Segurança em Redes de Comunicações

Corporate Networks Fundamentals

PART A

Traffic Monitoring (with Wireshark)

Install Wireshark on your PC.
In Linux, add your user name to the wireshark group (usermod -a -G wireshark USERNAME) and restart. In Windows if your network adapters are nor listed, start Wireshark as administrator.

1. Start a capture on the interface that provides Internet to your PC. Open your browser and access your favorite sites. Stop the capture and analyze the packets.
>> Try to identify the packets exchanged between your PC and the servers.
>> Try to identify used protocols and how are they encapsulated at different layers.
>> Identify the IP addresses of the hosts that exchanged packets.

You may save the capture with File→Save and reopening a previous capture with File→Open.

2. Start a capture on the interface that provides Internet to your PC. Perform a set of connectivity tests (pings). Stop the capture and analyze the packets.
>> Try to identify the packets exchanged.
>> Try to identify used protocols and how are they encapsulated at different layers.
>> Identify the IP addresses of the hosts that exchanged packets.

3. Start a capture on the interface that provides Internet to your PC. Go to YouYube and play a video. Stop the capture and analyze the packets. If possible, repeat the capture using a different browser (Firefox or Chrome/Chromium).
>> Try to identify the packets exchanged.
>> Try to identify used protocols and how are they encapsulated at different layers.
>> Identify the IP addresses of the hosts that exchanged packets.

4. Open the previous captures and apply the following visualization filters:
- IPv4 packets sent or received by your PC: ip.addr==<ipv4_address>
- IPv4 packets sent by your PC: ip.src==<ipv4_address>
- IPv4 packets received by your PC: ip.dst==<ipv4_address>
- Only ICMP packets: icmp
- ICMP **and** ARP packets: icmp **or** arp
- Only TCP packets: tcp
- Only HTTPS (TCP port 443): tcp.port==443

Note: Replace <ipv4_address> by your PC IPv4 address.
Note2: Wireshark has capture and visualization filters. At this stage do do not use capture filters.

5. Open one of the previous captures and explore the traffic analysis features from Wireshark. From Wireshark menu:
- Statistics→ Endpoints
- Statistics→ Conversations
- Statistics→ I/O Graphs

PART B

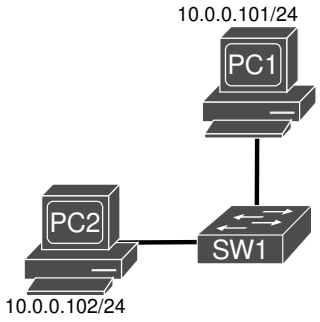
Network Deployment Fundamentals

1. Choose your operating system (Linux/Windows), download/install GNS3 (version>2.2.0) and related software (Wireshark, VirtualBox, QEMU and VPCS).
(Windows and MacOS) Download package from website https://gns3.com.
(Linux) Install from repositories; AUR for Arch/Manjaro distributions and PPA https://launchpad.net/~gns3/+archive/ubuntu/ppa for Debian/Ubuntu based distributions. Install packages gns3-server, gns3-gui, wireshark-qt, virtualbox, qemu, and VPCS. Add your user name to the wireshark group (usermod -a -G wireshark USERNAME) and restart.

2. At (Preferences-General), verify/setup all storing and program paths, avoiding paths with spaces and non ASCII characters.

3. At (Preferences-Server) enable **local server**, define **127.0.0.1** as host binding address.
Note: You do not need an external virtual machine (VM) to run emulation/simulation software. At (Preferences-GNS3 VM) disable the option “Enable the GNS3 VM”.

4. Create a new Blank Project (File menu or CTRL+N) and give it a name. Add Two Virtual PC (VPCS) and one basic Layer 2 Switch (Ethernet Switch) to your project.
Note: VPCS only provide basic PC emulation and should only be used to test network connectivity.
Note2: A basic Layer 2 Switch (Ethernet Switch) does not run Spanning-Tree Protocol and should not be used in redundant Layer 2 networks (with connection loops).



