interface=ether2
```

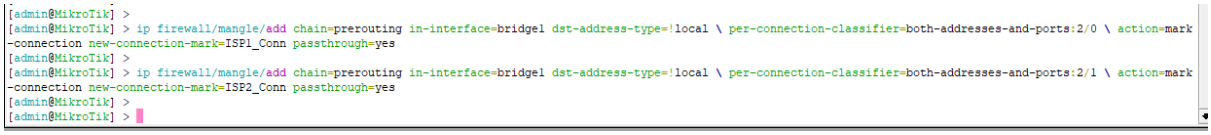


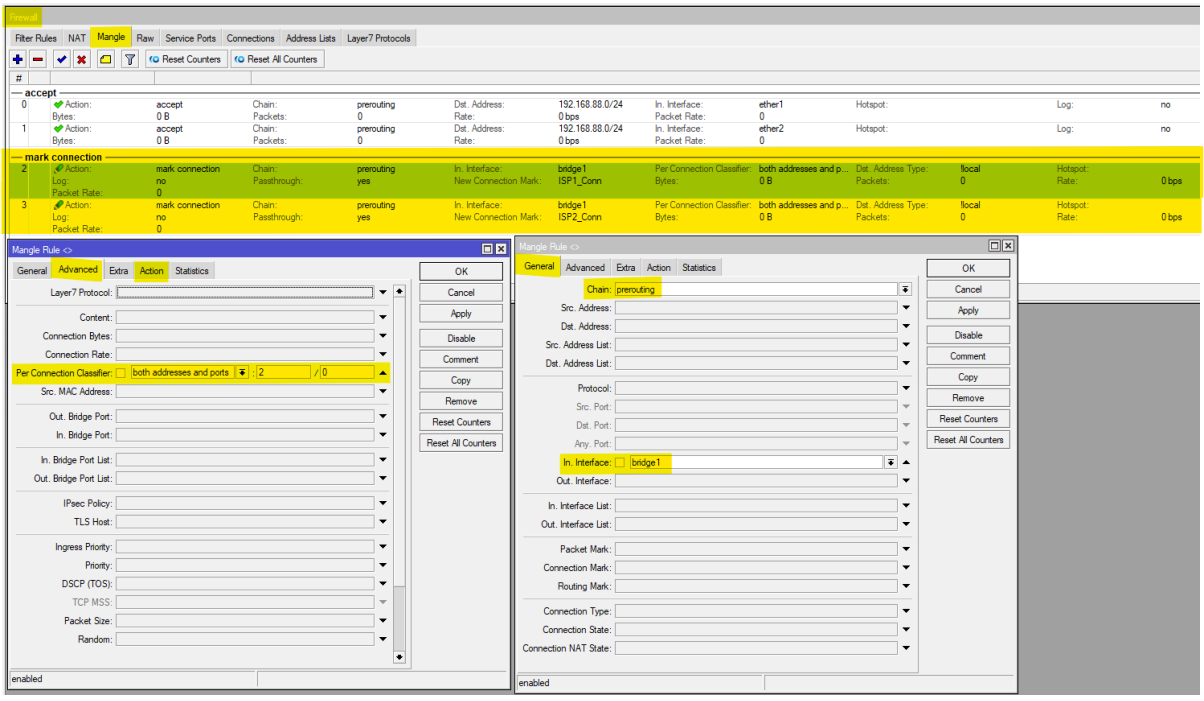


3.5.2. PCC Connection Marking

Two mangle rules classify new connections and divide them equally between ISP1 and ISP2. PCC uses source/destination addresses and ports to evenly distribute traffic while keeping each session on one ISP.

```
/ip firewall mangle add chain=prerouting in-interface=bridge1 dst-address-type=!local \ per-connection-classifier=both-addresses-and-ports:2/0 \ action=mark-connection new-connection-mark=ISP1_conn passthrough=yes

/ip firewall mangle add chain=prerouting in-interface=bridge1 dst-address-type=!local \ per-connection-classifier=both-addresses-and-ports:2/1 \ action=mark-connection new-connection-mark=ISP2_conn passthrough=yes
```



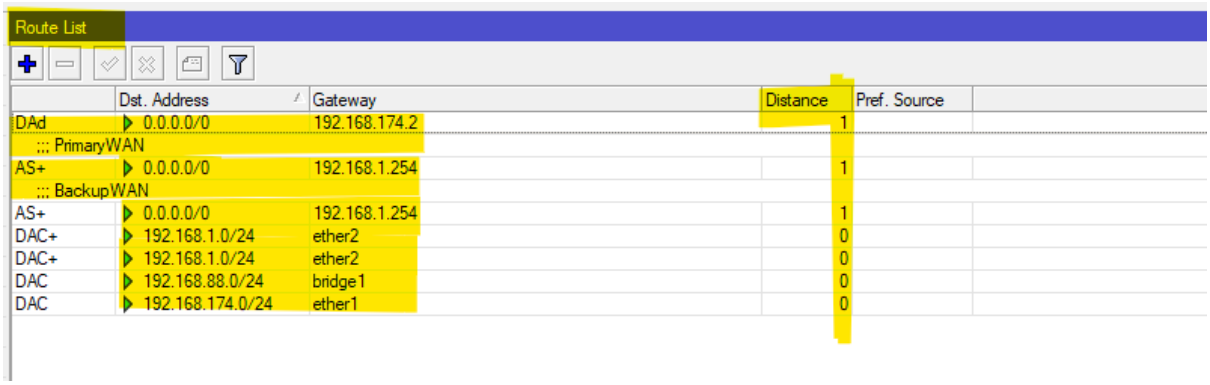


3.6. Routing Table

Routing marks are added to match the connection marks. This tells the router which ISP a specific connection should use and ensures consistent path selection. We add separate default routes for marked traffic, each pointing to its respective ISP gateway. Using check-gateway=ping allows automatic detection of link failure and ensures marked traffic avoids a down ISP. Two normal default routes are added with different distances (1 and 2). These routes are used by the router's own services-such as DNS, updates, and WinBox-and also act as failover paths if one ISP becomes unreachable.

