INR Lab Assignment:
Linux containers and IPv6 *

Creating the LXD Xen image

> Task 1. Create a new Xen image as per Lab 1, running Ubuntu 17.04 Zesty. It must run Zesty, as pogo is currently not available on a lower version. Make sure your virtual machine can be reached using ssh by installing the relevant packages. Be generous with the amount of RAM (4GB+). We will configure this image to run pogo, and we will use this same image for future INR lab exercises.Hint: Extract the zesty configuration from the xen-tools 4.7-1 package. The pogo gitlab page describes “https://gitlab.os3.nl/Networking/pogo”

Answer:

Since I don't have the xen-tools 4.7-1 package, I will install it:

```
kotaiba@bristol:~$ wget
https://xen-tools.org/software/xen-tools/xen-tools_4.7-1_all.deb
kotaiba@bristol:~$ sudo dpkg -i xen-tools_4.7-1_all.deb
```

Now, let's Create the image:

```
kotaiba@bristol:~$ sudo xen-create-image --hostname=Guest-Z --dist=zesty --
size=10Gb --swap=1024Mb --lvm=VolumeGroupXen --fs=ext3 --vcpus=2 --
ip=145.100.108.83 --netmask=255.255.255.240 --broadcast=145.100.108.95 --
memory=4096Mb --gateway=145.100.108.81 --password=********
```

Results:

```
General Information
--------------------
Hostname       :  Guest-Z
Distribution   :  zesty
Mirror         :  http://archive.ubuntu.com/ubuntu
Partitions     :  swap            1024Mb (swap)
                  /               10Gb  (ext3)
Image type     :  full
Memory size    :  4096Mb
Bootloader     :  pygrub

Networking Information
----------------------
```

```
IP Address 1   : 145.100.108.83 [MAC: 00:16:3E:07:4F:F9]
Netmask        : 255.255.255.240
Broadcast      : 145.100.108.95
Gateway        : 145.100.108.81


Creating swap on /dev/VolumeGroupXen/Guest-Z-swap
Done

Creating ext3 filesystem on /dev/VolumeGroupXen/Guest-Z-disk
Done
Installation method: debootstrap


Done

Running hooks
Done

No role scripts were specified.  Skipping

Creating Xen configuration file
Done

No role scripts were specified.  Skipping
Setting up root password
Generating a password for the new guest.
All done


Logfile produced at:
     /var/log/xen-tools/Guest-Z.log

Installation Summary
---------------------
Hostname        :  Guest-Z
Distribution    :  zesty
MAC Address     :  00:16:3E:07:4F:F9
IP Address(es)  :  145.100.108.83
SSH Fingerprint :  SHA256:lrFkbpkSi/Mxbl+0iSbujOMKCJOmzm2pzDFzehtNnYU (DSA)
SSH Fingerprint :  SHA256:fS6HkNF1YBg7iuKkUjl/7WrRat2SRdVuHjAuGx1eoj4
(ECDSA)
SSH Fingerprint :  SHA256:EExEUqTq8ZwgmpsKW8o877uyCIKXflU7yQFzFyXjmAI
(ED25519)
SSH Fingerprint :  SHA256:KCMGWBMcw30v3JuecLu3NqlYwfg6ZYWPiQjyE1ct3VE (RSA)
Root Password   :  ********
```

Now, Let's create it:

```
kotaiba@bristol:/etc/xen$ sudo xl create Guest-Z.cfg
```

```
Parsing config from Guest-Z.cfg
```

Check:

```
kotaiba@bristol:/etc/xen$ sudo xl console Guest-Z
Guest-Z login: root
Password:
dpkg-query: no packages found matching curl
Welcome to Ubuntu 17.04 (GNU/Linux 4.10.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

Check if I installed the zesty distribution:

```
root@Guest-Z:~# lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 17.04
Release:    17.04
Codename:   zesty
```

Now, Guest-Z also reachable through SSH:

Let's create a user:

```
root@Guest-Z:~# adduser kotaiba
Adding user `kotaiba' ...
Adding new group `kotaiba' (1000) ...
Adding new user `kotaiba' (1000) with group `kotaiba' ...
Creating home directory `/home/kotaiba' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for kotaiba
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
```

```
root@Guest-Z:~# usermod -aG sudo kotaiba
```

```
kotaiba@bristol:~$ ssh kotaiba@145.100.108.83
kotaiba@145.100.108.83's password:
Welcome to Ubuntu 17.04 (GNU/Linux 4.10.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Last login: Mon Nov  6 13:34:32 2017 from 145.100.108.81
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

kotaiba@Guest-Z:~$
```

Before I proceed, I already fixed the source.list (I replaced the old one with the one from Mick gitlab).

But also it's not working, so I have now to fix the DNS issues, so I added to my /etc/network/interfaces file the OS3 DNS servers:

```
dns-nameservers 145.100.96.11 145.100.96.22

Now, restart:
root@Guest-Z:~# service networking restart
```

So, now I can install THE BEST TEXT EDITOR IN THE 21 CENTURY NANOO :D

Source:

1- https://xen-tools.org/software/xen-tools/releases.html

Task 2. Once the machine is up and running, install the screen package and use ssh-copy id to enable ssh key login to this machine from your workstation. Start a screen session. Try to get familiarized with the screen keyboard commands. What does screen -ls do ? Hint: “http://www.linuxjournal.com/ article/6340” is a good reference for screen beginners.

Answer:

Install Screen package:

