# Introduction to lab assignments with GNS3

## User guide and helpful tips

Roberto Bonafiglia, Fulvio Risso

# Contents

# License

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License.

You are free:

- **to Share**: to copy, distribute and transmit the work
- **to Remix**: to adapt the work

Under the following conditions:

- **Attribution**: you must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
- **Noncommercial**: you may not use this work for commercial purposes.
- **Share Alike**: if you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

More information on the Creative Commons website (`http://creativecommons.org`).

# Acknowledgments

The authors would like to thank all the people who contributed to this document.

# 1 Requirements

Lab assignments can be carried out thanks to the GNS3 software, a network emulator that exploits the operating system of real network devices to create and emulate a complex network. In order to facilitate the deployment of the software, we provide a pre-installed virtual machine (VM) that can be either directly executed in your VM manager, or converted and transferred on an USB stick in order to create a bootable disk with the pre-installed software.

Particularly, the VM for the networking labs @ Politecnico di Torino requires the Virtual Box software running on your PC. This software is available for most operating systems; downloading instructions are available at `https://www.virtualbox.org/wiki/Downloads`.

Additional tools can be used to convert the VM in another format, e.g., for Linux KVM or to make it an ISO-bootable image.

# 2 Setting up the VM

1. Download the VM from `https://owncloud.ipv6.polito.it/index.php/s/vk2NVyynPgrmQr9` (this link is available only within the POLITO network)

2. Launch VirtualBox and select *File->Import Appliance...*

3. On the new window select the file downloaded before and continue until it requires to import the VM. Although at that point you will be asked to configure other parameters, they have already been configured so you can simply confirm them all.

4. After the import operation the VM is ready to run. The new VM will appear in the VM list on the main Virtual Box window, and it can be started by *double-clicking* on it.

5. The VM is based on an Ubuntu 16.04; the login username is `gns3` and the password is `courseVM`.

# 3 Using GNS3

1. Start the VM and wait until the main Ubuntu screen appears.

2. Open the terminal  and type *gns3*.

3. From the new window you can either create a new project or open an existing one, as shown in Figure 1. GNS3 projects are used to define different network topologies, so if it its the first time you execute the VM, you should create a new project.

4. The  icon allows to browse all the available routers. However, please note that all the exercises will require only:

   - a Cisco 2691 router, when an L3 device is needed[1];
   - the Ethernet switch that emulates a LAN.

---

[1]Note that any other router model will not work because the corresponding IOS image has not been saved in the VM.
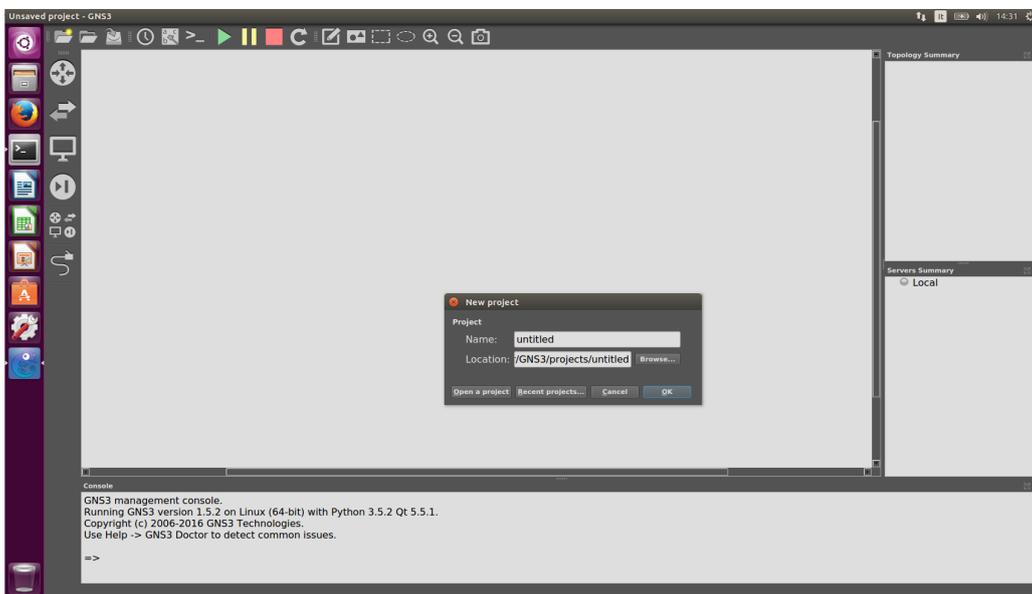
Figure 1: Main GNS3 window when opening an existing project or creating a new one.

# 4 Simulate your first topology

Now we can try to create a simple network topology with two routers directly connected through a link, as depicted in the Figure. First of all we need to create our topology. Go to the Device Library and drag and drop the *Cisco 2691* router to the network window. After inserting the routers we need to create a link that connects the two devices together. To do this select the ⇄ button and select the first router as the source device and a pop-up appears like the one depicted on the Figure with the list of the network devices.
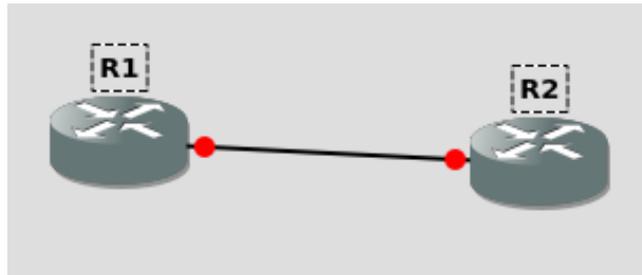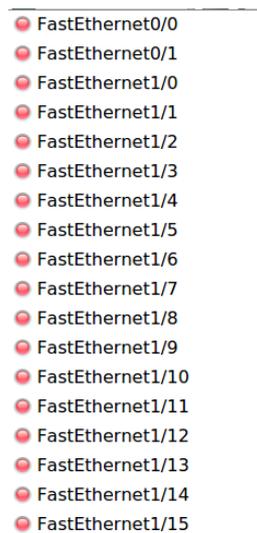


