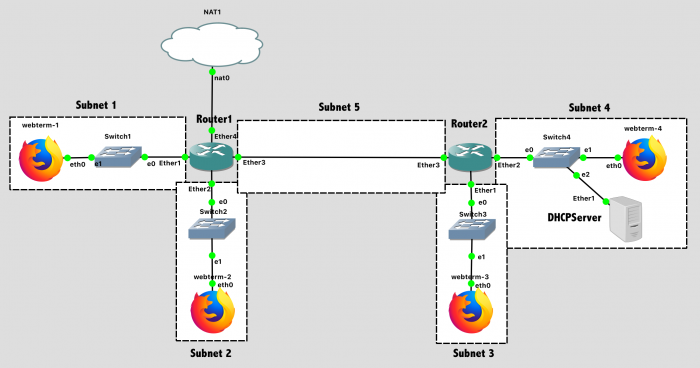
Lab 5/2. IPv4 & IPv6 configuration

In this lesson, you'll configure a network to support both IPv4 and IPv6.

Design Network

Your network topology should match this design. [](https://cyberlab.pacific.edu/user/pages/03.courses/02.comp177/labs/lab-11-ipv6/network-11.png)  
Lab 11 Network Diagram *(Note: Subnet labels and dashed borders are for informational use only)*

**However, BEFORE you even think about opening GNS3,** you should first complete the design of the network by filling in the following tables.

**IPv4 Network Design**

You are free to choose subnet addresses as you see fit, with the following restrictions:

1. All subnets must be in [**private IP address ranges**](https://en.wikipedia.org/wiki/Private_network) - It's a very bad practice to overlap with legitimate public IPs
2. All subnets must be /18 networks to accommodate large numbers of hosts, with the exception of subnet 5, which should be a /30
3. You should follow convention and assign the default gateway IPs to be either the lowest usable host address or highest usable host address in the subnet.

Tip: There are a variety of [**subnet calculators**](http://www.tunnelsup.com/subnet-calculator) available online. As with any calculator, it's important to know how to use it!

Choose IPv4 subnets for your network:

| **Subnet #** | **Subnet Address a.b.c.d/n** | **Number of IPs in Subnet** | **Lowest *Usable* Host Address** | **Highest *Usable***  **Host Address** |
| --- | --- | --- | --- | --- |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |

Notes:

* Your **Subnet** address should have all zero's for the host ID bits. For example, 192.168.10.0/24 (the lower 8 bits are all zero)
* Your **Number of IPs in Subnet** should include the first and last address (with a host ID of all 0's and all 1's, respectively)

Chose IPv4 addresses for all router interfaces that fit within the subnets you chose above:

| **Subnet** | **Host/Interface** | **IPv4 Address (include /n)** |
| --- | --- | --- |
| 1 | Router1, Ether1 Interface |  |
| 2 | Router1, Ether2 Interface |  |
|  |  |  |  |
| 3 | Router2, Ether1 Interface |  |  |
| 4 | Router2, Ether2 Interface |  |  |
|  |  |  |  |
| 5 | Router1, Ether3 Interface |  |  |
| 5 | Router2, Ether3 Interface |  |  |
|  |  |  |  |
| N /A | Router1, Ether4 Interface | *(Varies, use DHCP from the NAT)* |  |

You do not need to pick IPv4 addresses for the WebTerms - they will use DHCP to auto-configure their own IPv4 addresses.

**IPv6 Network Design**

An IPv6 address is represented as eight groups of four hexadecimal digits where each group represents 16 bits. Each group is separated by a colon (:). An example IPv6 address is: 2001:0db8:85a3:0000:0000:8a2e:0370:7334. To shorten addresses for display, leading zeros in each group can be omitted, and the longest sequence of consecutive all-zero groups can be replaced with two colons (::).

A standard unicast IPv6 address is composed of two parts: a 64-bit prefix used for routing, and a 64-bit interface identifier used to identify hosts.

| **Routing Prefix (64 bits)** | **Interface ID (64 bits)** |
| --- | --- |
| xxxx:xxxx:xxxx:xxxx | xxxx:xxxx:xxxx:xxxx |

Choose IPv6 subnets for your network. In this lab, the Internet Service Provider has assigned *Tiger Enterprises* the prefix 2001:0:F872::/48. (Why 2001? It's part of the [IPv6 Global Unicast Address Assignments from IANA](https://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml)). That 48-bit prefix leaves 16 bits available to specify subnets *within* the corporate network and a further 64 bits to identify hosts within each subnet. Pick a unique 16 bit subnet ID for subnets 1-5.