```
/ip route
add gateway=192.168.1.254   routing-mark=to_ISP1 check-gateway=ping distance=1
add gateway=192.168.174.2   routing-mark=to_ISP2 check-gateway=ping distance=1

add gateway=192.168.1.254 distance=1 check-gateway=ping comment="Primary WAN"
add gateway=192.168.174.2 distance=2 check-gateway=ping comment="Backup WAN"
```

```
[admin@MikroTik] > ip route add gateway=192.168.1.254 distance=1 check-gateway=ping comment="PrimaryWAN"
[admin@MikroTik] >
[admin@MikroTik] > ip route add gateway=192.168.1.254 distance=1 check-gateway=ping comment="BackupWAN"
[admin@MikroTik] >
[admin@MikroTik] > /ip route print
Flags: D - DYNAMIC; A - ACTIVE; c, s, d, y - COPY; + - ECMP
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#      DST-ADDRESS       GATEWAY        DISTANCE
;;; BackupWAN
0  As+ 0.0.0.0/0         192.168.1.254         1
;;; PrimaryWAN
1  As+ 0.0.0.0/0         192.168.1.254         1
  DAd+ 0.0.0.0/0         192.168.174.2         1
  DAc+ 192.168.1.0/24    ether2                0
  DAc+ 192.168.1.0/24    ether2                0
  DAc  192.168.88.0/24   bridge1               0
  DAc  192.168.174.0/24  ether1                0
[admin@MikroTik] >
```

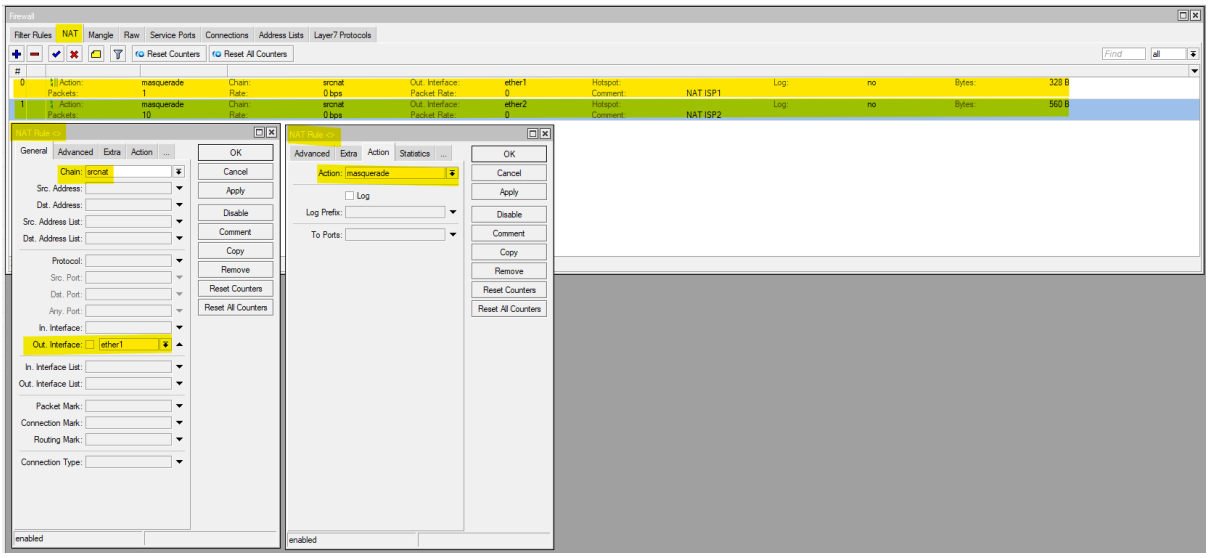


3.7. NAT (Masquerade)

Masquerade translates private LAN IP addresses to the ISP's public IP on each WAN. This ensures return packets are routed back correctly and allows all users to access the internet.

