Institute of Telecommunications
Faculty of Electronics and Information Technology
Warsaw University of Technology, Nov 2019

EINTE LAB EXERCISES

LAB EXERCISE – MPLS

1. PURPOSE AND GOALS

MPLS is a high-performance method for forwarding packets through a network and enables routers at the edge of a network to apply simple labels to packets. MPLS integrates the performance and traffic management capabilities of data link Layer 2 with the scalability and flexibility of network Layer 3 routing. This lab assignment will give you a hands-on experience in configuring MPLS and basic traffic engineering using MPLS tunnels. You will prepare the lab session, perform the practical exercises, and write a lab report, which describes the whole lab, including planning, preparations, and results. The lab report will be assessed; grading will be added to the total course score - check the course home page for details.

2. OVERVIEW

The Assignment is divided into preparation, practical part, and documentation.

A. PREPARATIONS

The preparations are crucial to the successful execution of the main exercise. Without proper study there is no possibility to fulfill the requirements within the specified time; trial-and-error will not work!

The Appendix contains some basic information related to Cisco IOS software, that you should read and understand to follow the instructions provided in this document. However, during this phase it is advised to study the additional Cisco routers documentation available in the Internet if any additional explanations to relevant IOS commands are required. Finally, you should refresh and expand your knowledge on the topic of the lab exercise.

B. EXERCISE

The main exercise is executed using the VirtualBox virtual machine available for download from the course web page.

The virtual machine (VM) runs the router emulation environment configured such that the individual router consoles are accessible via telnet service on specific TCP ports. The environment emulates Cisco routers, and so the router operation system is IOS. The emulated network topology is fixed – changes can be only introduced by opening and closing individual router interfaces.

To execute the lab you need to download and install the following free software:

- Oracle VirtualBox and VirtualBox Extension Pack, both available for download at https://www.virtualbox.org/wiki/Downloads
- MobaXTerm terminal for Windows, available at https://mobaxterm.mobatek.net/download-home-edition.html

The guide to running the virtual router lab can be found in Section 3. Note that if you already have the VM downloaded and installed (e.g. from previous lab exercise), you can skip the entire Section 3.

The tasks required to pass the exercise are described in detail in Section 4.

C. FINAL REPORT

The last phase consists of lab report preparation. You should plan the outline of the report in advance, during the preparation phase to be sure what input is necessary before attempting the main part of the exercise.

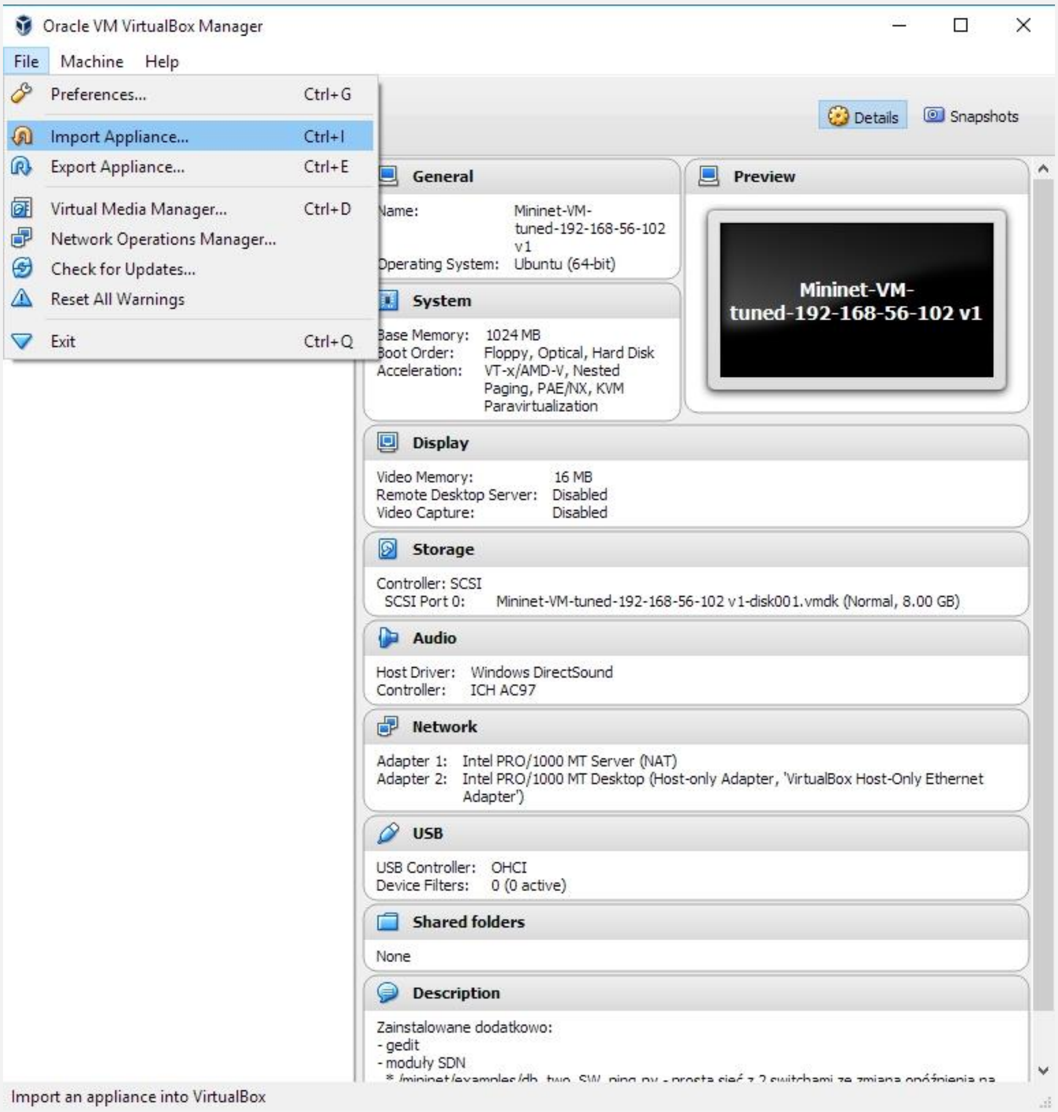
All phases of the exercise and your findings collected during the main practical part should be documented in the report. This instruction may also contain questions and remarks (usually marked with distinct colour) that shall be answered or considered in the report. Finally, you should attach the configuration files (running configs) from all routers with the final configurations that you have prepared during the exercise.

