

# Working with Addresses

## Subnetting

It's helpful to have this table on-hand (just memorize the column for /24 and figure out the rest by multiplying and dividing by 2):

| /21  | /22  | /23 | /24 | /25 | /26 | /27 | /28 | /29 | /30 |
|------|------|-----|-----|-----|-----|-----|-----|-----|-----|
| 2048 | 1024 | 512 | 256 | 128 | 64  | 32  | 16  | 8   | 4   |

The first row specifies the length of the prefix, the second is the number of addresses in the network for the selected prefix. Note that we need to subtract the number 2 to get the number of assignable addresses.

**Task:**

The base network address is 10.0.0.0/8. Divide this network into at least 5 subnets. We need at least 5 subnets, and 3 bits give us a count of 8 subnets, into which we can fit 5.

The subnet mask is: ..... The octet where the last "1" is present: .....

The magic number is: .....

| Subnet # | Subnet address | Host range | Broadcast |
|----------|----------------|------------|-----------|
| LAN1     | 10.0.0.0/      |            |           |
| LAN2     |                |            |           |
| LAN3     |                |            |           |
| LAN4     |                |            |           |
| LAN5     |                |            |           |
|          |                |            |           |

**Task:**

The base network address is 172.18.0.0/16. Create subnets so that each subnet contains at least 511 assignable addresses. Just create the first 5 subnets.

Bits for the new host portion: ..... Bits for the subnet portion: .....

New prefix length: ..... Number of subnets: .....

The subnet mask is: ..... The octet where the last "1" is present: .....

The magic number is: .....

| Subnet # | Subnet address | Host range | Broadcast |
|----------|----------------|------------|-----------|
| LAN1     | 172.18.0.0/    |            |           |
| LAN2     |                |            |           |
| LAN3     |                |            |           |
| LAN4     |                |            |           |
| LAN5     |                |            |           |

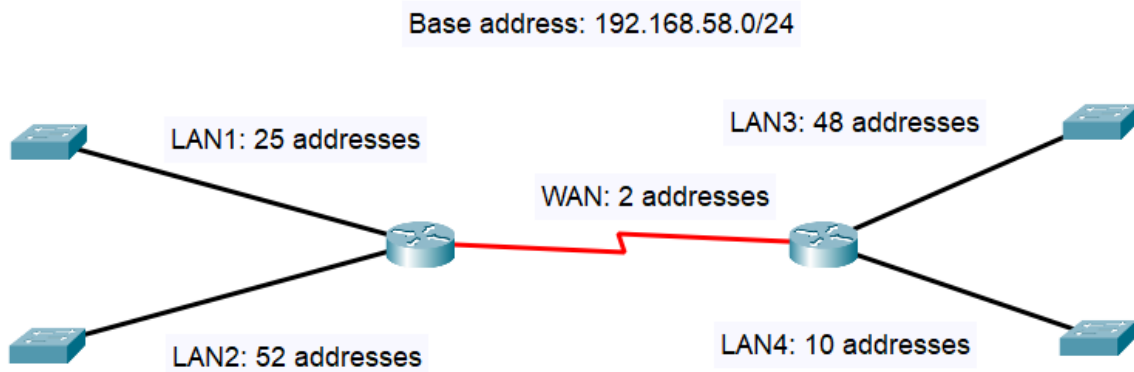
# VLSM

If we need differently sized subnets, we proceed similarly to subnetting, but we change the prefix length (number of bits for the subnet) on individual lines as needed.

In general, we proceed from the largest networks to the smallest, which can ensure unambiguity (individual networks must not overlap).

## Task:

Create subnets according to the following figure:



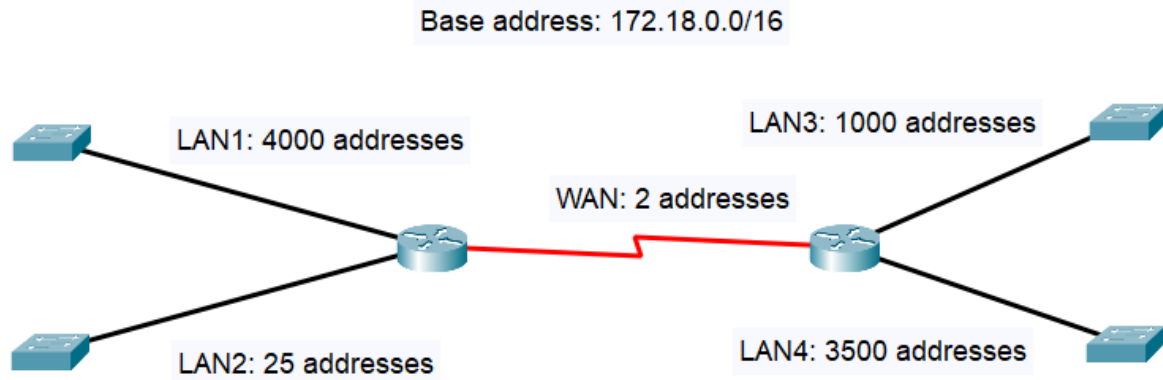
## Procedure:

- sort the subnets from largest to smallest,
- for each, determine the prefix length, mask and magic number,
- calculate the subnet addresses sequentially,
- fill in the remaining columns.

| Subnet #                | Subnet address | Host range | Broadcast | Mask, magic #                  |
|-------------------------|----------------|------------|-----------|--------------------------------|
| <b>LAN2</b><br>52 -> 64 | ...58.0/26     |            |           | ...255.192<br>1100 0000 -> +64 |
| <b>LAN3</b><br>48 -> 64 |                |            |           |                                |
|                         |                |            |           |                                |
|                         |                |            |           |                                |
|                         |                |            |           |                                |

**Task:**

Create subnets according to the following figure:



| Subnet #     | Subnet address | Host range | Broadcast | Mask, magic # |
|--------------|----------------|------------|-----------|---------------|
| LAN1<br>4096 | ...0.0/        |            |           |               |
| LAN4         |                |            |           |               |
|              |                |            |           |               |
|              |                |            |           |               |
|              |                |            |           |               |

## Routing

### Static Routing

In static routing, we specify the destination network, and then either the IP address of the neighbor (next hop router) through which the network is reachable, or the port leading to that neighbor.

```
ip route 192.168.10.0 255.255.255.0 g0/1
```

OR

```
ip route 192.168.10.0 255.255.255.0 10.30.0.1
```

To determine the gateway of the last resort, we do the following (expecting g0/1 to be the port leading to our ISP):

```
ip route 0.0.0.0 0.0.0.0 g0/1
```

We can also determine the backup default route (with slightly worse metrics than the default route). When the "original" default route is not operational, the backup route can be used:

```
ip route 0.0.0.0 0.0.0.0 g0/1  
ip route 0.0.0.0 0.0.0.0 g0/2 10
```

IPv6 routing: first we must turn up IPv6 unicast routing, then we can specify the IPv6 routes. Example:

```
ipv6 unicast routing  
ipv6 route ::0/0 g0/1
```

## RIPv2

This protocol is very simple to configure. We create the routing process, and then in the inner subconfiguration mode, we can make configuration.

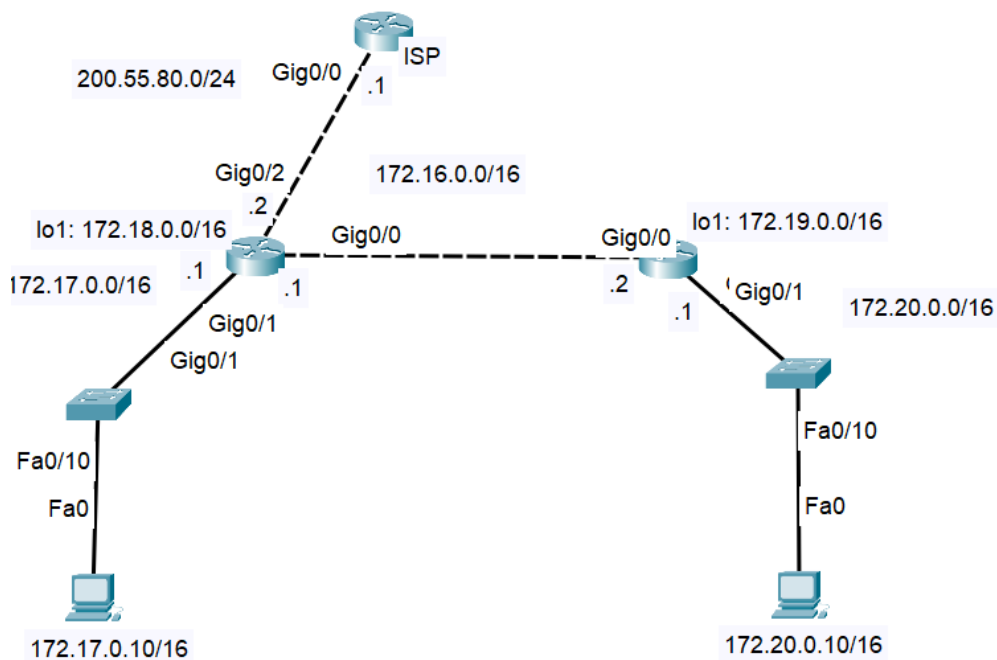
```
router rip
  version 2
  no auto-summary           ; networks will not be summarized in the routing table
  network 10.20.0.0         ; routing information is propagated here
  network 10.30.0.0
  passive-interface g0/0/0  ; no routing information is propagated here
  default-information originate ; the route of the last resort will be propagated
```

## show commands

```
sh ip route
sh ip route static
sh ip route connected
sh ipv6 route
sh ip route rip
debug ip rip
```

## Task:

Create the following network.



Configure all addresses on routers and end devices, including the loopbacks. Create the static route of the last resort on the first router leading to ISP (use the port number g0/2).

Configure RIPv2 on the both "non-ISP routers", propagate all connected networks. Turn off automatic summarization. The port leading to ISP will be passive interface. Propagate the gateway of the last resort.

Try ping between the end devices, try various show commands.