5. Interconnect them and configure their IPv4 address according to the above network diagram. Start all nodes in the project, double left-click on the VPCS to open their consoles, and configure their IPv4 addresses:
```
PC1> ip 10.0.0.101/24
...
PC2> ip 10.0.0.102/24
```
Verify the current VPCS IPv4 configuration with the command:
```
PC> show ip
```

6. Start a capture on the link between PC1 and SW1 (Right-click-Start Capture). From PC1 console test connectivity with PC2 with the ping command (and vice-versa):
```
PC1> ping 10.0.0.102
...
PC2> ping 10.0.0.101
```
>> Analyze the captured packets. Namely, Ethernet header (MAC address) and IPv4 header (IPv4 addresses).

7. Download the following routers' firmware: (i) Router 7200 Firmware 15.1(4) IOS Image, and (ii) Router 3725 Firmware 12.4(21) IOS Image.

8. At (Preferences-Dynamips-IOS Routers") create three new router templates ("New" button on the bottom left):
- **Router 7200** - recommended IOS image: 7200 with IOS 15.1(4) and network adapters C7200-IO- 2FE and PA-2FE-TX (4 FastEthernet → F0/0,F0/1+F1/0,F1/1), all other values can be the default ones;
- **Router 3725** - recommended IOS image: 3725 with IOS 12.4(21) and adaptersGT96100-FE and NM-1FE-TX (2 FastEthernet), all other values can be the default ones;
- **Switch L3** – will be a router 3725 with IOS image 12.4(21) and adapters GT96100-FE and NM-16ESW (1 FastEthernet + 16 port switch module). Choose option "This is an EtherSwitch router" when defining the device platform, all other values can be the default ones.

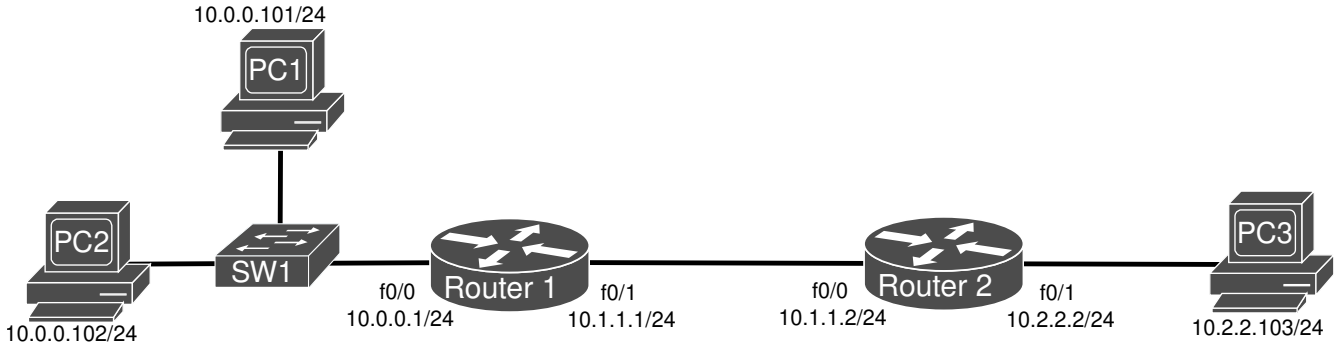
9. The definition of the "Idle-PC" value will allow the host machine to assign the correct amount of resources to the virtual devices. You must repeat this procedures every time your PC CPU reaches values higher than 90%. Check the CPU utilization with the "Task Manager" in Windows, top command in Linux and "monitor" in MacOS.
To define the "Idle-PC" value:
- Click "Idle-PC finder" during template setup, OR
- Add router to project, start it (should be the only one ON), open console (wait for prompt), left click the device and choose option "Auto Idle-PC", OR
- Add router to project, start it (should be the only one ON), open console (wait for prompt), left click the device and choose option "Idle-PC", choose one value (prefer the ones marked with *) and verify the CPU utilization. If any "dynamips" process is using more than 5%-10% CPU choose another value.