```
kotaiba@Guest-Z:~$ apt-get install screen
```

Test it:

```
kotaiba@Guest-Z:~$ screen -ls
No Sockets found in /run/screen/S-kotaiba.

kotaiba@Guest-Z:~$ screen

kotaiba@Guest-Z:~$ screen -ls
There is a screen on:
        5799.pts-1.Guest-Z      (11/06/2017 01:48:35 PM)        (Attached)
1 Socket in /run/screen/S-kotaiba.
```

What does screen -ls do ? Print a list of session identification strings (usually of the form pid.tty.host; Sessions marked ‘detached’ can be resumed with screen -r. Those marked ‘attached’ are running and have a controlling terminal. If the session runs in multiuser mode, it is marked ‘multi’. Sessions marked as ‘unreachable’ either live on a different host or are dead. An unreachable session is considered dead, when its name matches either the name of the local host, or the specified parameter, if any.

Then, let's generate RSA keys To get the SSH login:

```
kotaiba@bristol:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kotaiba/.ssh/id_rsa): myKey
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in myKey.
Your public key has been saved in myKey.pub.
The key fingerprint is:
SHA256:+fUUHq6/DeWSLw1llNfKrEnIOhkRVQEhcibDAzM9TCg kotaiba@bristol
The key's randomart image is:
+---[RSA 2048]----+
|    +B= =o++o.  o|
|  E .o**..     .+|
|   .   o o . ooo.|
|        ..o .o+oo|
|        S+ ..o+o.|
|        +. .o+.+ |
|         .. . =o.|
|             ..=.|
|              ooo|
+----[SHA256]-----+

kotaiba@bristol:~$ ls
capture.pca  cold_bristol  cold_oxford  myKey  myKey.pub  xen-
tools_4.7-1_all.deb
kotaiba@bristol:~$ mv myKey* .ssh/
kotaiba@bristol:~$ cd .ssh/
kotaiba@bristol:~/.ssh$ ls
known_hosts  myKey  myKey.pub
kotaiba@bristol:~/.ssh$ ssh-copy-id -i myKey kotaiba@145.100.108.83
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "myKey.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to
```

```
filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys
kotaiba@145.100.108.83's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'kotaiba@145.100.108.83'"
and check to make sure that only the key(s) you wanted were added.
```

Now, let's test the SSH connection using my KEY:

```
kotaiba@bristol:~$ ssh kotaiba@145.100.108.83 -i .ssh/myKey
Enter passphrase for key '.ssh/myKey':
Welcome to Ubuntu 17.04 (GNU/Linux 4.10.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Last login: Mon Nov  6 14:29:11 2017 from 145.100.108.81
kotaiba@Guest-Z:~$
```

Source:

1- https://www.ssh.com/ssh/copy-id, 2- https://www.gnu.org/software/screen/manual/screen.html

Creating a container

Task 3. On the virtual machine also install the lxd package, this will install a current lxd environment. Also install the net-utils, tcpdump and openvswitch-switch packages.

Answer:

To install the required packages:

```
kotaiba@Guest-Z:~$ sudo apt-get install git lxd lxc python
kotaiba@Guest-Z:~$ sudo apt-get install bridge-utils tcpdump screen
openvswitch-switch net-tools
```

Task 4. Install git and python and clone the git repository from https://gitlab.os3.nl/Networking/pogo.git. In the directory you will find the create-base-container.sh script that upon execution will create a container named ogobase. Open the create-base-container.sh with a text editor and try to understand what it does. Execute the script as root. Once the script finishes check that the container has been created using the lxc command. Do not dive into the rest of the scripts for now!

Answer:

I already installed git and python in the previous step.

Install pogo from git:

```
kotaiba@Guest-Z:~$ git clone https://gitlab.os3.nl/Networking/pogo.git
Cloning into 'pogo'...
remote: Counting objects: 80, done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 80 (delta 37), reused 44 (delta 22)
Unpacking objects: 100% (80/80), done.
kotaiba@Guest-Z:~$ cd pogo/
kotaiba@Guest-Z:~/pogo$ sudo ./create-base-container.sh
```

Check if the container installed:

```
kotaiba@Guest-Z:~/pogo$ sudo lxc ls
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

The create-base-container.sh do the following:

Check for root access, check if lxd initialisation. If this is not it will do it. Then check for the base container and create it if it does not exist. Then it creates directory for this container, it also set the default dns (in order to install the listed package in the code), but when it done it removes resolver.conf

Task 5. a. Clone the basecontainer using the lxc toolset and name this lx1. Start this the container and check that everything works.

Answer:

Clone and name it lx1

```
kotaiba@Guest-Z:~/pogo$ sudo lxc copy ogobase lx1
```

```
kotaiba@Guest-Z:~/pogo$ sudo lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| lx1     | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

Start the container:

```
kotaiba@Guest-Z:~/pogo$ sudo lxc start lx1

kotaiba@Guest-Z:~/pogo$ sudo lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| lx1     | RUNNING |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

Test if everything is works:

```
kotaiba@Guest-Z:~/pogo$ sudo lxc exec lx1 bash
root@lx1:~#
```

Task 5. b. Stop and delete the lx1 container using the lxc command from the host machine.

