South America Workshop

WALC 2006 (Quito, Ecuador – 26-28 July '06)

Stateful Auto-configuration
(continued)

Miguel Baptista (miguel.baptista@fccn.pt)

Pedro Lorga (lorga@fccn.pt)

Simon Muyal (muyal@renater.pt)

Laboratory Exercise: *Stateful Auto-configuration (continued)*

Objectives

In this laboratory exercise you will complete the following tasks:

- *Minor DHCPv6 client configuration*
- *Experience the usage of both auto-configuration modes (stateless and stateful)*

Visual Objective

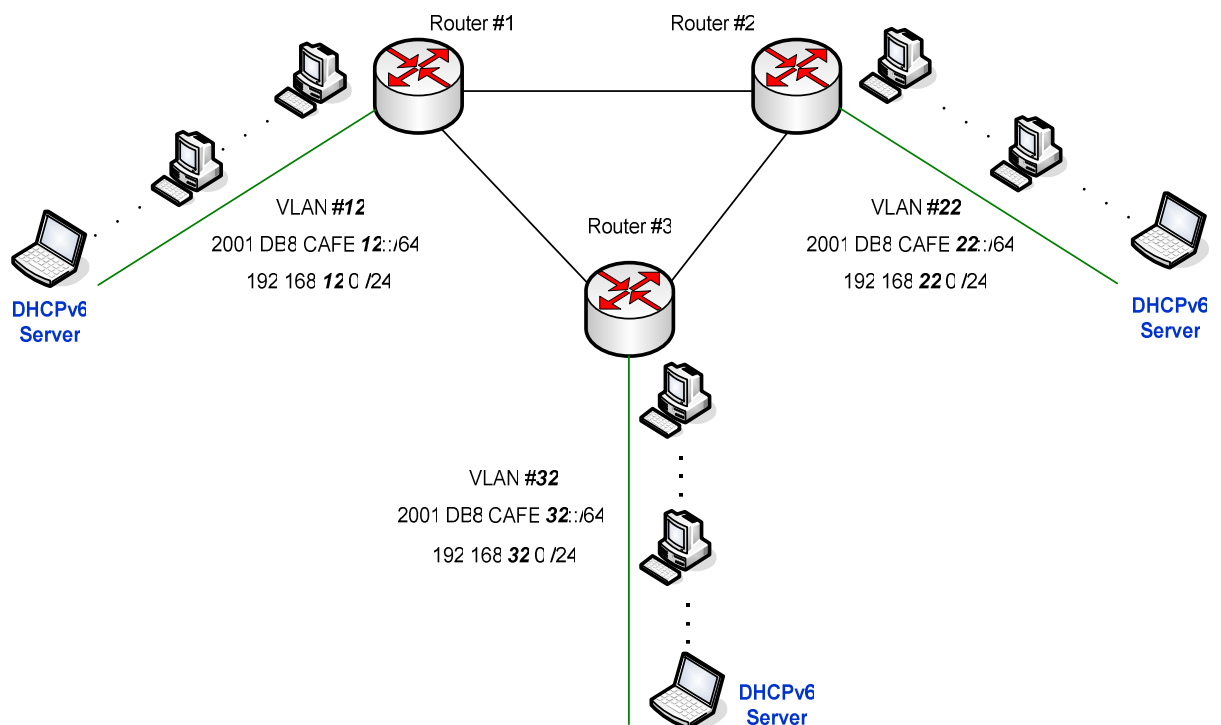The following figure shows the topology of the current laboratory.

Figure 1: Scenario topology

Setup

This time, all hosts connected to a specific router will be on the same VLAN.

Please, verify that your IPv6 address belongs to one of the following networks:

- *2001:DB8:CAFE:**12**::/64*
- *2001:DB8:CAFE:**22**::/64*
- *2001:DB8:CAFE:**32**::/64*

If you have manually configured an IPv6 address that doesn't belong to one of these networks please remove it.

On this scenario there is a DHCPv6 server (laptop) with three network interfaces. Each one is connected to a different VLAN. Because it will answer to each DHCPv6 request independently, we can consider that for each VLAN there will be a DHCPv6 server.

Scenario

On the previous scenario, you changed the values of two auto-configuration related parameters, on the router advertisement messages. But at the present those flags are ignored by Operating Systems. So, we have to run an external DHCPv6 client.

