

Quality of Service in IP Networks: An Experimental Approach

The global outcomes of this project consist of the following:

- Analyze the constraints of applications such as video transmission
- Analyze the impact of traffic shaping
- Deploy a DiffServ architecture and tune the configuration parameters
- Analyze the impact of QoS mechanisms in congestion scenarios

The description of the project and the tasks is available for download on Moodle.

-. Hardware



Figure 1. GL-iNet 6461 wireless router



Figure 2. TP-Link WR741nd wireless router

-. Software

The following tools can help in assessing the QoS on the platform:

Install and compile netperf:

```
wget https://github.com/HewlettPackard/netperf/archive/master.zip
unzip master .zip
cd netperf-master/
./configure --enable-demo=yes
make
make install
```

Install matplotlib and fping

```
apt-get install fping
apt-get install python-qt4
pip install matplotlib
```

Install flent

```
pip install flent
```

Install VLC as in this [tutorial](#).

-. Addressing

the addressing plan of the platform is shown in Fig. 2. Both routers have static addresses on their wan and lan interfaces. DHCP is activated on the LAN interfaces and static leases are configured so as to obtain the addresses on the terminals according to the figure.

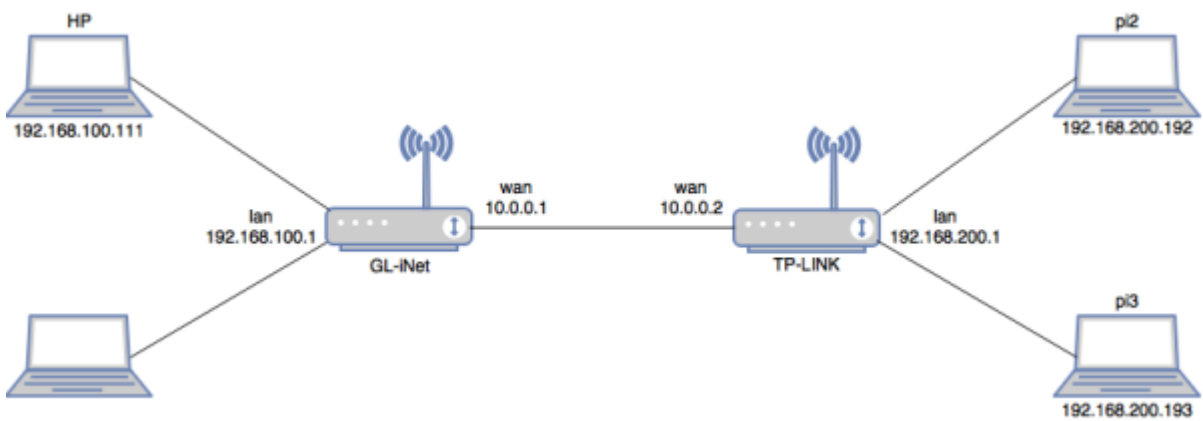


Figure 2. Addressing plan for QoS platform

-. Access and Configuration

SSH is activated on the two Raspberry Pi devices connected to the TP-LINK router. Thus, they can be accessed by typing `ssh -l pi ip_address` on any terminal. The default password is `raspberrypi` and the `pi` user is a sudoer. For advanced debugging, an HDMI cable is available so you can connect the Pi to a display monitor.

For both routers, SSH is also activated, and they can be accessed by any SSH client on the platform. For easiness, you can also connect to the WiFi interfaces (OpenWrt and GL-iNet SSID) and configure the routers from your personal laptop. The root password for GL-iNet is `helloworld`, while no password is set for OpenWrt.



Before you leave, make sure to go through the following steps.

- Gracefully shutdown the raspberry pi modules with `shutdown -h now`.



- Switch off the TP-LINK and remove the USB power cable from the GL-iNet.
- Power off the laptop.

-. [CO1] Connecting the Platform

Describe and analyze the basic steps for ensuring the platform connectivity: addressing and routing



- Accomplished
 - Enable end-to-end communication
 - Analyse the addressing and routing (routing tables, DHCP, etc.)
- Exceeded
 - Identify the necessary commands and configurations for enabling end-to-end communication

In order to analyse the addressing and routing on the platform, we need to look carefully on the interface configuration and routing tables of the different devices.

Let us start with the routing devices. The routing table of the GL-iNet is shown using the command `ip route` and gives the following:

```
10.0.0.0/24 dev eth0 proto kernel scope link src 10.0.0.1 # loopback
address
192.168.8.0/24 dev wlan0 proto kernel scope link src 192.168.8.1 #
connection to the wifi interface
192.168.100.0/24 dev eth1 proto kernel scope link src 192.168.100.1 #
connection to the LAN 192.168.100.0/24
192.168.200.0/24 via 10.0.0.2 dev eth0 proto static # static route to the
192.168.200.0/24 network via the interface
10.0.0.2 of the TP-LINK router
```

Similarly, the routing table of the TP-LINK router shows the following:

```
10.0.0.0/24 dev eth1 src 10.0.0.2 # connection to directly connected
network 10.0.0.0/24
192.168.100.0/24 via 10.0.0.1 dev eth1 # static route to the network
192.168.100.0/24 via the interface
10.0.0.1 of the GL-iNet router
192.168.200.0/24 dev br-lan src 192.168.200.1 # connection to our LAN
192.168.200.0/24
```

We note on the two routers that static routes are used in order to give access to the two LANs. Particularly, the configuration of the routers is given in the `/etc/config/network` files as on a typical OpenWrt system.