Answer:

List, stop, and delete:

```

kotaiba@Guest-Z:~/pogo$ sudo lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| lx1     | RUNNING |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+

kotaiba@Guest-Z:~/pogo$ sudo lxc stop lx1

kotaiba@Guest-Z:~/pogo$ sudo lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| lx1     | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+

kotaiba@Guest-Z:~/pogo$ sudo lxc delete lx1

kotaiba@Guest-Z:~/pogo$ sudo lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

Task 5. c. Investigate what other functionalities lxc provides, and briefly describe each function in your log.

Answer:

```
kotaiba@Guest-Z:~/pogo$ sudo lxc
Usage: lxc <command> [options]

This is the LXD command line client.

All of LXD's features can be driven through the various commands below.
For help with any of those, simply call them with --help.

Commands:
  config          Change container or server configuration options
  copy            Copy containers within or in between LXD instances
  delete          Delete containers and snapshots
  exec            Execute commands in containers
  file            Manage files in containers
  image           Manipulate container images
  info            Show container or server information
  launch          Create and start containers from images
  list            List the existing containers
  move            Move containers within or in between LXD instances
  network         Manage and attach containers to networks
  profile         Manage container configuration profiles
  publish         Publish containers as images
  remote          Manage the list of remote LXD servers
  restart         Restart containers
  restore         Restore containers from snapshots
  snapshot        Create container snapshots
  start           Start containers
  stop              Stop containers
  storage           Manage storage pools and volumes

Options:
  --all             Print less common commands
  --debug           Print debug information
  --verbose         Print verbose information
  --version         Show client version

Environment:
  LXD_CONF          Path to an alternate client configuration directory
  LXD_DIR           Path to an alternate server directory
```

Setting up a simple IPv4 network

Task 6. What nictypes are available for Linux containers and how do they work?

lxc.network.type: specify what kind of network virtualization to be used for the container.

none: will cause the container to share the host's networknamespace. This means the host network devices are usable in the container. It also means that if both the container and host have upstart as init, 'halt' in a container (for instance) will shut down the host.

empty: will create only the loopback interface.

veth: a virtual ethernet pair device is created with one side assigned to the container and the other side attached to a bridge specified by the lxc.network.link option. If the bridge is not specified, then the veth pair device will be created but not attached to any bridge. Otherwise, the bridge has to be created on the system before starting the container.

vlan: a vlan interface is linked with the interface specified by the lxc.network.link and assigned to the container. The vlan identifier is specified with the option lxc.network.vlan.id.

macvlan: a macvlan interface is linked with the interface specified by the lxc.network.link and assigned to the container. lxc.network.macvlan.mode specifies the mode the macvlan will use to communicate between different macvlan on the same upper device.

phys: an already existing interface specified by the lxc.network.link is assigned to the container.

Answer:

Source:

1- http://manpages.ubuntu.com/manpages/zesty/man5/lxc.container.conf.5.html

Task 7. Investigate openvswitch-switch and then set up the simple network depicted in Figure 1 (S1 is a openvswitch switch), so that both instances can ping each other via IPv4 successfully. The hosts can be created by cloning the base container.

Hint: Use ovs-vsctl to create a bridge. You can connect your virtual machines to this switch by executing e.g. lxc config device add {HostA} eth66 nic nictype=bridged parent={S1} Remove the switch and cloned containers afterwards.

Answer:

In order to investigate, we can simply:

```
kotaiba@Guest-Z:~/pogo$ ovs-vsctl --help
```

Now, let's add new bridge:

```
kotaiba@Guest-Z:~$ sudo ovs-vsctl add-br s1
kotaiba@Guest-Z:~$ sudo ovs-vsctl list-br
s1
```

Now, Let's create two containers HostA and HostB:

```
root@Guest-Z:/home/kotaiba# lxc list
If this is your first time using LXD, you should also run: lxd init
To start your first container, try: lxc launch ubuntu:16.04

+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+

root@Guest-Z:/home/kotaiba/pogo# lxc copy ogobase HostA
root@Guest-Z:/home/kotaiba/pogo# lxc copy ogobase HostB
root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| HostA   | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| HostB   | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

Now, connect the virtual machines to this switch:

```
root@Guest-Z:/home/kotaiba/pogo# lxc config device add HostA eth0 nic
nictype=bridged parent=s1
Device eth0 added to HostA
root@Guest-Z:/home/kotaiba/pogo# lxc config device add HostB eth0 nic
nictype=bridged parent=s1
Device eth0 added to HostB
```

Now, we need to start the both Hosts in order to add static IP for each host:

```
root@Guest-Z:/home/kotaiba/pogo# lxc start HostA
root@Guest-Z:/home/kotaiba/pogo# lxc start HostB
root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| HostA   | RUNNING |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| HostB   | RUNNING |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
```

Now, Add static IP for HostA:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec HostA bash
root@HostA:~# vi /etc/network/interfaces