| **Subnet #** | **Routing Prefix (48 bits)** | **Subnet ID (16 bits)** | **Interface ID (64 bits)** |
| --- | --- | --- | --- |
| *Example* | 2001:0:F872 | xxxx | xxxx:xxxx:xxxx:xxxx |
| 1 | 2001:0:F872 |  | (Varies) |
| 2 | 2001:0:F872 |  | (Varies) |
| 3 | 2001:0:F872 |  | (Varies) |
| 4 | 2001:0:F872 |  | (Varies) |
| 5 | 2001:0:F872 |  | (Varies) |

Based on your chosen IPv6 subnets, next choose IPv6 addresses for all the router interfaces. These addresses should be of the form xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/64, indicating that the upper 64 bits represents the network prefix, and the lower 64 bits represents a host within the network. As in IPv4, it is considered a good convention that the router IPs be either the lowest usable host address or highest usable host address in the subnet.

| **Subnet** | **Host/Interface** | **IPv6 Address (include /n)** |
| --- | --- | --- |
| 1 | Router1, Ether1 Interface |  |
| 2 | Router1, Ether2 Interface |  |
|  |  |  |
| 3 | Router2, Ether1 Interface |  |
| 4 | Router2, Ether2 Interface |  |
|  |  |  |
| 5 | Router1, Ether3 Interface |  |
| 5 | Router2, Ether3 Interface |  |
|  |  |  |
| N/A | Router1, Ether4 Interface | *(No address - NAT not configured for IPv6)* |

You do not need to pick IPv6 addresses for the WebTerms - they will use network discovery to auto-configure their own IPv6 addresses.

Configure Network

Pre-Configuration Steps

1. Configure the **hostnames** of the routers in GNS3 to prevent confusion (via the GUI).
2. Configure the **hostnames** of the routers in the router itself to prevent confusion (via the CLI).
3. Disable the **DHCP Client** on each router. (Unless you ran Wireshark on a link, you wouldn't have previously noticed that a new MikroTik router defaults to running a DHCP client on Ether1, trying to auto-configure that network port so a new administrator can access the router over the network. It's not helpful for us here.) ip dhcp-client print followed by ip dhcp-client remove numbers=0 will eliminate this.
4. Configure **DNS** for Router1 and Router2 and DHCPServer using the Google Public DNS IP addresses: ip dns set servers=8.8.8.8,8.8.4.4 followed by ip dns print to confirm your setup.

IPv4 Configuration

1. Configure **IPv4 addresses** on all router interfaces that are connected to subnets.
2. Enable the **DHCP Client** on Router1 on the port connected to the NAT.
3. Enable **NAT** (IP Masquerading) on Router1 so that (after routing is enabled) the entire network can access the public Internet

Verification tests:

1. Can Router1 ping Router2? (only the interfaces on Subnet 5)
2. Can Router1 ping 8.8.8.8 and google.com?

OSPF Configuration

1. Configure **OSPF** on Subnets 1-5.
2. Configure OSPF to **distribute** (advertise) the **default route** present on Router1 across the network. (This default route was obtained via the DHCP client from the NAT, and allows Internet access)

Verification tests:

* Can Router2 ping 8.8.8.8 and google.com?
* Provide a **screenshot** showing the output of routing ospf route print from Router2. Routes to Subnets 1-5 should be visible, as well as a default route via Router1 (towards the NAT)

DHCP Configuration

1. Configure DHCPServer with a static IP address and a default route
   1. Test: Before continuing, ensure that DHCPServer can ping Router1, 8.8.8.8, and google.com. Until it's on the network as a functional host, there's no point in configuring it as a DHCP server.
2. Configure DHCPServer with the necessary settings to function as a DHCP server for Subnets 1-4. Note: Subnet 4 does not need a DHCP relay - the server is within that subnet. Subnet 5 does not use DHCP, because it only has two devices - the routers.
3. Configure **DHCP relays** on Router1 and Router2 for Subnets 1-3.
4. Configure the Webterms to use DHCP

* Provide a **screenshot** of ip dhcp-server lease print on DHCPServer showing the four expected leases.
* Provide a **screenshot** showing webterm-1 pinging both webterm-4 and google.com

IPv6 Configuration

To configure a Mikrotik router for IPv6:

* Enable the IPv6 package, and reboot the router: system package enable ipv6 followed by system reboot. (If you use safe mode regularly, make sure you've exited safe mode with [CTRL]+[X] so you don't lose all your work!)
* Configure an IPv6 interface: ipv6 address add address=xxxx:xxxx:xxxx:.......:xxxx/n interface=etherX
* Verify interface configuration: ipv6 address print (You will see link-local addresses of the form fe80:xxxx - those are automatically configured and can be ignored)
* Neighbor discovery is running automatically: ipv6 nd print. The result of this is that the router broadcast "Router Advertisement" packets out all interfaces periodically, and hosts wishing to join the network use that router as their default gateway and generate IPv6 addresses based on the advertised prefix. This process is called [Stateless Address Autoconfiguration (SLAAC)](https://en.wikipedia.org/wiki/IPv6_address#Stateless_address_autoconfiguration). You can see the specific advertised prefixes via ipv6 nd prefix print. (Tip: If the prefixes show as "Invalid" and you're confident your IPv6 addresses on each interface are valid, try rebooting your router)

To examine IPv6 configuration on the webterms (which are generic Linux devices):

* View IP address: ip addr (IPv4 is the inet lines, IPv6 is the inet6 lines)
  + View *only* IPv4 addresses: ip -4 addr
  + View *only* IPv4 addresses: ip -6 addr
* View routing table:
  + IPv4 routing table: ip route
  + IPv6 routing table: ip -6 route
* Network ping:
  + IPv4 ping: ping
  + IPv6 ping: ping6
* Provide a **screenshot** showing webterm-1 successfully pinging webterm-2 over **IPv6** with the ping6 command.
* Provide a **screenshot** showing IPv6 pings captured in Wireshark. You can filter on icmpv6 to see all ICMPv6 traffic (pings and other network notifications), or filter on just icmpv6.type==128 || icmpv6.type==129 to see *just* the IPv6 pings, both echo response and echo reply.
* Provide a **screenshot** of a Wireshark capture of a "Router Advertisement" packet used to notify all IPv6-enabled hosts on a subnet of the attached router **and the router's prefix** (for use in network autoconfiguration). Use filter icmpv6. Expand the fields so that the advertised prefix (**your selected prefix** for the given subnet) is clearly shown.

IPv6 Routing

From webterm-1, try to ping webterm-4 over IPv6. What error do you get? *(Destination unreachable: No route)*

You need to set up IPv6 routing tables in the same manner that you set up IPv4 routing tables. For IPv6, OSPFv3 is used, whereas for IPv4, OSPFv2 was used. One key difference with OSPFv3 is that there is no configuration for networks anymore, and interface configuration becomes mandatory, since OSPFv3 runs on a link basis, not an IP subnet basis.

On each router:

* Add the desired links connected to IPv6 subnets (subnets 1-5): routing ospf-v3 interface add interface=etherX area=backbone
* Verify that the links are added: routing ospf-v3 interface print
* Verify that the routing table shows Subnets 1-5 in IPv6: routing ospf-v3 route print
* Verify that the routing table shows Subnets 1-5 in IPv6: ipv6 route print

Lab Submission

Submit the following items to the email message with 1st part to 9999924816@ya.ru assignment:

* The **design tables** (from above) documenting your subnet design and selected IP addresses for both IPv4 and IPv6
* Provide a **screenshot** showing your GNS3 topology
* Provide a **screenshot** showing the output of routing ospf route print from Router2. Routes to Subnets 1-5 should be visible, as well as a default route via Router1 (towards the NAT)
* Provide a **screenshot** of ip dhcp-server lease print on DHCPServer showing the four expected leases.
* Provide a **screenshot** showing webterm-1 pinging both webterm-4 and google.com
* Provide a **screenshot** showing webterm-1 successfully pinging webterm-2 over **IPv6** with the ping6 command.
* Provide a **screenshot** showing IPv6 pings captured in Wireshark. You can filter on icmpv6 to see all ICMPv6 traffic (pings and other network notifications), or filter on just icmpv6.type==128 || icmpv6.type==129 to see *just* the IPv6 pings, both echo response and echo reply.
* Provide a **screenshot** of a Wireshark capture of a "Router Advertisement" packet used to notify all IPv6-enabled hosts on a subnet of the attached router **and the router's prefix** (for use in network autoconfiguration). Use filter icmpv6. Expand the fields so that the advertised prefix (**your selected prefix** for the given subnet) is clearly shown.
* Provide a **screenshot** showing webterm-1 successfully pinging webterm-4 over **IPv6** with the ping6 command.
* Provide a **screenshot** of ipv6 route print on Router1 showing routes to IPv6 subnets 1-5

When finished, press the Stop button and exit GNS3. The GNS3 VM (in VMware) should halt and exit automatically.