For both client and server we are going to use the Dibbler implementation. Currently there are ports available for Windows XP, Windows 2003 and Linux systems.

On the present setup the server will run on a Windows XP and the clients will run on linux.

The DHCPv6 client is configured via the **/etc/dibbler/client.conf** file. You already have a configuration file there. We are only going to do some basic configurations. Therefore, you can use this file as a starting point for your client configuration.
Please remove the unnecessary comments (character "#") in order to have a configuration file similar to this one:

```
log-mode short

# 7 = omit debug messages
log-level 7
```

```
iface eth0 {
  option dns-server
  option domain
  ia
}
```

If you want to know how to fully configure the client, please consult the Dibbler User's Guide (/usr/local/dibbler/doc/dibbler-user.pdf).
For the purpose of this scenario what's important for you to understand is that with this configuration, for interface eth0, the client will request the following parameters from the server:

- *List of dns-servers (keyword **option dns-server**)*
- *List of domains   (keyword **option domain**)*
- *One IPv6 address (keyword **ia**)*

Regarding the DHCPv6 server, it has 3 interfaces, each one connected to a different network. The server has also three different pools of addresses, each one associated with a specific interface. As you can observe on table 1:

Network	Interface	Pool
*2001:DB8:CAFE:**12**::/64*	*Eth0*	*2001:DB8:CAFE:**12**::1212:1 - 2001:DB8:CAFE:**12**::1212:FFFF*
*2001:DB8:CAFE:**22**::/64*	*Eth1*	*2001:DB8:CAFE:**22**::2222:1 - 2001:DB8:CAFE:**22**::2222:FFFF*
*2001:DB8:CAFE:**32**::/64*	*Eth2*	*2001:DB8:CAFE:**32**::3232:1 - 2001:DB8:CAFE:**32**::3232:FFFF*

Table 1: Addresses' pools per interface

The dns-server address and the domain name will be the same for all three interfaces/networks:

Network	Interface	DNS server	Domain
*2001:DB8:CAFE:**12**::/64*	*Eth0*		
*2001:DB8:CAFE:**22**::/64*	*Eth1*	*2001:DB8:CAFE:38::5*	*6diss.org*
*2001:DB8:CAFE:**32**::/64*	*Eth2*		

Table 2: DNS Server and Domain name per interface

Task 1: *Using Router Advertisement messages and DHCPv6*

Complete the following exercise's steps:

Step 1: Configure a DHCPv6 client
Enter your Linux operating system and configure the DHCPv6 client according to the instructions on the previous Setup/Scenario section.
Verify if your computer has more than one network interface. If so, make sure you configure the correct interface.

Step 2: Check your current global IPv6 address(es).

Step 3: With Ethereal, start capturing IPv6 packets and then run the DHCPv6 client.

You can run Dibbler client as a Daemon – detached from console and run in the background - but during the test phase it is better to watch its behaviour in real time. In this case use the `run` parameter. Dibbler will show its log messages in the console.
(**Tip**: `/usr/local/dibbler/dibbler-client run`)

Step 2: On the DHCPv6 client's running console, check the messages and verify if you received the requested data (IPv6 address, DNS servers and Domains).

Step 4: With Ethereal, analyze the request messages sent to the DHCPv6 server. What are the source and destination addresses? Then, examine the response messages from the server. Can you identify on these messages the response data from the server?

Step 5: Using Linux commands check if you received the data you requested.
(**Tip**: `ifconfig` and `cat /etc/resolver.conf`)

Check also your current IPv6 routing table (`route –A inet6 -n`). Do you have a default gateway?

Step 6: Kill your DHCPv6 client (Ctrl + C on the running console)

Step 7: Check that you don't have any address assigned by the server neither the information regarding the DNS and domain lists.

Task 2: *No Router Advertisement messages and DHCPv6*

On this task the Router Advertisement messages are disable. This means there are no RA messages on the network.

Step 1: Restart your network (**Tip**: `/etc/init.d/network restart`)

Step 2: Check if you have any IPv6 address.

Step 3: Run Dibbler client.

Step 4: Check if you have IPv6 addresses and the DNS server list.

Step 5: Now check your current IPv6 routing table (**Tip**: `route -A inet6 -n`).
Do you have a default gateway? Why?

Summary

After completing these exercises, you should be able to:
- *Do basic DHCPv6 client configuration*
- *Understand the usage of both auto-configuration modes (stateless and stateful)*