This must be done for each router template, NOT each router! Each template will have a different "Idle-PC" value. All routers from the same template will share the same value.
Note 1: All devices from the same template must be equal in terms of virtual hardware.
Note 2: After changing any device hardware characteristic or adding/removing network modules, the "Idle-PC" value must be changed in the template. If necessary, create a new template with different characteristics/modules.

At this phase your GNS3 installation should have (at least):
- 2 Routers: a Cisco c7200 and a Cisco c3725;
- An "EtherSwitch" (Layer 3 switch) based on a router c3725 with a 16 port switch module;

Router configurations

10. Add two routers (c7200) and a third PC, and interconnect them according to the diagram below.



11. Preform IPv4/Routing configurations in Router1 (e.g., IP address/mask, activation of interfaces, and dynamic routing protocol - OSPFv2):

```
Router1# configure terminal
Router1(config)# interface FastEthernet 0/0
Router1(config-if)# ip add 10.0.0.1 255.255.255.0
Router1(config-if)# ip ospf 1 area 0
Router1(config-if)# no shutdown
Router1(config-if)# interface FastEthernet 0/1
Router1(config-if)# ip add 10.1.1.1 255.255.255.0
Router1(config-if)# ip ospf 1 area 0
Router1(config-if)# no shutdown
Router1(config-if)# end
Router1# write
```

Do an equivalent configuration in Router2, and verify the Routers' IPv4 routing tables with the command:

```
Router1# show ip route
```

>> Analyze the routing tables in both routers.

Note: You may save the router configuration in an external file; right-click the router and save configuration. Analyze the configuration (*.cfg) file created in your project folder.

12. (Re)Configure the PC IPv4 addresses and respective gateway.

```
PC1> ip 10.0.0.101/24 10.0.0.1
...
PC2> ip 10.0.0.102/24 10.0.0.1
...
PC3> ip 10.2.2.102/24 10.2.2.2
```

Start a capture on the link between Router1 and Router2 (Right-click-Start Capture). Test connectivity between all devices.

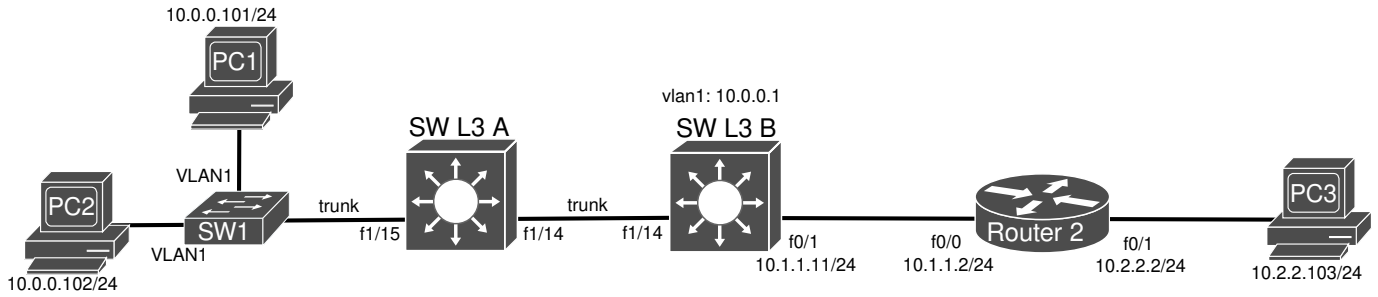
>> Analyze the captured packets. Namely, the OSPF and ICMP packets.

Note: A PC gateway is the IP address of the router in than LAN that interconnects it to other LAN.

13. Replace Router1 with two Layer3 Switches (EtherSwitch routers), and interconnect them according to the diagram below.