Figure 2: Our first network topology



Figure 3: Router Link Form

Select a network adapter from the menu option (e.g., *FastEthernet0/0*), the red icon means that the interface is not used when it is selected it will signed with a green one, after the interface selection click the second device to connect and do the same operation. Now that the two routers are connected they will be linked and we just need to activate the virtual machines associated to the routers, which translates into the "start topology" command. Routers can be configured through the "console" only when they are active, i.e., their associated virtual machine is up and running.

For starting the topology (which is required in order to enable the console), we can press ▶ button.

GNS3 uses different icons to show the status of the router. For instance, a router that is switched off is shown with a red icon, while an active router is shown with the green icon. When the router is active, we can open the associated console window and configure the device with a double click on the icon. To configure our topology, open the console of the **router R1** and press the Enter key on your keyboard a couple of times; this is a known problem, as sometimes the router prompt does not appear if you do not press **Enter**. If everything is ok, after some seconds you should see the following prompt on the console:

```
R1#
```

You will configure the machines manually. You need now to:

- Enter in *privileged-mode* of the router using the command `enable`;
- Enter in *configuration mode* using the command `configure terminal`[2].

Now we need to select the interface we want to configure. To do that we need to use the command `interface FastEthernet x/y` where "x" is the number of the slot you chose before, and "y" is the number of the interface (within that slot) you used. You can also use the shortcut "`in fa x/y`". For instance, as we choose the slot 0 and interface 0, we write the command `interface FastEthernet 0/0`. Now we will configure the IP address with the following commands:

| Command | Description |
|---|---|
| `ip address 192.168.0.1 255.255.255.0` | Configure the IP address 192.168.0.1/24 on the selected interface. This address will become immediately active. |
| `no shutdown` | Turn on the interface. Some interfaces of the router are turned off by default, hence it is always better to type this command although this may not be always needed. |
| `end` | Exit from the configuration mode. |

Obviously, a similar configuration must be repeated on the second router. On the console of R1 you can apply do the same commands, just changing the IP address into 192.168.0.2.

Here a brief summary of all the commands you typed in order to configure the address on the router, where the last digit of the IP address (written as "z") will be the actual IP address of the router:

```
enable
configure terminal
  interface FastEthernet0/0
    ip address 192.168.0.Z 255.255.255.0
    no shutdown
  end
```

You have now configured the interfaces on the link.

---

[2]Notice that you can also use shortcuts such "`ena`" and "`conf t`" for those commands.

## 4.1 Verify that the network operates correctly

Now that you finished your configuration you can use the `ping` command in order to generate a stream of 5 ICMP Echo Request and ICMP Echo Reply to verify that the network topology works and that routers can exchange traffic between them. Open the console of the router `R1`. Assuming that the configuration of the router is the one suggested in previous section, you just need to use the command below to ping the router `R2`.

```
ping 192.168.0.2
```

You should see a similar text appearing on each console:

```
Router# ping 192.168.0.2
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
.!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 4/4/20 ms
Router#
```

Please note that Cisco routers tend to miss the first reply to the `ping` command, while the remaining four are received correctly. A missing answer is shown with the dot sign ".", while a successful answer is shown with the exclamation mark ("!"). Therefore, if you type the `ping` command for the first time and the first (or, in some cases, the first two) answers are missing, nothing is wrong on your side. Vice versa, if all the five answers are negative, something is wrong in your configuration.

## 4.2 Make your first capture

Now we can try to capture the traffic generated by the `ping` command.

Right click on the link and select from which interface you start the capture and press *ok*. This will start capturing the traffic generated on that link opening a new wireshark window.

Now come back to the router `R1` and type again the `ping` command as we saw before.

On the Wireshark window you will see the stream of packet generated between the two routers, i.e., most likely 5 ICMP Echo Requests and 5 ICMP Echo Replies. Some additional traffic may be present, which is generated by some protocols active on the router and that are used to exchange some data on the network. For instance, the Cisco Discovery Protocol (CDP) is active by default on a Cisco device and it is used to discovery the capabilities (e.g., installed hardware, software version, etc.) of the devices present on the network.

# 5 Appendix: Creating an ISO bootable disk

This section briefly presents a possible way to convert the VM image into a bootable ISO image that can be written on a USB stick.

In this section we make use of some free tools, such as *VirtualBox* and *HDD Raw Copy*, that are based on the Windows operating system.

- Open a command prompt and move to the folder in which VirtualBox has been installed (e.g., `C:\Program Files\Oracle\VirtualBox`).

- Type the following command:

  ```
  VBoxManage internalcommands converttoraw Downloaded_VM_Image_Name.ova
  BootableImageName.iso
  ```

  where `Downloaded_VM_Image_Name.ova` is the name of the VM image you downloaded from the POLITO servers, and `BootableImageName.iso` is the new ISO image that has to be created.

- Now you can use HDD Raw Copy to copy the image on an USB stick. Please note that (*i*) your USB stick will be completely overwritten by the new image, and (*ii*) the raw size of your USB stick must be greater than 8 billions of bytes (approx 8GB).

Now you can reboot your PC and press the proper keys in order to use the USB stick as boot disk.