Note: If you find any errors or inconsistencies in this document and referenced manuals, please report them to the lab exercise supervisor(s). It will help to improve the lab exercise in the future.

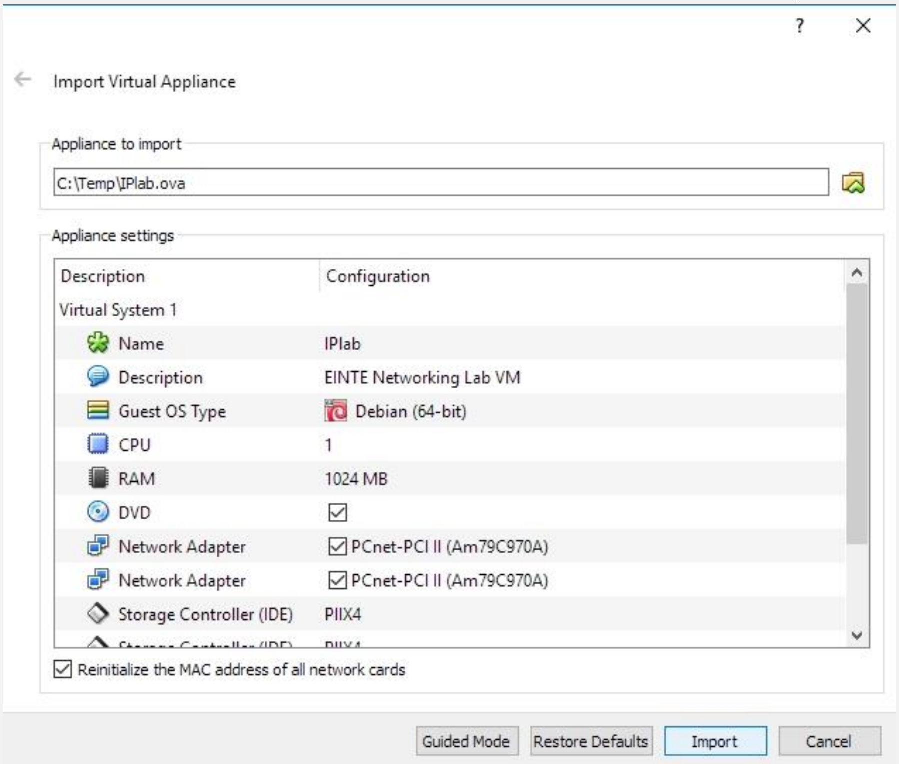
3. USING THE VIRTUAL LAB ENVIRONMENT

After downloading and installing the software indicated in Section 2B on your local PC, the following steps are necessary to run and use the virtual router lab environment.

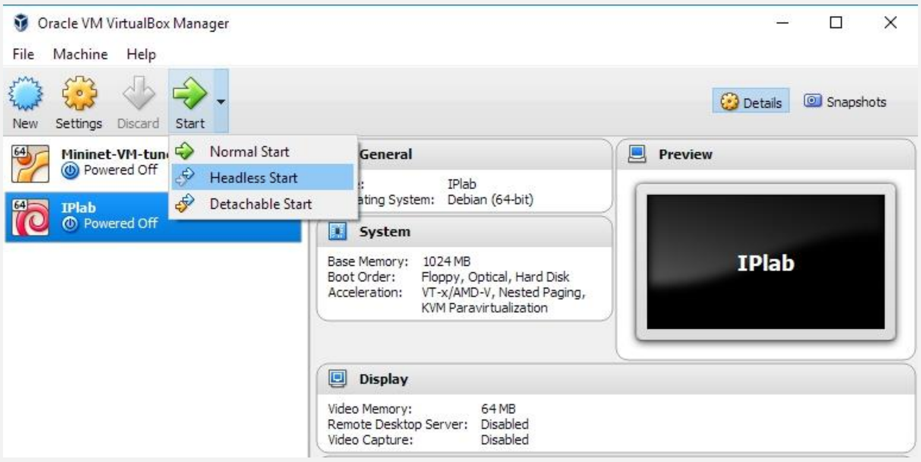
1. Run the VirtualBox and select File/Import Appliance from the main menu