Place the cursor over Layer3 Switches (A and B) and identify (on the pop-up) the 16 ports numbered from FastEthernet x/0 to FastEthernet x/15, those are the ones of the switch module that must be used to connect to other switches using Trunk links (usually is F1/0 to F1/15). Connect SW1 to port FastEthernet x/15, and use ports FastEthernet x/14 to interconnect SWL3 A and B. The connection between SW L3 B and Router2 is a Layer3 connection and must be done using Layer3 interfaces (non-switch port, e.g. FastEthernet 0/1).

Note: a Layer3 switch Layer2 port (switch port) can be converted to a Layer3 (non-switch) port with the command no switchport.



14. To configure SW1, double left-click and (i) define the switch ports connected to PC1 and PC2 as VLAN1 and access ports, and (ii) define the port connected to SW L3 A as trunk port (protocol 802.1Q - type dot1q).

15. Start both Layer3 switches by double right clinking it and choosing Start. Wait a few seconds, open the console of the switches by double left clinking it, and press enter to obtain prompt.
>> Use the command show ip interface brief to verify the status of the connected ports. The status and protocol (of the ports in use) should be up and up. If not, stop and start the equipment.

16. To configure SWL3 A, first, verify the existence of VLAN1 on the switch database:

```
SWL3A# show vlan-switch
```

Second, configure ports F1/14 and F1/15 as trunks

```
SWL3A# configure terminal
SWL3A(config)# interface FastEthernet 1/15
SWL3A(config-if)# switchport mode trunk
SWL3A(config)# interface FastEthernet 1/14
SWL3A(config-if)# switchport mode trunk
SWL3A(config-if)# end
SWL3A# write
```

Note: To verify Cisco routers running configuration type: show run.

17. To configure SWL3 B, it is necessary to configure the trunk port to SW L3 A, configure the virtual interface of VLAN1 and Layer interface F0/1 with IPv4 addresses, and activate dynamic IPv4 routing (OSPFv2 protocol).
Verify the existence of VLAN1 on the switch database:

```
SWL3B# show vlan-switch
```

Configure port F1/14 as trunks:

```
SWL3B# configure terminal
SWL3B(config)# interface FastEthernet 1/14
SWL3B(config-if)# switchport mode trunk
```

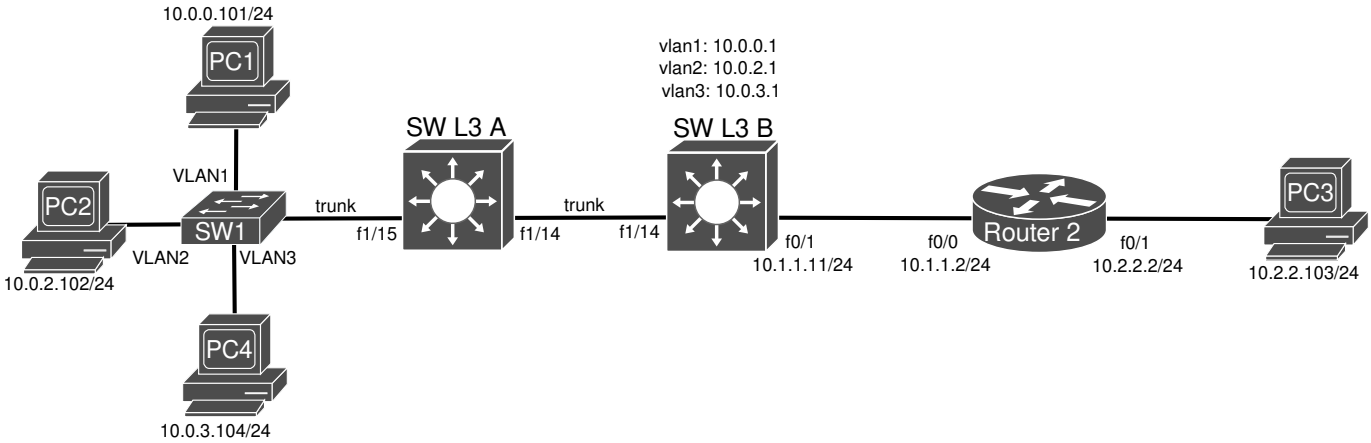
Activate IPv4 routing, and configure virtual interface VLAN1 address and routing:

```
SWL3B(config)# ip routing
SWL3B(config)# interface vlan 1
SWL3B(config-if)# no autostate
SWL3B(config-if)# ip address 10.0.0.1 255.255.255.0
SWL3B(config-if)# ip ospf 1 area 0
SWL3B(config-if)# no shut
```