#add to the /etc/network/interfaces:

auto eth0
iface eth0 inet static
 address 10.0.0.1
 netmask 255.255.255.0


root@HostA:~# service networking restart
```

Now, Add static IP for HostB:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec HostB bash
root@HostB:~# vi /etc/network/interfaces

#add to the /etc/network/interfaces:

auto eth0
iface eth0 inet static
 address 10.0.0.2
 netmask 255.255.255.0
```

```
root@HostAB:~# service networking restart
```

Test:

```
root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+---------+-----------------+------+------------+-----------+
|  NAME   |  STATE  |      IPV4       | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+-----------------+------+------------+-----------+
| HostA   | RUNNING | 10.0.0.1 (eth0) |      | PERSISTENT | 0         |
+---------+---------+-----------------+------+------------+-----------+
| HostB   | RUNNING | 10.0.0.2 (eth0) |      | PERSISTENT | 0         |
+---------+---------+-----------------+------+------------+-----------+
| ogobase | STOPPED |                 |      | PERSISTENT | 0         |
+---------+---------+-----------------+------+------------+-----------+

root@Guest-Z:/home/kotaiba/pogo# lxc exec HostB ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=1.08 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.099 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.087 ms
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64 time=0.092 ms
^C
--- 10.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3028ms
rtt min/avg/max/mdev = 0.087/0.340/1.085/0.430 ms

root@Guest-Z:/home/kotaiba/pogo# lxc exec HostA ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.300 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.084 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.082 ms
^C
--- 10.0.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2053ms
rtt min/avg/max/mdev = 0.082/0.155/0.300/0.102 ms
```

Now, we need to remove everything:

```
root@Guest-Z:/home/kotaiba/pogo# lxc delete HostA --force

root@Guest-Z:/home/kotaiba/pogo# lxc delete HostB --force

root@Guest-Z:/home/kotaiba/pogo# ovs-vsctl del-br s1

root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
root@Guest-Z:/home/kotaiba/pogo# ovs-vsctl list-br
```

IPv6

Task 8. Using the configurations/other/simple.cfg and configurations/other/bridge.cfg as guidelines, create a config file that replicates the network in Figure 2. Because you will not use any dynamic routing protocol and no bridges are involved, set all the roles of the devices to host. For the IP addressing use the following IP blocks: “ Check at assignment pdf ”

Answer:

Appendix:

```
FName: Kotaiba
LName: Alachkar

IPv4: a = 7, b = 8

IPv4 = 7.8.0.0/17

A-R = first of /23 subnet:

Network:   7.8.0.0/23            00000111.00001000.0000000 0.00000000 (Class
A)
Broadcast: 7.8.1.255             00000111.00001000.0000000 1.11111111
HostMin:   7.8.0.1               00000111.00001000.0000000 0.00000001
HostMax:   7.8.1.254             00000111.00001000.0000000 1.11111110
Hosts/Net: 510

HostA = 7.8.0.1/23
HostR = 7.8.0.2/23


R-B = 42nd of /23 subnet

Network:   7.8.82.0/23           00000111.00001000.0101001 0.00000000 (Class
A)
Broadcast: 7.8.83.255            00000111.00001000.0101001 1.11111111
HostMin:   7.8.82.1              00000111.00001000.0101001 0.00000001
HostMax:   7.8.83.254            00000111.00001000.0101001 1.11111110
Hosts/Net: 510

HostR = 7.8.82.1/23
```

```
HostB = 7.8.82.2/23


IPv6 - 2001:0db8:0x00:0y00::1/58 - where:

x: kotaiba -> 107 111 116 097 105 098 097 013 010 = 754 mod 16 = 2
y: alachkar -> 097 108 097 099 104 107 097 114 013 010 = 846 mod 16 = 14 ( e
in hex)

IPv6 network: 2001:0db8:0200:0e00::1/58

HostA = 2001:0db8:0200:0e00::2/64
HostR = 2001:0db8:0200:0e00::3/64

HostR = 2001:0db8:0200:0e2A::2/64
HostB = 2001:0db8:0200:0e2A::3/64
```

After I edited simple.cnf, it look like this now:

```

############################################################################
####
#GLOBAL CONFIG
############################################################################
####
#simple config that starts up a few linux containers
#  host1 --- (hub0)  ---  host2
#

[global]
#this defines where to store the control files and pcaps
session_path= /tmp
tcpdump_path= /tmp

#number of hubs in  the topology
hubs = 2

############################################################################
####
#HOST CONFIG
############################################################################
####

#section name = device hostname
[host1]
#role can be host, bridge, router
role= host

#interface name = <the broadcast domain id to which the interface is
connected to> , <ipv4 address>,$
#domain ids start from 0! Notice The domain id is mandatory. You can specify
```

```
only ipv4 , only ipv6 o$
# note that ipv6 is enabled in the host only when an ipv6 address is present
here
eth12 = 0,7.8.0.1/23,2001:0db8:0200:0e00::2/64

[host2]
role=host
eth12=0,7.8.0.2/23,2001:0db8:0200:0e00::3/64
eth23=1,7.8.82.1/23,2001:0db8:0200:0e2A::2/64

[host3]
role=host
eth23=1,7.8.82.2/23,2001:0db8:0200:0e2A::3/64
```

Source:

1- http://jodies.de/ipcalc?host=7.8.0.0&mask1=17&mask2=23

Task 9. Create the network environment by using python pogo.py create config.cfg

Answer:

```
root@Guest-Z:/home/kotaiba/pogo# python pogo.py create config.cfg
INFO: Creating ogo-ovs0 switch
INFO: Creating ogo-ovs1 switch
INFO: Creating host host1. This will take some time.
INFO: Creating host host2. This will take some time.
INFO: Creating host host3. This will take some time.
```

List:

```
root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+---------+------+------+------------+-----------+
|  NAME   |  STATE  | IPV4 | IPV6 |    TYPE    | SNAPSHOTS |
+---------+---------+------+------+------------+-----------+
| host1   | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| host2   | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| host3   | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+
| ogobase | STOPPED |      |      | PERSISTENT | 0         |
+---------+---------+------+------+------------+-----------+

root@Guest-Z:/home/kotaiba/pogo# lxc network list
```

```
+----------+----------+---------+---------+
|   NAME   |   TYPE   | MANAGED | USED BY |
+----------+----------+---------+---------+
| eth0     | physical | NO      | 0       |
+----------+----------+---------+---------+
| lxcbr0   | bridge   | NO      | 0       |
+----------+----------+---------+---------+
| ogo-ovs0 | bridge   | NO      | 2       |
+----------+----------+---------+---------+
| ogo-ovs1 | bridge   | NO      | 2       |
+----------+----------+---------+---------+
```

Task 10. Bring up the network using python pogo.py start config.cfg and do the following:

- Inspect the IP configuration (addresses, routing table) for all A,B,R (IPv4 and IPv6)
- Check connectivity between A-R, B-R, A-B over IPv4 and IPv6. For IPv6 use both the link local and the global addresses.
- Add IPv4 and IPv6 static routes on A and B such that there is connectivity between the two. Show A can reach B via IPv4 and IPv6 Hint: do not forget to enable routing on R!

Answer:

Bring up the network using python pogo.py start config.cfg:

```
root@Guest-Z:/home/kotaiba/pogo# python pogo.py start config.cfg
INFO: Starting sniffer on ogo-ovs0 switch
INFO: Starting sniffer on ogo-ovs1 switch
INFO: Starting host host1.
INFO: Starting host host2.
INFO: Starting host host3.
There are screens on:
	4188.host3	(11/06/2017 06:13:24 PM)	(Detached)
	3872.host2	(11/06/2017 06:13:22 PM)	(Detached)
	3526.host1	(11/06/2017 06:13:19 PM)	(Detached)
3 Sockets in /run/screen/S-root.
```

Check:

```
root@Guest-Z:/home/kotaiba/pogo# lxc list
+---------+--------+--------------------------------+---------------------
----------+-----------+----------+
|  NAME   |  STATE  |              IPV4              |              IPV6
|    TYPE    | SNAPSHOTS |
+---------+---------+--------------------------------+----------------------
----------+------------+-----------+
| host1   | RUNNING | 7.8.0.1 (eth12)                | 2001:db8:200:e00::2
(eth12)    | PERSISTENT | 0         |
+---------+---------+--------------------------------+----------------------
----------+------------+-----------+
| host2   | RUNNING | 7.8.82.1 (eth23)               | 2001:db8:200:e2a::2
(eth23)    | PERSISTENT | 0         |
|         |         | 7.8.0.2 (eth12)                | 2001:db8:200:e00::3
(eth12)    |            |           |
+---------+---------+--------------------------------+----------------------
----------+------------+-----------+
| host3   | RUNNING | 7.8.82.2 (eth23)               | 2001:db8:200:e2a::3
(eth23)    | PERSISTENT | 0         |
+---------+---------+--------------------------------+----------------------
----------+------------+-----------+
| ogobase | STOPPED |                                |
| PERSISTENT | 0         |
+---------+---------+--------------------------------+----------------------
----------+------------+-----------+
```

Inspect the IP configuration

I will get the bash of each one and test it.

HostA:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host1 bash
root@host1:~# ifconfig
eth12     Link encap:Ethernet  HWaddr 00:16:3e:06:ce:1e
          inet addr:7.8.0.1  Bcast:7.8.1.255  Mask:255.255.254.0
          inet6 addr: fe80::216:3eff:fe06:ce1e/64 Scope:Link
          inet6 addr: 2001:db8:200:e00::2/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:25 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2038 (2.0 KB)  TX bytes:1172 (1.1 KB)

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:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:260 (260.0 B)  TX bytes:260 (260.0 B)
```

```
root@host1:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
7.8.0.0         *               255.255.254.0   U     0      0        0
eth12
```

HostR:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host2 bash
root@host2:~# ifconfig
eth12     Link encap:Ethernet  HWaddr 00:16:3e:16:75:a5
          inet addr:7.8.0.2  Bcast:7.8.1.255  Mask:255.255.254.0
          inet6 addr: 2001:db8:200:e00::3/64 Scope:Global
          inet6 addr: fe80::216:3eff:fe16:75a5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:20 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1612 (1.6 KB)  TX bytes:1172 (1.1 KB)

eth23     Link encap:Ethernet  HWaddr 00:16:3e:eb:42:79
          inet addr:7.8.82.1  Bcast:7.8.83.255  Mask:255.255.254.0
          inet6 addr: 2001:db8:200:e2a::2/64 Scope:Global
          inet6 addr: fe80::216:3eff:feeb:4279/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:25 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2038 (2.0 KB)  TX bytes:1172 (1.1 KB)

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:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:260 (260.0 B)  TX bytes:260 (260.0 B)

root@host2:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
7.8.0.0         *               255.255.254.0   U     0      0        0
eth12
7.8.82.0        *               255.255.254.0   U     0      0        0
eth23
```