```
/ip firewall nat
add chain=srcnat out-interface=ether1 action=masquerade comment="NAT ISP1"
add chain=srcnat out-interface=ether2 action=masquerade comment="NAT ISP2"
```

```
[admin@MikroTik] >
[admin@MikroTik] > ip firewall nat add chain=srcnat out-interface=ether1 action=masquerade comment="NAT ISP1"
[admin@MikroTik] >
[admin@MikroTik] > ip firewall nat add chain=srcnat out-interface=ether2 action=masquerade comment="NAT ISP2"
[admin@MikroTik] >
[admin@MikroTik] > ip firewall nat print
Flags: X - disabled, I - invalid; D - dynamic
 0    ;;; NAT ISP1
      chain=srcnat action=masquerade out-interface=ether1

 1    ;;; NAT ISP2
      chain=srcnat action=masquerade out-interface=ether2
[admin@MikroTik] >
```

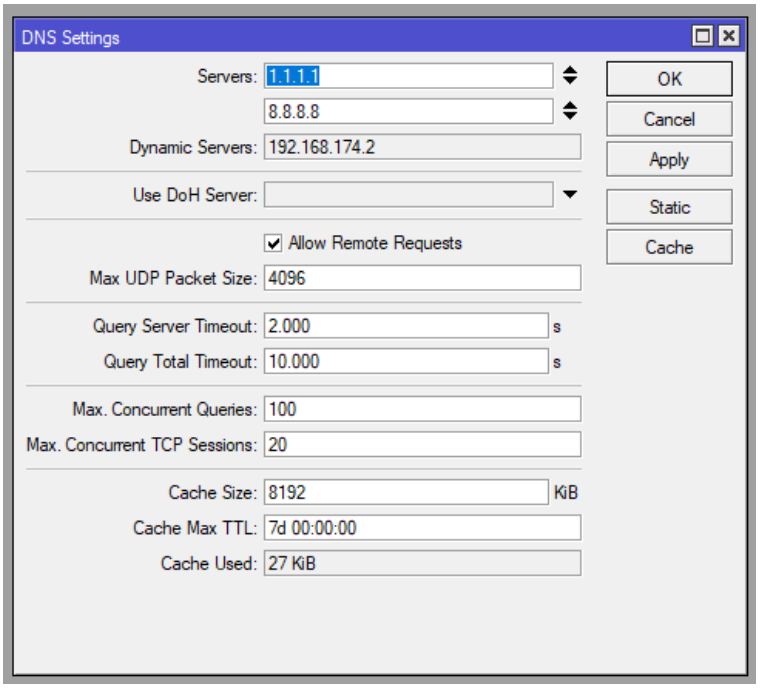


3.8. DNS

We set reliable public DNS servers (such as 1.1.1.1 and 8.8.8.8) and enable "allow-remote-requests" so the router can answer DNS queries for LAN clients, improving speed and reliability.

```
/ip dns
set allow-remote-requests=yes servers=1.1.1.1,8.8.8.8 cache-size=8192
```

```
[admin@MikroTik] >
[admin@MikroTik] > ip dns set allow-remote-requests=yes servers=1.1.1.1,8.8.8.8 cache-size=8192
[admin@MikroTik] >
[admin@MikroTik] > ip dns print
                    servers: 1.1.1.1,8.8.8.8
            dynamic-servers: 192.168.174.2
             use-doh-server:
            verify-doh-cert: no
  doh-max-server-connections: 5
  doh-max-concurrent-queries: 50
               doh-timeout: 5s
      allow-remote-requests: yes
        max-udp-packet-size: 4096
       query-server-timeout: 2s
        query-total-timeout: 10s
     max-concurrent-queries: 100
max-concurrent-tcp-sessions: 20
                cache-size: 8192KiB
             cache-max-ttl: 1w
               cache-used: 28KiB
[admin@MikroTik] >
```



4. Verification Commands

/ip firewall mangle print stats	# Should show balanced traffic
/ip route print where routing-mark~"to_"	# Confirm routing entries
/ping 8.8.8.8 interface=ether1	# Test ISP1 link
/ping 8.8.8.8 interface=ether2	# Test ISP2 link
/tool torch interface=ether1	# Live traffic usage
/tool torch interface=ether2	

5. Optional: Weighted Load Balancing (Example: ISP1 = 100 Mbps, ISP2 = 50 Mbps)

per-connection-classifier=both-addresses-and-ports:4/0 --- > ISP1_conn
per-connection-classifier=both-addresses-and-ports:4/1 --- > ISP1_conn
per-connection-classifier=both-addresses-and-ports:4/2 --- > ISP2_conn
per-connection-classifier=both-addresses-and-ports:4/3 --- > ISP2_conn

6. Conclusion

The implementation of MikroTik dual-WAN PCC load balancing in the GNS3 simulator was successful and met all expected objectives. The system provided stable 50/50 traffic distribution, maintained session persistence, and demonstrated reliable automatic failover and failback when ISP connections were intentionally disrupted. Using GNS3 allowed for safe experimentation, detailed analysis, and real-time troubleshooting without requiring physical hardware. The results confirm that the configuration developed in the simulator can be deployed directly on actual MikroTik routers with full compatibility. Overall, this project validates PCC as an effective and economical method for enhancing network performance and redundancy using multiple Internet connections.