Configure Layer3 interface with address and routing:

```
SWL3B(config)# interface F0/1
SWL3B(config-if)# ip address 10.1.1.11 255.255.255.0
SWL3B(config-if)# ip ospf 1 area 0
SWL3B(config-if)# no shut
SWL3B(config-if)# end
SWL3B# write
```

Note: To verify Cisco routers running configuration type: show run.
>> Analyze IPv4 routing tables of SWL3 A, SWL3 B, and Router2.
>> Test connectivity between all devices.



18. Add a fourth PC (PC4) to VLAN3, and change PC2 to VLAN2.
Note: A device VLAN depends on the VLAN configured on the access switch (and IPv4 address).
Reconfigure SW1, double left-click and (i) define the switch ports connected to PC2 as VLAN2 and access, and (ii) define the switch ports connected to PC4 as VLAN3 and access. (Re)Configure PC2 and PC4 IPv4 addresses and gateways:

```
PC2> ip 10.0.2.102/24 10.0.2.1
...
PC4> ip 10.0.3.104/24 10.0.3.1
```

19. Reconfigure SWL3 A and SWL3 B by adding VLAN2 and VLAN3 to their databases:

```
SWL3A# vlan database
SWL3A(vlan)# vlan 2
SWL3A(vlan)# vlan 3
SWL3A(vlan)# exit
SWL3A# show vlan-switch
```

Configure SWL3 B virtual interfaces for VLAN2 and VLAN3 (address and routing):

```
SWL3A(config)# interface vlan 2
SWL3A(config-if)# no autostate
SWL3A(config-if)# ip address 10.0.2.1 255.255.255.0
SWL3A(config-if)# ip ospf 1 area 0
SWL3A(config-if)# no shut
SWL3A(config)# interface vlan 3
SWL3A(config-if)# no autostate
SWL3A(config-if)# ip address 10.0.3.1 255.255.255.0
SWL3A(config-if)# ip ospf 1 area 0
SWL3A(config-if)# no shut
SWL3A(config-if)# end
SWL3A# write
```

Note: To verify Cisco routers running configuration type: show run.
>> Analyze IPv4 routing tables of SWL3 A, SWL3 B, and Router2.
>> Test connectivity between all devices.

