# Solution M4: Network. Software. Services (Debian)

There are many options and combinations to solve the tasks.

A picture containing chart

Description automatically generated

This solution is based on two **Debian** machines – one with two NICs, acting as server/router and a second one with one NIC, acting as client.

*Both machines are imported from templates and the MAC addresses of their network adapters are reinitialized.*

## Machine #1 – Rila

One mandatory preliminary step – set the right host name:

lsauser@debian:~$ **sudo hostnamectl set-hostname rila.lsa.lab**

Then we can adjust one more file to avoid seeing errors like this one when using **sudo**:

***sudo: unable to resolve host rila.lsa.lab: Name or service not known***

Do the actual change by executing:

lsauser@debian:~$ **sudo sed -i s/debian/rila/g /etc/hosts**

And finally, reboot the machine:

lsauser@debian:~$ **sudo reboot**

Once back in, we can continue.

The remaining steps could vary. Here you can see one possible and working combination.

Get information about connections

lsauser@rila:~$ **ip address show**

...

2: **enp0s3**: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000

link/ether 08:00:27:80:a6:f3 brd ff:ff:ff:ff:ff:ff

inet **10.0.2.15/24** brd 10.0.2.255 scope global dynamic enp0s3

...

3: **enp0s8**: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000

link/ether 08:00:27:08:c8:3b brd ff:ff:ff:ff:ff:ff

And configure the second adapter with static IP address by adding the following block at the end:

lsauser@rila:~$ **sudo nano /etc/network/interfaces**

...

**# The secondary network interface**

**allow-hotplug enp0s8**

**iface enp0s8 inet static**

**address 192.168.148.1/24**

...

Save and close the file.

Now apply the configuration *(the following assumes that the interface in question is DOWN):*

lsauser@rila:~$ **sudo ifup enp0s8**

And check the resulting configuration:

lsauser@rila:~$ **ip address show enp0s8**

3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000

link/ether 08:00:27:08:c8:3b brd ff:ff:ff:ff:ff:ff

inet **192.168.148.1/24** brd 192.168.148.255 scope global enp0s8

valid\_lft forever preferred\_lft forever

inet6 fe80::a00:27ff:fe08:c83b/64 scope link

valid\_lft forever preferred\_lft forever

lsauser@rila:~$

Install the **DHCP** service:

lsauser@rila:~$ **sudo apt-get update && sudo apt-get install -y isc-dhcp-server**

...

*We may see some errors, but we can ignore them for now. With our next actions, we will fix them.*

First, configure the interface that we are about to use. Open the respective file for editing:

lsauser@rila:~$ **sudo nano /etc/default/isc-dhcp-server**

And make sure that the last two lines look like this (adjust the name of the adapter to match your case):

**INTERFACESv4="enp0s8"**

INTERFACESv6=""

Save and close the file.

Then, open the main configuration file for editing:

lsauser@rila:~$ **sudo nano /etc/dhcp/dhcpd.conf**

And make sure that it has the following block (it will be best to remove everything and enter the lines below):

**option domain-name "lsa.lab";**

**option domain-name-servers 8.8.8.8;**

**default-lease-time 600;**

**max-lease-time 7200;**

**subnet 192.168.148.0 netmask 255.255.255.0 {**

**range 192.168.148.50 192.168.148.59;**

**option routers 192.168.148.1;**

**option broadcast-address 192.168.148.255;**

**}**

Then test the configuration with:

lsauser@rila:~$ **sudo dhcpd -t**

Internet Systems Consortium DHCP Server 4.4.3-P1

Copyright 2004-2022 Internet Systems Consortium.

All rights reserved.

For info, please visit https://www.isc.org/software/dhcp/

Config file: /etc/dhcp/dhcpd.conf

Database file: /var/lib/dhcp/dhcpd.leases

PID file: /var/run/dhcpd.pid

And finally, restart the service and check its status:

lsauser@rila:~$ **sudo systemctl restart isc-dhcp-server**

lsauser@rila:~$ **systemctl status isc-dhcp-server**

...

All should be “green” and the service should be running without any errors.

**SSH** is usually installed during the installation process of **Debian**. Let’s check its status:

lsauser@rila:~$ **systemctl status ssh**

● ssh.service - OpenBSD Secure Shell server

Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)

Active: active (running) since Fri 2024-10-04 13:15:42 EEST; 6min ago

Docs: man:sshd(8)

man:sshd\_config(5)

Main PID: 499 (sshd)

Tasks: 1 (limit: 2307)

Memory: 9.1M

CPU: 126ms

CGroup: /system.slice/ssh.service

└─499 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Warning: some journal files were not opened due to insufficient permissions.