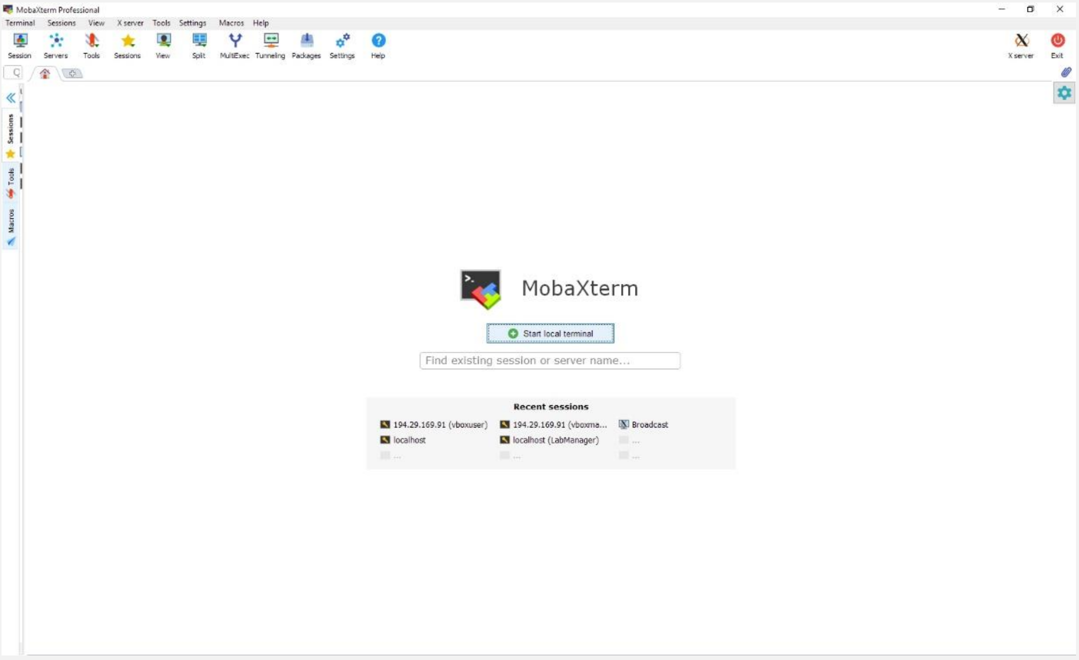
2. Select the downloaded EINTE IP Lab virtual machine (in example below the name is IPlab.ova), check the “Reinitialize the MAC address of all network cards” checkbox and click the “Import” button.



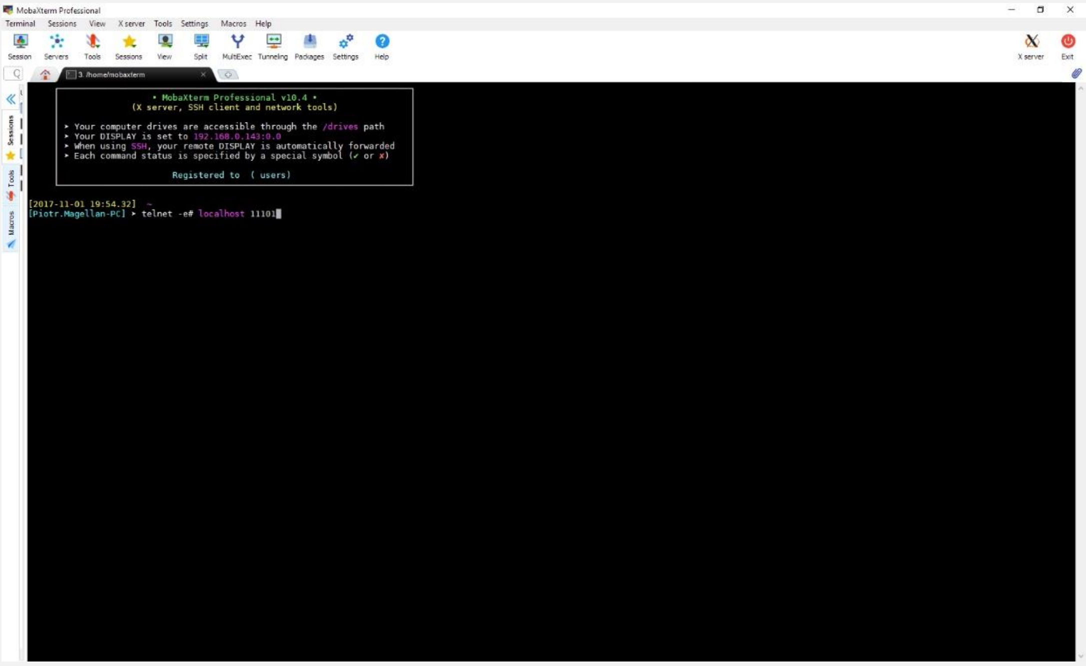
3. After the IPlab VM (Virtual Machine) is imported, select it on the panel, click the right side of the “Start” button and select “Headless Start” from the drop-down menu. The VM is based on Debian linux without any graphical interface, and needs about one minute to fully boot (optionally, you can observe the boot process by selecting the “Show” button, available after the VM is started – this will open the separate window with the VM). <u>Note: there’s no need to log in to the VM – you just need to start it.</u>