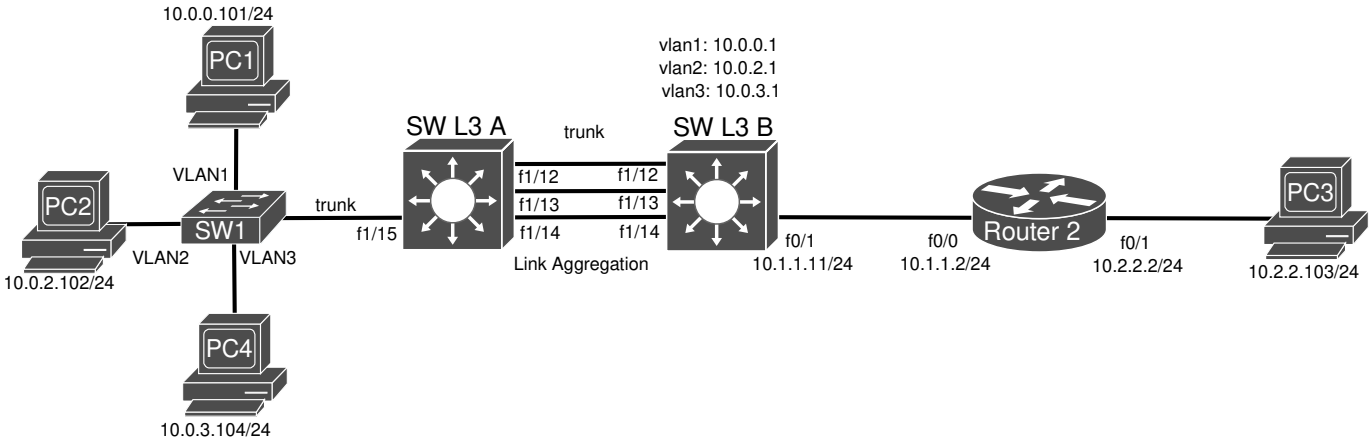
20. Start a capture on the link between SWL3A and B, from PC2 and PC4 ping their gateways.
>> Analyze the captured packets. Namely, the ICMP packets' Ethernet header (802.1Q protocol).
>> How Layer2 packets from different VLAN are discriminated in a trunk link?

21. Trunk links can be restricted to some VLAN traffic. On the trunk link between SWL3A and B allow only traffic from VLAN1 and VLAN2:

```
SWL3A# configure terminal
SWL3A(config)# interface FastEthernet 1/14
SWL3A(config-if)# switchport trunk allowed vlan 1,2,1002-1005
```

Note: VLAN 1002-1005 permissions are required by Cisco mechanisms and cannot be be restricted.

>> Test connectivity between all devices. Verify if PC4 has connectivity with its gateway.



22. Lets assume the bandwidth between SWL3A and B must be tripled. First, remove the trunk permissions. Link aggregation must be done with trunks with the same configuration/permissions.

```
SWL3A(config)# interface FastEthernet 1/14
SWL3A(config-if)# no switchport trunk allowed vlan 1,2,1002-1005
```

Add two additional links between the Layer3 switches and aggregate them (in an EtherChannel) as a unique trunk:

```
SWL3A(config)# interface range FastEthernet 1/12 - 14
SWL3A(config-if-range)# channel-group 1 mode on
SWL3A(config-if-range)# interface Port-channel 1
SWL3A(config-if)# switchport mode trunk
```

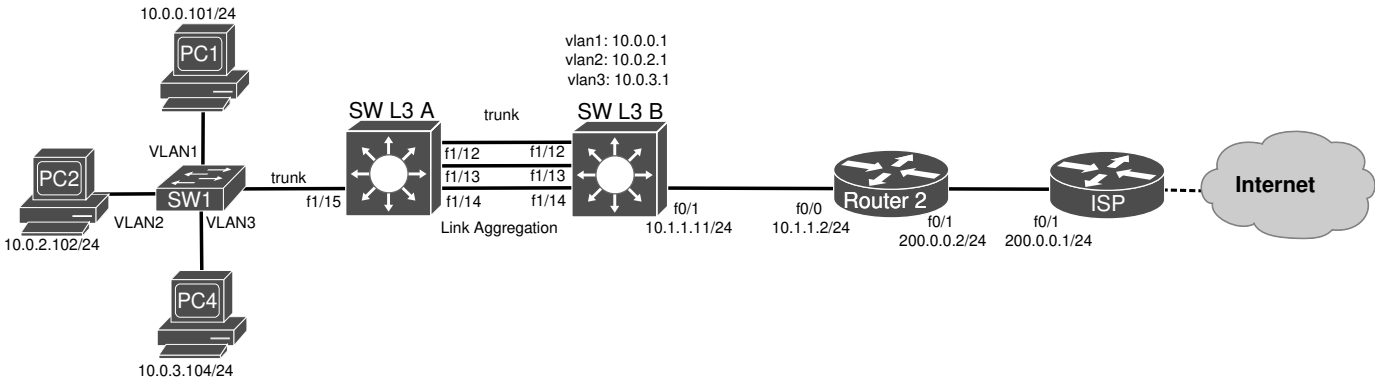
Perform the same configurations in SWL3 B.