As given below, the configuration of the TP-Link router shows the static addressing of the interface

eth0 (LAN interface) and the interface eth1 (WAN interface). We also note the section route that configures a static routers towards the LAN connected to the TP-LINK router.

[/etc/config/network](#)

```
#Configuration of LAN interface:
config interface 'lan'
    option type 'bridge'
    option ifname 'eth0'
    option proto 'static'
    option ipaddr '192.168.200.1'
    option netmask '255.255.255.0'
    option ip6assign '60'

#Configuration of WAN interface:
config interface 'wan'
    option ifname 'eth1'
#    option proto 'dhcp'
    option proto 'static'
    option ipaddr '10.0.0.2'
    option netmask '255.255.255.0'

#Configuration of the static route :

config route
    option interface 'wan'
    option target '192.168.100.0'
    option netmask '255.255.255.0'
    option gateway '10.0.0.1'
```

Similarly, the configuration of the GL-iNET router

[/etc/config/network](#)

```
config interface 'lan'
    option force_link '1'
    option proto 'static'
    option ipaddr '192.168.8.1'
    option netmask '255.255.255.0'
    option ip6assign '60'
    option _orig_ifname 'eth1'
    option _orig_bridge 'false'
This interface (wifi) is configured with a static IP address
192.168.8.1/24

config interface 'wan'
    option ifname 'eth0'
    option hostname 'GL-iNet-b2d'
    option proto 'static'
    option ipaddr '10.0.0.1'
```

```
option netmask '255.255.255.0'
```

This interface (connected to the other router) is configured with a static IP address 10.0.0.1/24

```
config route
```

```
option interface 'wan'  
option target '192.168.200.0'  
option netmask '255.255.255.0'  
option gateway '10.0.0.2'
```

This route enables GL-iNET to reach the network 192.168.200.0/24 via 10.0.0.2 of the other router.

Extracts

#On TPLink router

```
config dnsmasq  
option domainneeded '1'  
option boguspriv '1'  
option filterwin2k '0'  
option localise_queries '1'  
option rebind_protection '1'  
option rebind_localhost '1'  
option local '/lan/'  
option domain 'lan'  
option expandhosts '1'  
option nonegcache '0'  
option authoritative '1'  
option readethers '1'  
option leasefile '/tmp/dhcp.leases'  
option resolvfile '/tmp/resolv.conf.auto'  
option localservice '1'  
  
config dhcp 'lan'  
option interface 'lan'  
option start '100'  
option limit '150'  
option leasetime '12h'  
option dhcpv6 'server'  
option ra 'server'  
  
config dhcp 'wan'  
option interface 'wan'  
option ignore '1'  
  
config odhcpd 'odhcpd'  
option maindhcp '0'  
option leasefile '/tmp/hosts/odhcpd'
```

```
option leasetrigger '/usr/sbin/odhcpd-update'
```

```
config host
```

```
option name 'pi2'  
option mac 'b8:27:eb:0f:f8:95'  
option ip '192.168.200.192'
```

```
config host
```

```
option name 'pi3'  
option mac 'b8:27:eb:20:aa:54'  
option ip '192.168.200.193'
```

DHCP configuration on router GL-iNet:

```
config dnsmasq
```

```
option domainneeded '1'  
option boguspriv '1'  
option localise_queries '1'  
option rebind_protection '1'  
option rebind_localhost '1'  
option local '/lan/'  
option domain 'lan'  
option expandhosts '1'  
option authoritative '1'  
option readethers '1'  
option leasefile '/tmp/dhcp.leases'  
option resolvfile '/tmp/resolv.conf.auto'
```

```
config dhcp 'lan'
```

```
option interface 'lan'  
option start '100'  
option limit '150'  
option leasetime '12h'  
option dhcpv6 'server'  
option ra 'server'  
option ra_management '1'
```

```
config dhcp 'wan'
```

```
option interface 'wan'  
option ignore '1'
```

```
config odhcpd 'odhcpd'
```

```
option maindhcp '0'  
option leasefile '/tmp/hosts/odhcpd'  
option leasetrigger '/usr/sbin/odhcpd-update'
```

```
config dhcp
```

```
option start '100'  
option leasetime '12h'  
option limit '150'  
option interface 'eth1'
```

```
config domain 'localhost'
  option name 'console.gl-inet.com'
  option ip '192.168.8.1'
```

#ifconfig on raspberry pi2

```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:0f:f8:95
          inet addr:192.168.200.192  Bcast:192.168.200.255
Mask:255.255.255.0
          inet6 addr: fdd5:bc83:a776:0:2377:1496:eb81:1433/64 Scope:Global
          inet6 addr: fe80::20ae:134e:ae88:4bc5/64 Scope:Link
          inet6 addr: fdd5:bc83:a776::192/128 Scope:Global
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:1216 errors:0 dropped:3 overruns:0 frame:0
TX packets:983 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:229130 (223.7 KiB)  TX bytes:121127 (118.2 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING  MTU:65536  Metric:1
RX packets:221 errors:0 dropped:0 overruns:0 frame:0
TX packets:221 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:20606 (20.1 KiB)  TX bytes:20606 (20.1 KiB)
```

#ip route on raspberry pi2

```
pi@raspberrypi:~ $ ip route
default via 192.168.200.1 dev eth0  metric 202
192.168.200.0/24 dev eth0  proto kernel  scope link  src 192.168.200.192
metric
202
```

From:

<http://wiki.lahoud.fr/> - wikiroute

Permanent link:

http://wiki.lahoud.fr/doku.php?id=qos_project&rev=1511943997

Last update: **2017/11/29 09:26**