*Note that you can check the status for both* ***ssh*** *and* ***sshd****. The service is available under both names*

Firewall is usually not installed on **Debian**, so we will continue with the steps related to packet forwarding.

Open the respective file for editing:

lsauser@rila:~$ **sudo nano /etc/sysctl.conf**

And make sure that the following line is present and not commented:

**net.ipv4.ip\_forward=1**

Then apply the changes with:

lsauser@rila:~$ **sudo sysctl -p**

We can check if the setting has been updated with:

lsauser@rila:~$ **cat /proc/sys/net/ipv4/ip\_forward**

1

*If the result is* ***1*** *then the change has been applied.*

Install the **iptables** packages

lsauser@rila:~$ **sudo apt-get install iptables iptables-persistent**

*If asked to save any existing rules, reply with* ***No.***

And then add the following NAT rule:

lsauser@rila:~$ **sudo iptables -t nat -A POSTROUTING -s 192.168.148.0/24 -o enp0s3 -j MASQUERADE**

...

And then save the rule:

lsauser@rila:~$ **sudo iptables-save | sudo tee /etc/iptables/rules.v4**

By now, we should be done with the networking part. Our client station should already have its IP address and should have access to the Internet.

Let’s continue with the rest of the steps.

First, make sure that you have the needed binaries. Try to install them just in case:

lsauser@rila:~$ **sudo apt-get install software-properties-common**

Check the instructions at **repos.zahariev.pro** and register the repository. Let’s use the first option

lsauser@rila:~$ **sudo apt-add-repository 'deb https://repos.zahariev.pro/apt stable main'**

...

*A warning is displayed because the repository is not signed. We can ignore this in our case*

Install the required package

lsauser@rila:~$ **sudo apt-get update --allow-insecure-repositories**

...

lsauser@rila:~$ **sudo apt-get install hello-lsa**

Check that the package is installed and working

**lsauser@rila:~$ hello-lsa**

**\_ \_ \_ \_ \_ \_\_\_\_ \_**

**| | | | \_\_\_| | | \_\_\_ | | / \_\_\_| / \**

**| |\_| |/ \_ \ | |/ \_ \ | | \\_\_\_ \ / \_ \**

**| \_ | \_\_/ | | (\_) | | |\_\_\_ \_\_\_) / \_\_\_ \**

**|\_| |\_|\\_\_\_|\_|\_|\\_\_\_/ |\_\_\_\_\_|\_\_\_\_/\_/ \\_\**

**lsauser@rila:~$**

That is, it. We are done here.

## Machine #2 – Pirin

One preliminary step – set the right host name and reboot the machine:

lsauser@debian:~$ **sudo hostnamectl set-hostname pirin.lsa.lab**

Then we can adjust one more file to avoid seeing errors like this one when using **sudo**:

***sudo: unable to resolve host pirin.lsa.lab: Name or service not known***

Do the actual change by executing:

lsauser@debian:~$ **sudo sed -i s/debian/pirin/g /etc/hosts**

And finally, reboot the machine:

lsauser@debian:~$ **sudo reboot**

Once back in, we can continue.

Other steps may vary. In the next few lines, you can find one possible working solution.

By default, the network is set up to acquire IP address via **DHCP**.

If we check, now we will see that there is an address acquired via **DHCP**:

lsauser@pirin:~$ **ip a**

...

**inet 192.168.148.50/24**

...

Let’s test that we can reach resources on the local network and Internet

lsauser@pirin:~$ **ping -c 4 192.168.148.1**

...

lsauser@pirin:~$ **ping -c 4 8.8.8.8**

...

Create a new user with the required attributes

lsauser@pirin:~$ **sudo useradd -m -c 'Homework User' homework**

...

Set the requested password

lsauser@pirin:~$ **sudo passwd homework**

...

Finally, make it part of the **sudoers** users by adding it to the **sudo** group:

lsauser@pirin:~$ **sudo usermod -aG sudo homework**

We are done with this.

**SSH** is usually installed during the installation process of **Debian**.

Then, we must change the default configuration and switch to port **50022**

lsauser@pirin:~$ **sudo nano /etc/ssh/sshd\_config**

...

**Port 50022**

...

Once done, we must restart the service and check its status

lsauser@pirin:~$ **sudo systemctl restart sshd**

lsauser@pirin:~$ **systemctl status sshd**

...

That is, it. We are done here as well.