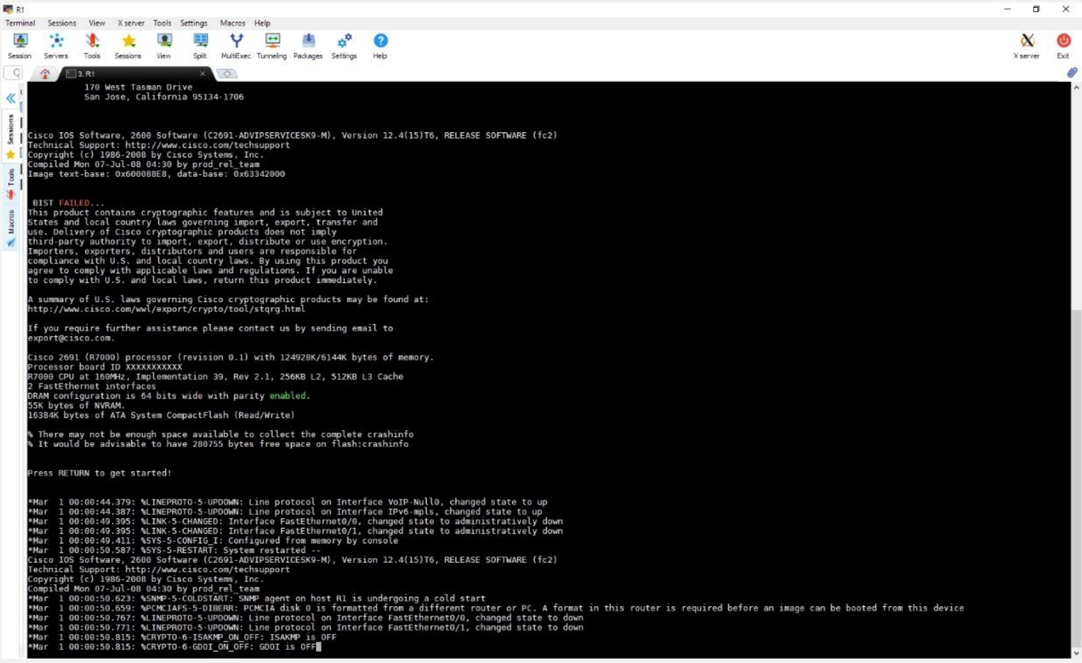
4. Start the MobaXTerm software and click the “Start local terminal” button – this will open the terminal first tab .



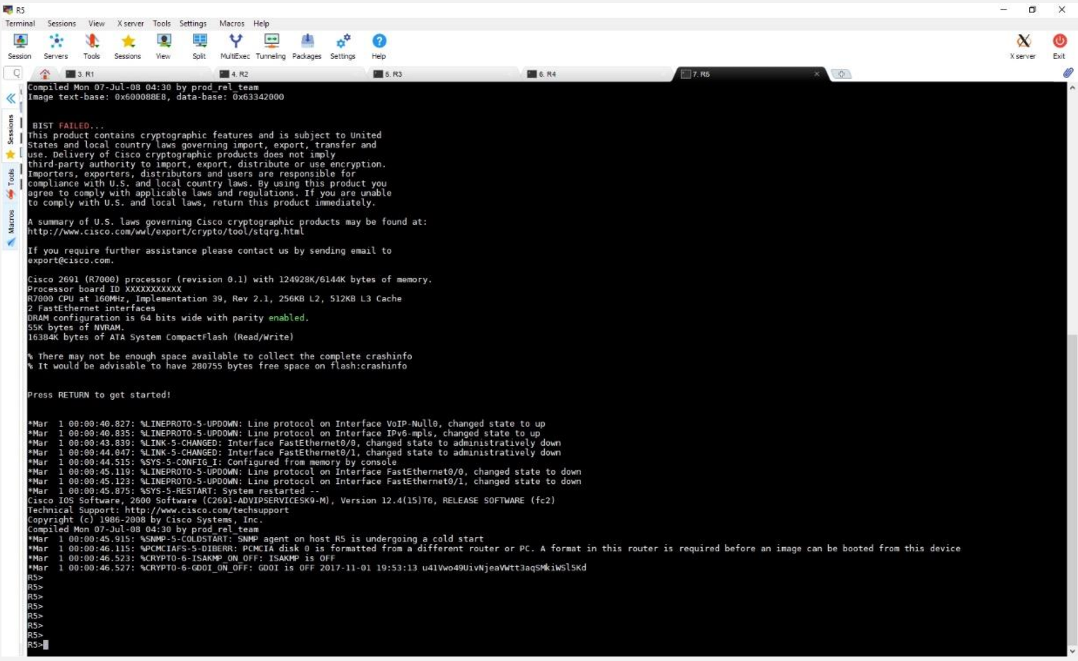
5. In terminal window, run the following command: `telnet -e# localhost 11101`



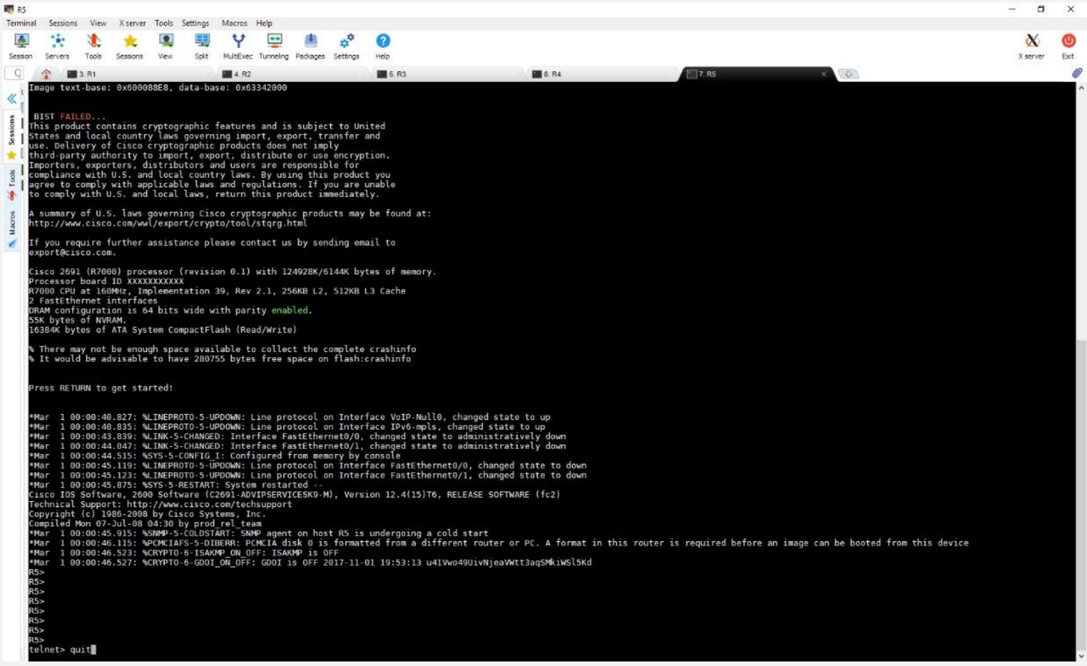
This should provide access to the console of router R1 – you should see the boot process of the router:



6. Open the second tab by clicking on the "+" sign on the right of the first tab. This will open the second terminal tab. After typing the following command: `telnet -e# localhost 11102` you should see the console of router R2.
   Repeat this step, opening three more tabs for routers: R3, R4 and R5, changing only the port number in `telnet` command to 11103, 11104 and 11105, respectively.



7. After the routers end the booting process (typically 1-2 min), the Cisco IOS command line interface will be available on each tab to configure the routers according to the requirements of the exercise.



If you want to close any of the terminal sessions, the best way is to press # (the "escape" sign defined for the telnet session in invoking command) and type "quit" after the "telnet" prompt appears.

Important remarks:

- To save all the text appearing on the terminal tab, select the "Terminal/Save terminal text" from the main menu. You can save the terminal text even after the terminal session is torn down, provided that the session tab is still opened. <u>Note that you must do such save for each tab (router console) separately at the end of the exercise, as it is required to attach the saved files to the report archive.</u>
- After finishing the exercise, close the terminal sessions and power off the virtual machine selecting the "Machine/Close/ACPI Shutdown" from the VirtualBox menu.
- You can save the state of the VM using "Machine/Pause". Even after closing the VirtualBox, the VM state can be restored afterwards. However, the opened terminal sessions will be closed.
- <u>If you close (power off) the VM, at the next power-on all virtual routers will be fully reset to base configurations</u> (no routing protocol enabled, no IP addresses assigned to interfaces etc.).

4. LAB EXERCISE

The topology of the network emulated in the virtual environment is shown in Figure 1. All emulated routers are Cisco 3600. The experience of configuring the routers running within the virtual environment is indistinguishable from configuration of actual devices via typical ssh console access.

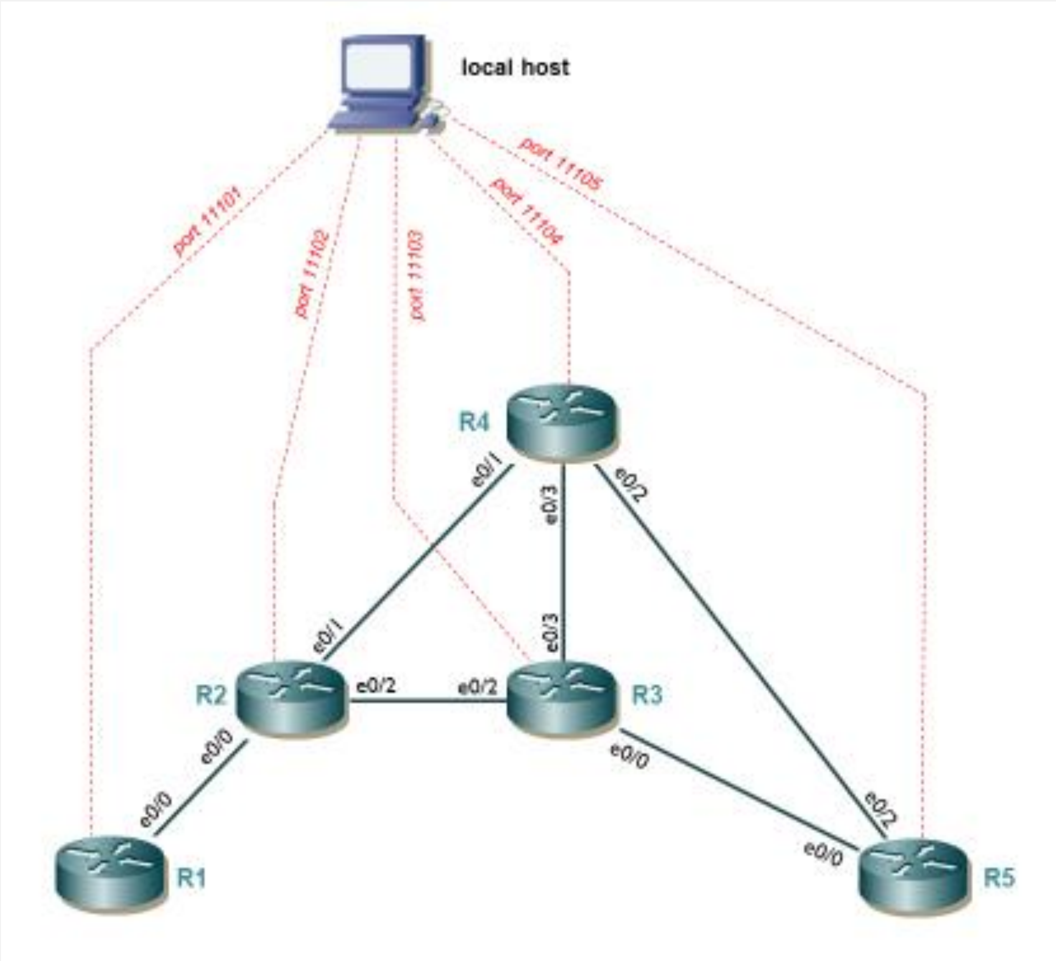


Figure 1. Router Lab configuration

The lab exercise is divided into two main parts:

- basic MPLS configuration using LDP protocol,
- traffic engineering with MPLS using RSVP-TE protocol.