Verify the correct implementation of the EtherChannel:

```
SwitchL3A# show ip interface brief
SwitchL3A# show etherchannel brief
SwitchL3A# show etherchannel detail
SwitchL3A# show etherchannel summary
SwitchL3A# show etherchannel load-balance
```

>> Test connectivity between all devices.

>> Start captures on the aggregated links. Verify the load balancing process.



23. Lets assume that Router2 is the network Internet access. Remove PC3 and replace it with an additional router. Reconfigure Router2's F0/1 interface address and remove OSPFv2 activation from that interface.

```
Router2# configure terminal
Router2(config)# interface FastEthernet 0/1
Router2(config-if)# ip add 200.0.0.2 255.255.255.0
Router2(config-if)# no ip ospf 1 area 0
Router2(config-if)# no shutdown
```

Note: The IPv4 routing between the network and ISP can not be the same internal routing process. May be another OSPF process, another protocol, or static routing.

24. Lets assume that the network has the public IPv4 network 100.0.0.0/29. Configure static routing from Router2 to ISP, ISP IPv4 address, and static routing from ISP to Router2.

```
Router2(config)# ip route 0.0.0.0 0.0.0.0 200.0.0.1
---
ISP(config)# interface FastEthernet 0/1
ISP(config-if)# ip address 200.0.0.1 255.255.255.0
ISP(config-if)# no shut
ISP(config-if)# exit
ISP(config)# ip route 100.0.0.0 255.255.255.248 200.0.0.2
ISP(config)# end
ISP# write
```

Note: The network prefix 0.0.0.0/0 is the default route, includes all IPv4 networks and is used as a generic prefix towards Internet.

>> Test connectivity between the PC and the IPv4 public addresses (200.0.0.2 and 200.0.0.1).

>> Explain the lack of connectivity.

25. Router 2 must dynamically announce a default for all internal routers:

```
Router2# configure terminal
Router2(config)# router ospf 1
Router2(config-router)# router ospf 1
Router2(config-router)# default-information originate always
```

>> Verify the new entry on SWL3 B IPv4 Routing table: `show ip route`

>> Test connectivity between the PC and the IPv4 public addresses (200.0.0.2 and 200.0.0.1).

26. **Packets with a IPv4 private address as source must never flow on public networks (Internet).** It is required to activate address and port translation mechanisms (NAT/PAT) in Router 2:

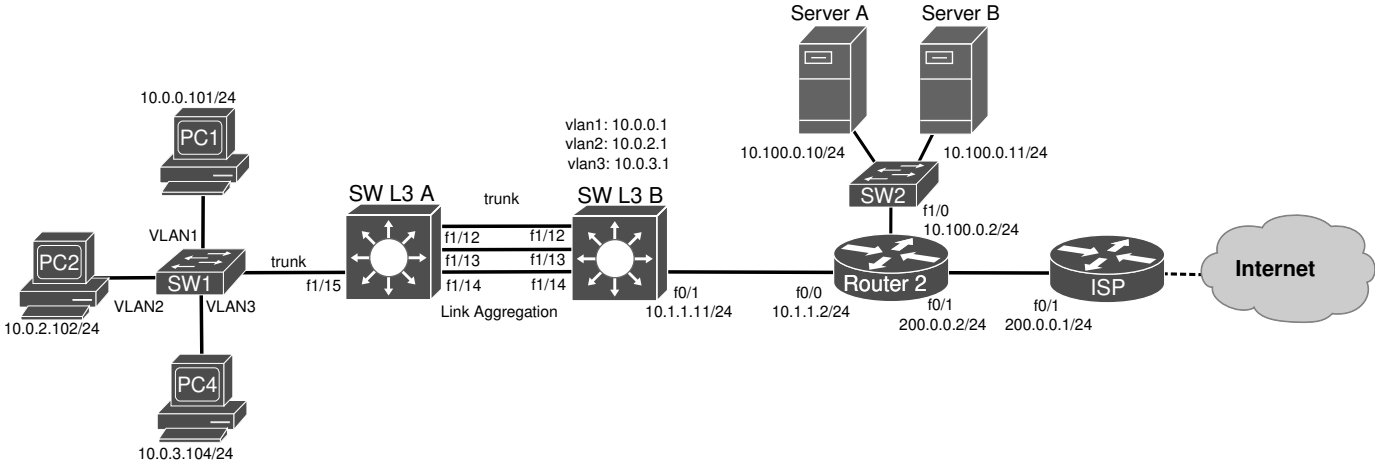
```
Router2(config)# ip nat pool MYNATPOOL 100.0.0.1 100.0.0.7 netmask 255.255.255.248
Router2(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Router2(config)# ip nat inside source list 2 pool MYNATPOOL overload
Router2(config)# interface FastEthernet 0/0
Router2(config-if)# ip nat inside
Router2(config-if)# interface FastEthernet 0/1
Router2(config-if)# ip nat outside
Router2(config)# end
Router2# write
```

>> Test connectivity between the PC and the IPv4 public addresses (200.0.0.2 and 200.0.0.1).

>> Verify the NAT/PAT translation table in Router2: show ip nat translation

>> Capture packets in the link between Router2 and ISP. Verify that the packets source address has been changed.

Part C



Interconnection with virtual machines (VirtualBox)

1. Go to (Edit-Preferences-VirtualBox-VirtualBox VMs" and create a new VM template based on an existing VirtualBox machine. Use an Debian LXDE VirtualBox (login/password: labcom/labcom).

Note1: To use the VM in GNS3, the VM should be powered off and the network adapter should be "not attached".

Note2: To connect the VM to the Internet, start the VM from VitualBox GUI with the network adapter attached to "NAT".

Note3: To use multiple VM instances, you may clone the original machine.

2. Configure interface F1/0 of Router2 address and routing, and define it as a NAT/PAT internal interface:

```
Router2(config)# interface FastEthernet 1/0
Router2(config-if)# ip address 10.100.0.2 255.255.255.0
Router2(config-if)# ip ospf 1 area 0
Router2(config-if)# ip nat inside
Router2(config-if)# no shutdown
```

3. In your project, add Server A as an end device based on the created VM template. Configure its IPv4 address and gateway, as root do:

```
ip link set up dev eth0
ip addr add 10.100.0.10/24 dev enp1s0
ip route add default via 10.100.0.2
```

>> Test connectivity to the other network devices.

Note: your virtual Ethernet port may have another name. List devices with `ip addr` to identify it.

Interconnection with virtual machines (QEMU)

4. Go to (Edit-Preferences-QEMU-QEMU VMs" and create a new VM template based on an existing virtual disk image (*.img). Use an Debian LXDE QEMU virtual disk (LabComServer2.qcow2) (login/password: labcom/labcom).

Note1: To use the VM in GNS3, the VM should be powered off.

Note2: To connect the VM to the Internet, start the VM from the command line (or *virt-manager*) using the command "qemu-system-x86_64 -m 1024 -enable-kvm LabComServer2.qcow2".

Note3: To use multiple VM instances, you may copy the original VM disk file "LabComServer2.img" and start another VM.

Note4: In Windows, QEMU requires HAXM, see how to install here. Also, replace option "-enable-kvm" with option "-accel hax" when running from the command line.

5. In your project, add Server B as an end device based on the created VM template. Configure its IPv4 address and gateway, as root do:

```
ip link set up dev eth0
ip addr add 10.100.0.11/24 dev enp1s0
ip route add default via 10.100.0.2
```

>> Test connectivity to the other network devices.