



Short Guide for the configuration of Linksys WRT54GS routers



Computer Networks and Services

2012-2013

1. Introduction

This document is a short guide that it is intended for helping with the configuration of the Linksys WRT54GS/GL router in the lab. Be aware that this guide is not a replacement for the Linksys WRT54GS/GL manual [1].

2. Linksys WRT54GS/GL Router

2.1 Introduction

The Linksys WRT54GS is a Linux-based SOHO (Small Office Home Office) router. It allows installing different versions of Linux with different applications and utilities, such as Open WRT (<http://openwrt.org/>). Additionally, it is a cheap router (it costs less than 70€).



2.2 Network Interfaces

The router has 6 network interfaces, which are configured with the following IP addresses:

- Ethernet interfaces:
 - eth0.0: 192.168.0.1/24
 - eth0.1: 192.168.1.1/24
 - eth0.2: 192.168.2.1/24
 - eth0.3: 192.168.3.1/24
 - eth0.4: 192.168.4.1/24
- Wireless interface (WLAN):
 - wlan0: 192.168.5.1/24



2.3 Ways to access the router

The router can be only accessed remotely through any of its interfaces and using the IP address corresponding to that interface. Logically, that implies connecting the host we are using to configure the router to one of its interfaces. Therefore, we must configure previously the IP address of the host to be able to effectively communicate the host and the router.

In particular, to connect one of the PCs in the lab to the router and gain access to it you need to perform the following steps:

- 1- Connect a physical cable from the eth1 card in the PC to one of the interfaces of the router, let's consider the eth0.0
- 2- Configure an IPv4 address to eth1 in the PC belonging to the same subnet as eth0.0. You can use the following command for this purpose:

```
$> ip addr add 192.168.0.X/24 dev eth1 (X can be any number between 2 and 254).
```

Note that interfaces can have multiple IP address, then you do not need to delete or modify any of the existing addresses in eth1.

Once the PC is connected to the router, there are 2 complementary ways to access the router:

- Via TELNET:
 - It provides access to a configuration console.
 - Through that console, it is possible to configure:
 - Interfaces (turn-on, turn-off)
 - IP (addresses, static routes)
 - Dynamic routing protocols (RIP, OSPF and BGP).
- Via SSH:
 - It provides access to a Linux console.
 - This console provides the basic commands found in a Linux system.
 - This way of accessing the router is used mainly to configure the wireless interface. The required access parameters are:

2.4 Configuration of the router

The router's default configuration is restored every time the router is turned on. Therefore, all changes previously done are lost if the router is reset or turned off. Additionally, erase the default IP addresses of the router that you are not going to use to avoid any problem in the lab session.

Sometimes the host (PC) used to configure the router is also an element of the lab scenario and the IP addressing required in the scenario is different from the IP addressing required to configure the router. In that case, configure first the PC interface to access to one of the router interfaces (e.g., eth0.0), as explained above. Modify the configuration of the router interfaces according to the lab scenario, but do not delete the default IP address of the interface that you are using to communicate with the router (e.g., eth0.0). Once you have finished the configuration of the router, modify the configuration of the PC to be able to access to one of the router interfaces that are already configured.

We will establish a TELNET connection to configure the router. Once the connection has been successfully established, we will have accessed the router's console in TERMINAL MODE (the *prompt* "router#" appears). In this mode, you cannot change the configuration of the router, only see it and use some basic commands (*ping*, *traceroute* and *telnet*). To configure the router, we need to access to the CONFIGURATION MODE. To do so, type the command "*configure terminal*" and the prompt "router(config) #" will appear. In this mode, we can use the online help typing ("?",) and the autocomplete function (using the tab). This is the typical interface of CISCO routers.

If you use that PC to configure several routes, you should take care with the ARP table of the PC. Specifically, when you connect the PC to the first router, the MAC and IP addresses of that interface of the router will be stored in the ARP table of the PC. If you connect the PC to the same interface of the next router (you should use the same interface to avoid configuring again the IP addresses of the PC), the IP address of the interface will be the same but not the MAC address, so the communication between the router and the PC will fail. To solve this problem, erase the corresponding entry of the ARP table of the PC before connecting the PC to the next router.

Basic Configuration Commands

Next you can find a list of basic configuration commands (although we recommend to use the manual to understand what different commands do).

```
> ? [To obtain the list of commands available in this moment]
> <comando> ? [Available options for the indicated command]
> show interface [Show the information about all router interfaces]
> show run [Show the current router configuration]
> show ip route [Show the IPv4 forwarding table]
> show ipv6 route [Show the IPv6 forwarding table]
> ping <IPv4 address> [ping to an IPv4 address]
> ping ipv6 < IPv6 address> [ping to an IPv6 address]
> ping ipv6 <dirección IPv6>%< interface> [ping to an IPv6 address using the indicated interface]
> conf t [Pass to configuration mode-> config]
config> ip route <prefix> <next hop_address> [Add an entry to the IPv4 forwarding table]
config> ip route <prefix> <interface> [Add a direct entry to the IPv4 forwarding table]
config> ipv6 route <IPv6_prefix> < next_hop_address > [Add an entry to the IPv6 forwarding table]
config> ipv6 route <IPv6_prefix> < next_hop_address > <interface> [Add an entry to the IPv6 forwarding table1]
config> ipv6 route <IPv6_prefix> <interface> [Add a direct entry to the IPv6 forwarding table]
config> interface <interface name > [To enter in the configuration of a specific interface->config-if]
config-if> ip address <IPv4_addr/prefix> [Configure an IPv4 address in the interface]
config-if> ipv6 address <IPv6_addr/prefix> [Configure an IPv6 address in the interface]
config-if> no ipv6 nd suppress-ra [Activate the utilization of Router Advertisements2. Dissable by default]
config-if> ipv6 nd suppress-ra [To deactivate the utilization of Router Advertisements]
config-if> ipv6 nd prefix < Pv6_prefix> [To set up the prefix to announce in Router Advertisements]
config-if> ipv6 nd ra-interval <interval_in_seconds> [To set up the time interval between Router Advertisements messages]
```

References

- [1] Manual Routers Linksys WRT54GS/GL UC3M. <http://www.it.uc3m.es/linksys/>
- [2] Jeff Doyle, Jennifer D. Carroll, "CCIE Professional Development. Routing TCP/IP, Volume I", Second Edition, CISCO Press, 2005.
- [3] Ian J. Brown, Kevin Dooley, "Cisco Cookbook", O'Reilly, 2003.

¹ This command is used to configure routes in the forwarding table when a link-local address is included in the next-hop field.

² The Router Advertisement messages through an interface allow the stateless address configuration for devices in that subnetwork. The devices must be configured to accept Router Advertisements. In Linux, we can check if an interface accepts Router Advertisements checking the content of file `/proc/sys/net/ipv6/conf/eth1/accept_ra` (for interface eth1). If the content of the file is '1', then the interface accepts Router Advertisements. If it is '0' it does not accept Router Advertisements. To modify this, you should execute in the terminal: `'sudo sysctl -w net.ipv6.conf.eth1.accept_ra=1'` o `'sudo sysctl -w net.ipv6.conf.eth1.accept_ra=0'` depending if you want to switch on/off the Router Advertisement. In the described commands eth1 can be substituted by the specific interface we want to configure.