The remainder of this section covers all practical tasks that constitute the scope of this exercise.

A. CONFIGURATION OF IP ADRESSES OF ROUTERS AND INTERFACES

Start this lab assignment with allocating appropriate IP addresses to all routers' interfaces, in accordance with the lab topology presented in Fig. 1, and the information provided in Table 1. Remember that each router interface must have a unique IP address on the subnet that it belongs to. Remember to configure loopback interfaces.

Note: The MPLS LSRs must have (up) Loopback interfaces with an address mask of 32 bits and these interfaces must be reachable with the global IP routing table.

Subnet	Subnet address
R1-R2	10.0.12.0/30
R2-R4	10.0.24.0/30
R3-R4	10.0.34.0/30
R3-R5	10.0.35.0/30
R4-R5	10.0.45.0/30
R2-R3	10.0.23.0/30
Router	**Loopback Address**
R1	1.1.1.1/32
R2	2.2.2.2/32
R3	3.3.3.3/32
R4	4.4.4.4/32
R5	5.5.5.5/32

Table 1. Addressing plan

Configure IP addresses of all interfaces. Use CDP and check your configuration (use the following commands):

```
#show cdp neighbour,
#show cdp neighbour <interface> detail
```

Place the details of the addressing plan in the report.

B. RUNNING OSPF

Before you can configure MPLS you must have routing protocol (OSPF in our case) up and running in the network. Enabling OSPF is required to compute routing tables of the different LSRs (Label Switching Routers). A label distribution protocol (LDP) advertises the bindings between routes and labels. These bindings are then checked against the routing table. If the route (prefix/mask and next hop) learned from the LDP matches the route learned from OSPF in the routing table, an entry is created in the Label Forwarding Information Base (LFIB) on the LSR.

In the config mode, initiate OSPF on every router, and add interfaces (including loopbacks) to the OSPF:

```
#router ospf 1
#network <X.X.X.X> <Y.Y.Y.Y> area 0
X.X.X.X – IP network prefix
Y.Y.Y.Y – wildcard mask
```

All interfaces (except loopbacks) should run OSPF in point-to-point mode – set this mode if necessary (using **`ip ospf network point-to-point`** command in the context of the selected interface).

Check the routing information and routing table in all routers. Ensure that OSPF protocol works properly (ping each router from any other to verify).

Check OSPF metric (link cost) for all network links. You can use **`show ip ospf interface brief`** command, or the more advanced **`show ip ospf interface | include protocol|Cost`**

For the needs of the exercise, all link costs should be set to the value of 10, except the link between the routers R2 and R3, where the cost should be set to 100 (this is to simplify the output of *traceroute* command in case of equal cost paths). Use the following command in the interface configuration mode to change the link costs if required (set the metric on both sides of the link, if necessary).

```
#ip ospf cost <your cost>
```

Verify via *traceroute* command between R1 and R5 what is the effect of cost differentiation on R2-R3 link. *Copy/paste the output of this command into the report and explain the content.*

C. BASIC MPLS CONFIGURATION

Configure MPLS protocol in the network. For better performances enable *ip cef* (use *ip cef distributed* when available) in the general configuration mode. You must enable MPLS both on the router and on the specific interfaces you want to add to MPLS.

Enable MPLS and LDP protocols on each router (in global configuration mode):

#ip cef (enable cef mode if not already enabled)
#mpls ip

To run MPLS on a given interface you must explicitly enable MPLS on that interface:

```
#interface e0/1
#mpls ip
#exit
```

Study the appropriate documentation to extend your knowledge about MPLS configuration if needed.

D. VERIFY MPLS/LDP SETUP

Analyze *show* command output for selected router. Short summary of *show* command functionality that can be used to verify MPLS configuration is provided below.

Information about interfaces added to the MPLS process.
```
#show mpls interfaces
```

IP route for selected destination in the IP routing table:
```
#show ip route <dest>
```

MPLS equivalent of the IP routing table; contains inbound and outbound labels and descriptions of the packets:
```
#show mpls forwarding-table
```

MPLS forwarding table details (if <prefix> is given, only one entry is shown):
```
#show mpls forwarding-table <prefix> detail
```

Copy-paste the outputs of the above commands from router R2 into the report and explain briefly.

E. CHECK PATHS

Now that you have MPLS running in your network you can check if packets are forwarded using the expected paths. You shall compare the MPLS forwarding table of R1 and the result of a *traceroute*.

- From the forwarding table in router R1 find the path to the loopback interface of router R5. Do *a traceroute* to this interface (from R1). *Add the output from this command to the report and explain it. Did the packets take the expected path?*
- Use the appropriate commands to check what label numbers are used on each router for the above IP flow. *Present this information in the report.*

F. CONFIGURE BASIC TRAFFIC ENGINEERING

This part of the exercise explains the basics of implementing traffic engineering (TE) on top of an existing MPLS configuration using OSPF routing protocol. MPLS TE uses an extension to existing OSPF protocol to calculate and establish unidirectional tunnels that are set according to the specific constraints. For setting and maintaining tunnels based on the calculated path the RSVP-TE protocol is used (PATH and RESV messages). The typical RSVP capabilities are extended to allow it to distribute label information.

I. ENABLE TRAFFIC ENGINEERING FOR THE ROUTING PROTOCOL

First, you must enable MPLS TE for a given routing protocol and set the loopback interface as MPLS router id – in the routing protocol configuration mode (use *router ospf 1*) enter the following commands:

```
mpls traffic-eng area 0
mpls traffic-eng router-id loopback0
```

II. ENABLE TRAFFIC ENGINEERING CAPABILITIES

You must enable MPLS traffic engineering both on the router and on each concerned interface. In the global configuration mode as well as in the interface configuration mode enter the following command (hint: enter this command in the global configuration mode first):

```
mpls traffic-eng tunnels
```

III. ENABLE RSVP

This can be done by entering **ip rsvp bandwidth XXX** on each concerned interface. Limit RSVP bandwidth to very low value on the link between router **R2 and R4** (e.g. setup some small value like 64 kbps on both ends). Setup higher bandwidth for other interfaces e.g. 512 kbps. Use the following command:

```
ip rsvp bandwidth 512
```

IV. SET UP TUNNEL TO BE USED FOR TE

Now you can define MPLS tunnels with specific traffic engineering parameters between routers. First set up dynamic tunnel between routers R1 and R5 with specific bandwidth demand. On router R1 and R5 define the tunnel interfaces. To do this enter the interface configuration mode giving the tunnel name as the interface (e.g. **interface tunnel1**). This will create the tunnel interface with the specified name on the router. Next, issue the following commands:

```
mpls ip
ip unnumbered loopback0
no ip directed-broadcast
tunnel destination y.y.y.y
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 5
tunnel mpls traffic-eng bandwidth 256
tunnel mpls traffic-eng path-option 2 dynamic
```

The *tunnel mode mpls traffic–eng* command is mandatory. The *tunnel mpls traffic–eng autoroute announce* command announces the presence of the tunnel by the routing protocol. Note: Do not forget to use *ip* *unnumbered loopback0* for the IP address of the tunnel interfaces. As destination address use the loopback address of the destination router.

- From R1, find the route calculated for the created MPLS traffic engineering tunnel (tunnel1) using command: **`show mpls traffic-eng tunnels`**. *Add the output from this command to the report.*
- Confirm that the packets sent to loopback interface of R5 take the tunnel by analyzing the OSPF and MPLS forwarding tables in the appropriate routers. *Note the route in the report and explain the difference between current route and route from section E, if any.*
- Do *a traceroute* to this interface from R1. *Add the output from this command to the report and explain it.*

Next, restore the RSVP bandwidth limit on the link between router R2 and R4 to normal value (eg. 512 kbps) and set the RSVP bandwidth limit to 64 kbps on the link between R2 and R3.

- From router R1 find the route calculated for *tunnel1* and compare with previous result. *Note the findings in the report and explain the difference.*

V. SET UP EXPLICIT PATH TUNNEL

Make sure that the RSVP bandwidth limit is set to the same value on all links. Shutdown the tunnel1 (using the *configure interface* context). Traceroute the loopback interface of router R5 from R1, observe and explain the result.

Next, Set up explicit MPLS path from R1 to R5, traversing routers R2, R3 and R4. Add second tunnel interface for this path (e.g. **`interface tunnel2`**) and use the following commands :

```
mpls ip
ip unnumbered loopback0
tunnel destination y.y.y.y
tunnel mode mpls traffic-eng
tunnel mpls traffic-eng autoroute announce
tunnel mpls traffic-eng priority 2
tunnel mpls traffic-eng bandwidth 256
tunnel mpls traffic-eng path-option 1 explicit name MyPath
```

The explicit path between R1 and R5 can be defined (in the general config context mode) as follows *(study the CISCO documentation if required).*

```
ip explicit-path name MyPath enable
next-address z.z.z.z
…
```

Check how packets are routed between R1 and R5, *prove that the explicit tunnel is utilized and place this information in the report.*

VI. ADDITIONAL INFO

In addition to relevant *show* commands mentioned earlier in this document, you can use the following commands:

```
show ip ospf mpls traffic-eng link
show ip ospf database opaque-area
```

Execute the above commands on R1, copy/paste the output to the report and explain the content.

G. CLOSING REMARKS

Before closing the lab environment, you are required to do the following:

1. Run “show running config” command on each router console
2. Save the terminal text for each router console and attach the saved files to the zip archive containing the report (see remarks at the end of Section 3).

The reports that do not fulfill the above requirements will be rejected.

Do not use the `reload` command if you want to restore the initial state of the router during the exercise, or you will lose access to the router console. Use the procedure described in the Appendix instead, or reboot the Virtual Machine hosting the IP Lab environment to reset all routers to initial state.

5. APPENDIX – INTRODUCTION TO CISCO IOS

The routers used in the lab exercise run Cisco IOS. There is an abundance of in-depth information related to the IOS in the Internet, so here only a very brief introduction is given.

IOS is the Cisco routers' operating system. You control IOS, and thereby the router, using IOS command line interface, in short – the CLI.

A. COMMAND COMPLETION AND HELP

In most cases it is not necessary to write the IOS commands in full. As soon as there are enough characters so that the CLI can differentiate between commands available in the specific context, you can stop entering characters.

If you are not sure which commands are available you can always enter a **?** sign for help. This is also true if you want to check subcommands.

You can also use the TAB key for command completion.

B. MODES

IOS has several command levels or modes. Depending on mode you can use different commands. When you connect to a router you enter the EXEC mode. The command you will use most in this mode is the show command. In EXEC mode the command prompt ends with a >

R1>

To be able to control the router you must change the mode to PRIVILEGED. You can do this by entering the enable command in EXEC mode. In the PRIVILEGED mode the command prompt ends with a #:

R1#

You can return to EXEC mode from PRIVILEGED mode with the command exit.

Another mode is the CONFIG mode (see subsection D).

C. CONFIGURATION

The router has two configuration storages. The first one is the ***startup-config***. This configuration is stored in non-volatile memory and is read into ***running-config*** memory when the router starts up or reboots.

The ***running-config*** memory contains is the configuration that is used when the router is up and running. When you are in CONFIG mode and enter configuration commands you change the ***running-config*** immediately, thereby changing the behavior of the router. You can copy the ***running-config*** to ***startup-config*** memory by using

copy running-config startup-config

Doing so at the beginning of the exercise will create the ***startup-config*** and allow reverting to the initial state by typing:

configure replace nvram:startup-config

The following should also work:

copy startup-config running-config

*Note! Do not use the **reload** command to revert to the default configuration as you will lose access to the router console afterwards.*

D. THE „CONFIGURE" COMMAND

To enter CONFIG mode, you issue the ***config*** command. This command takes parameters, and in our case, you want to enter configuration commands from the terminal. So, the command should be ***configure terminal*** or in short just ***conf t.***

Once in CONFIG mode remember that each configuration command you enter is activated immediately. It is easy to cut off the branch you are sitting on, then. In our lab, though, you are accessing the router via the console port which is very hard to shut down.

Each function in the router can be set or unset. To set a function you just use the specific configuration command. To unset a function, you write **no** in front of the same command. All functions have a default status, for most of them this state is unset. The default state is not printed in the configuration listing. So those few commands that have the state set as the default state, you will not see in the configuration print out. They will only be listed if you have unset them, that is they will show in the listing with a no in front of them and will be unseen again if you activate that function.

CONFIG mode has several sub modes, for instance the interface configuration sub mode. You enter this mode by typing the interface configuration command:

(config)# interface ethernet 0/0

or in short just

(config)# in e0/0

In configure interface sub mode you can assign the interface an IP address. Use the command ***ip address <ip address> <mask>***. In this sub mode you can also open and close individual interfaces. To close an interface, use the ***shutdown*** command. And in consequence of what was said above, you open an interface with the command ***no shutdown***. You can also create and delete virtual interfaces, so called loopback interfaces, in the configure interface sub mode. To create a new loopback interface just type the configure command ***interface loopback <interface-number>.*** You can exit from CONFIG mode or any sub mode to PRIVILEGED mode by typing ***ctrl'Z***. To exit from a sub mode or from the CONFIG mode use the ***exit*** command.

E. THE „DEBUG" COMMAND

Another nifty command is the **debug** command. In general, it is dangerous to use, since by issuing this command you might end up in a situation where all packets going through the router are displayed on the console terminal. This may have severe impact on the router throughput. Do not use this command in the lab until it is explicitly stated in the lab manual or it is absolutely necessary for troubleshooting. You can debug nearly anything you want, from each single IP-packet to routing announcements sent between the routers. To see the output from the debugging you must direct it to the terminal console that you are connected to. Use the ***terminal monitor*** command. To turn on debugging you issue command ***debug <parameter …>***. To turn it off it is often best to use ***no debug all***.

F. THE "SHOW" COMMAND

The ***show*** command is the one that you will use the most. All parameters of the router can be inspected with this command. Here are some typical uses of this command that you will need.

show running-config	With this command you inspect the running-configuration. It must be given in PRIVILEDGED mode.
show interface	With this command you inspect the status of an interface. You can enter an interface name if you don't want to list them all.
show IP interface brief	With this command you get a list of all the router's interfaces with IP addresses.
show IP protocol	This command gives you information on parameters and status of routing processes running on the router.
show IP route	You show the router's current forwarding table with this command. If you want to see a routing table for the one of perhaps several routing protocols, use the command ***show ip route <routing protocol>***.
show CDP neighbor	Cisco Discovery Protocol is Cisco's proprietary protocol used to exchange information between Cisco equipment. It allows retrieving information on which neighbors are connected to a device, and some basic information about them. This command is a good way to check your connections. If you add the parameter ***detailed*** as a suffix to this command, you will get a lot of information.

G. THE "PING" AND "TRACEROUTE" COMMANDS

Both ***ping*** and ***traceroute*** are available tools in IOS. In their normal form they take the remote host as parameter. Example:

ping 192.168.101.10
traceroute 192.168.7.17

Since a router has several interfaces, i.e. more than one, there is a minor issue here: Which of the several addresses of a router should be used as a source address?

All functions that make use of IP packets, including ***ping*** and ***traceroute***, use the interface closest to the remote host, and therefore the IP address of this interface is used as source address. In our lab we will meet situations where this is not what we want. We might want to check connection with a router loopback interface as source.

In PRIVILEDGED mode you can use the extended version of ***ping*** and ***traceroute***. Just enter the ***ping*** or ***traceroute*** command without any parameters, and you will have several ways to control these commands, like number of packets sent, packet size, and more. Answer ***yes*** to Extended commands question and you will have the possibility to declare source interface or source IP address. In the latter case the IP address must be one of the router's own IP addresses.

H. ADDITIONAL HINTS

- Router interfaces are inactive by default. After proper configuration they must be activated using ***no shutdown*** command.
- Command lines starting from **show** (and ***debug***) work properly only in the PRIVILEGED mode (indicated by the **#** sign next to the router name). However, you can issue these command from CONFIG mode using the ***do*** suffix (e.g. ***do show ip bgp***)
- After configuring a router and routing protocol it is advised to check the validity of configuration by reviewing the config file (***show running-config***) and router's routing table (***show ip route***).