HostB:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host3 bash
root@host3:~# ifconfig
eth23     Link encap:Ethernet  HWaddr 00:16:3e:cb:46:89  
          inet addr:7.8.82.2  Bcast:7.8.83.255  Mask:255.255.254.0
          inet6 addr: fe80::216:3eff:fecb:4689/64 Scope:Link
          inet6 addr: 2001:db8:200:e2a::3/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:19 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000 
          RX bytes:1522 (1.5 KB)  TX bytes:1172 (1.1 KB)

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:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000 
          RX bytes:260 (260.0 B)  TX bytes:260 (260.0 B)

root@host3:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use 
Iface
7.8.82.0        *               255.255.254.0   U     0      0        0 
eth23
```

Check connectivity

HostA:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host1 bash
root@host1:~# ping 7.8.0.2
PING 7.8.0.2 (7.8.0.2) 56(84) bytes of data.
64 bytes from 7.8.0.2: icmp_seq=1 ttl=64 time=1.01 ms
64 bytes from 7.8.0.2: icmp_seq=2 ttl=64 time=0.090 ms
^C
--- 7.8.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.090/0.551/1.013/0.462 ms



root@host1:~# ping6 -I eth12 2001:0db8:0200:0e00::3
PING 2001:0db8:0200:0e00::3(2001:db8:200:e00::3) from 2001:db8:200:e00::2 
eth12: 56 data bytes
64 bytes from 2001:db8:200:e00::3: icmp_seq=1 ttl=64 time=1.26 ms
64 bytes from 2001:db8:200:e00::3: icmp_seq=2 ttl=64 time=0.139 ms
^C
```

```
--- 2001:0db8:0200:0e00::3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.139/0.699/1.260/0.561 ms

root@host1:~# ping6 -I eth12 fe80::216:3eff:fe06:ce1e
PING fe80::216:3eff:fe06:ce1e(fe80::216:3eff:fe06:ce1e) from
fe80::216:3eff:fe06:ce1e eth12: 56 data bytes
64 bytes from fe80::216:3eff:fe06:ce1e: icmp_seq=1 ttl=64 time=0.048 ms
64 bytes from fe80::216:3eff:fe06:ce1e: icmp_seq=2 ttl=64 time=0.081 ms
^C
--- fe80::216:3eff:fe06:ce1e ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.048/0.064/0.081/0.018 ms
```

HostR:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host2 bash

root@host2:~# ping 7.8.0.1
PING 7.8.0.1 (7.8.0.1) 56(84) bytes of data.
64 bytes from 7.8.0.1: icmp_seq=1 ttl=64 time=0.601 ms
^C
--- 7.8.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.601/0.601/0.601/0.000 ms

root@host2:~# ping 7.8.82.2
PING 7.8.82.2 (7.8.82.2) 56(84) bytes of data.
64 bytes from 7.8.82.2: icmp_seq=1 ttl=64 time=0.907 ms
64 bytes from 7.8.82.2: icmp_seq=2 ttl=64 time=0.092 ms
^C
--- 7.8.82.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.092/0.499/0.907/0.408 ms

root@host2:~# ping6 2001:0db8:0200:0e00::2
PING 2001:0db8:0200:0e00::2(2001:db8:200:e00::2) 56 data bytes
64 bytes from 2001:db8:200:e00::2: icmp_seq=1 ttl=64 time=0.900 ms
64 bytes from 2001:db8:200:e00::2: icmp_seq=2 ttl=64 time=0.113 ms
^C
--- 2001:0db8:0200:0e00::2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.113/0.506/0.900/0.394 ms

root@host2:~# ping6 2001:0db8:0200:0e2A::3
PING 2001:0db8:0200:0e2A::3(2001:db8:200:e2a::3) 56 data bytes
64 bytes from 2001:db8:200:e2a::3: icmp_seq=1 ttl=64 time=1.28 ms
64 bytes from 2001:db8:200:e2a::3: icmp_seq=2 ttl=64 time=0.113 ms
^C
--- 2001:0db8:0200:0e2A::3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
```

```
rtt min/avg/max/mdev = 0.113/0.697/1.281/0.584 ms

root@host2:~# ping6 -I eth12 fe80::216:3eff:fe16:75a5
PING fe80::216:3eff:fe16:75a5(fe80::216:3eff:fe16:75a5) from
fe80::216:3eff:fe16:75a5 eth12: 56 data bytes
64 bytes from fe80::216:3eff:fe16:75a5: icmp_seq=1 ttl=64 time=0.202 ms
64 bytes from fe80::216:3eff:fe16:75a5: icmp_seq=2 ttl=64 time=0.089 ms
^C
--- fe80::216:3eff:fe16:75a5 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1017ms
rtt min/avg/max/mdev = 0.089/0.145/0.202/0.057 ms
```

HostB:

```
root@Guest-Z:/home/kotaiba/pogo# lxc exec host3 bash

root@host3:~# ping 7.8.82.1
PING 7.8.82.1 (7.8.82.1) 56(84) bytes of data.
64 bytes from 7.8.82.1: icmp_seq=1 ttl=64 time=0.681 ms
^C
--- 7.8.82.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.681/0.681/0.681/0.000 ms

root@host3:~# ping6 2001:0db8:0200:0e2A::2
PING 2001:0db8:0200:0e2A::2(2001:db8:200:e2a::2) 56 data bytes
64 bytes from 2001:db8:200:e2a::2: icmp_seq=1 ttl=64 time=0.726 ms
64 bytes from 2001:db8:200:e2a::2: icmp_seq=2 ttl=64 time=0.108 ms
^C
--- 2001:0db8:0200:0e2A::2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1008ms
rtt min/avg/max/mdev = 0.108/0.417/0.726/0.309 ms

root@host3:~# ping6 -I eth23 fe80::216:3eff:fecb:4689
PING fe80::216:3eff:fecb:4689(fe80::216:3eff:fecb:4689) from
fe80::216:3eff:fecb:4689 eth23: 56 data bytes
64 bytes from fe80::216:3eff:fecb:4689: icmp_seq=1 ttl=64 time=0.216 ms
64 bytes from fe80::216:3eff:fecb:4689: icmp_seq=2 ttl=64 time=0.081 ms
^C
--- fe80::216:3eff:fecb:4689 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1031ms
rtt min/avg/max/mdev = 0.081/0.148/0.216/0.068 ms
```

IPv4 and IPv6 static routes

- **IPv4**

Before I start let me enable Ipv4-forwarding:

```
root@host2:~# sysctl net.ipv4.ip_forward=1
```

```
net.ipv4.ip_forward = 1
```

HostA:

```
root@host1:~# route add -net 7.8.82.0/23 gw 7.8.0.2 eth12
root@host1:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
7.8.0.0         *               255.255.254.0   U     0      0        0
eth12
7.8.82.0        7.8.0.2         255.255.254.0   UG    0      0        0
eth12
7.8.82.0        *               255.255.254.0   U     0      0        0
eth12
```

HostB:

```
root@host3:~# route add -net 7.8.0.0/23 gw 7.8.82.1 eth23
root@host3:~# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use
Iface
7.8.0.0         7.8.82.1        255.255.254.0   UG    0      0        0
eth23
7.8.0.0         *               255.255.254.0   U     0      0        0
eth23
7.8.82.0        *               255.255.254.0   U     0      0        0
eth23
```

Test IPv4, Ping from A to B:

```
root@host1:~# ping 7.8.82.2
PING 7.8.82.2 (7.8.82.2) 56(84) bytes of data.
64 bytes from 7.8.82.2: icmp_seq=1 ttl=63 time=0.531 ms
64 bytes from 7.8.82.2: icmp_seq=2 ttl=63 time=0.114 ms
64 bytes from 7.8.82.2: icmp_seq=3 ttl=63 time=0.099 ms
^C
--- 7.8.82.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.099/0.248/0.531/0.200 ms
```

- **IPv6**

let me enable Ipv6-forwarding:

```
root@host2:~# sysctl -w net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1
```

HostA:

```
root@host1:~# route add -A inet6 2001:db8:200:e2a::3/64 gw
2001:db8:200:e00::3 eth12
```

HostB:

```
root@host3:~# route add -A inet6 2001:db8:200:e00::2/64 gw
2001:db8:200:e2a::2 eth23
```

Test, ping from B to A:

```
root@host3:~# ping6 2001:db8:200:e00::2
PING 2001:db8:200:e00::2(2001:db8:200:e00::2) 56 data bytes
64 bytes from 2001:db8:200:e00::2: icmp_seq=1 ttl=63 time=1.02 ms
64 bytes from 2001:db8:200:e00::2: icmp_seq=2 ttl=63 time=0.135 ms
^C
--- 2001:db8:200:e00::2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.135/0.580/1.025/0.445 ms
```

Auto configuration

Task 11. Stop and start your network using pogo.

Answer:

Stop:

```
root@Guest-Z:/home/kotaiba/pogo# python pogo.py stop config.cfg
INFO: Stopping sniffer on ogo-ovs0 switch
INFO: Stopping sniffer on ogo-ovs1 switch
INFO: Stopping host host1
INFO: Stopping host host2
INFO: Stopping host host3
```

Start:

```
root@Guest-Z:/home/kotaiba/pogo# python pogo.py start config.cfg
INFO: Starting sniffer on ogo-ovs0 switch
INFO: Starting sniffer on ogo-ovs1 switch
INFO: Starting host host1.
INFO: Starting host host2.
INFO: Starting host host3.
There are screens on:
    12291.host3    (11/06/17 21:43:00)    (Detached)
    11684.host2    (11/06/17 21:42:59)    (Detached)
    11150.host1    (11/06/17 21:42:57)    (Detached)
3 Sockets in /run/screen/S-root.
```

Task 12. On R configure and enable the radvd daemon, using the IP blocks mentioned in the previous task. Explain all the configuration parameters used. As with the previous task, inspect the IP configuration, do a connectivity check and explain the differences.

Answer:

First, what is radvd daemon, it is “The Router Advertisement Daemon (radvd) is an open source software product that implements link-local advertisements of IPv6 router addresses and IPv6 routing prefixes using the Neighbor Discovery Protocol (NDP) ”

Before I enable the radvd daemon on hostR, first I need to create a configuration file /etc/radvd.conf and add the following to it:

```

interface eth12
{
   AdvSendAdvert on;
   prefix 2001:0db8:0200:0e00::/64
   {
   };
};
interface eth23
{
   AdvSendAdvert on;
   prefix 2001:0db8:0200:0e2a::/64
   {
   };
};
```

As we see above in this configuration we make the radvd daemon to use a specific prefix on the interfaces. The AdvSendAdvert option is “A flag indicating whether or not the router sends periodic router advertisements and responds to router solicitations.”

CREDITS TO ARNO TO HELP ME IN THIS.

Now, we enable and start the radvd daemon:

```
root@Guest-Z:~# lxc exec host2 service radvd start
```

Now, we need to check on Hosts:

HostA

```
root@Guest-Z:/home/kotaiba# lxc exec host1 bash
```

```
root@host1:~# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
20: eth12@if21: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP group default qlen 1000
    link/ether 00:16:3e:06:ce:1e brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 7.8.0.1/23 brd 7.8.1.255 scope global eth12
       valid_lft forever preferred_lft forever
    inet6 2001:db8:200:e00:216:3eff:fe06:ce1e/64 scope global mngtmpaddr
dynamic
       valid_lft 86324sec preferred_lft 14324sec
    inet6 2001:db8:200:e00::2/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::216:3eff:fe06:ce1e/64 scope link
       valid_lft forever preferred_lft forever
root@host1:~# ip -6 route
2001:db8:200:e00::/64 dev eth12  proto kernel  metric 256  pref medium
fe80::/64 dev eth12  proto kernel  metric 256  pref medium
default via fe80::216:3eff:fe16:75a5 dev eth12  proto ra  metric 1024
expires 1708sec hoplimit 64 pref medium
```

HostB

```
root@Guest-Z:/home/kotaiba# lxc exec host3 bash
root@host3:~# ip adress
Object "adress" is unknown, try "ip help".
root@host3:~# ip adresses
Object "adresses" is unknown, try "ip help".
root@host3:~# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
26: eth23@if27: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP group default qlen 1000
    link/ether 00:16:3e:cb:46:89 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 7.8.82.2/23 brd 7.8.83.255 scope global eth23
       valid_lft forever preferred_lft forever
    inet6 2001:db8:200:e2a:216:3eff:fecb:4689/64 scope global mngtmpaddr
dynamic
       valid_lft 86271sec preferred_lft 14271sec
    inet6 2001:db8:200:e2a::3/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::216:3eff:fecb:4689/64 scope link
       valid_lft forever preferred_lft forever
root@host3:~# ip -6 route
2001:db8:200:e2a::/64 dev eth23  proto kernel  metric 256  pref medium
fe80::/64 dev eth23  proto kernel  metric 256  pref medium
default via fe80::216:3eff:feeb:4279 dev eth23  proto ra  metric 1024
expires 1664sec hoplimit 64 pref medium
```

Test A-B:

```
root@Guest-Z:/home/kotaiba# lxc exec host1 bash
root@host1:~# ping6 2001:db8:200:e2a::3
PING 2001:db8:200:e2a::3(2001:db8:200:e2a::3) 56 data bytes
64 bytes from 2001:db8:200:e2a::3: icmp_seq=1 ttl=63 time=1.57 ms
64 bytes from 2001:db8:200:e2a::3: icmp_seq=2 ttl=63 time=0.127 ms
^C
--- 2001:db8:200:e2a::3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.127/0.849/1.571/0.722 ms

root@host1:~# ping 7.8.82.2
connect: Network is unreachable
```

As we notice above the IPv4 connectivity test from HostA to HostB is not working “connect: Network is unreachable”. The possible reason is that we we restarted the VMs networks the static routes are no more configured. However, for the IPv4 conectivity it works since the radvd daemon is working and advertise through the pre-defined prefixes.

Source:

1- https://en.wikipedia.org/wiki/Radvd, 2- https://linux.die.net/man/5/radvd.conf

Task 13. Explain how the IPv6 address received by host A was derived.

Answer:

Host A got its IPv6 address from message type of the neighbor discovery protocol (NDP) since we have router advertisement. The first 64-bits came from the prefix that's configured on the interface (in the radvd.conf). The next 64-bits (interface ID) came from the EUI-64 process:

- The first part of the mac address of the interface is 16-bits represent the Network Interface Controller (NIC).
- according to the EUI-64 specification, FFFE is hardcoded in the address.
- Finally, the Organizationally Unique Identifier (OUI) is used. Where the 7th local or universal bit is set to “1”.

Source:

1- http://www.fredrikholmberg.com/2012/05/ipv6-autoconfiguration-with-slaac-and-ndp-how-does-it-work/, 2- https://howdoesinternetwork.com/2013/slaac-ipv6-stateless-address-autoconfiguration

Task 14. Why wasn’t it necessary to manually add routes ?

Answer:

Because there is something in RA called ICMPv6 (we receive the routes through it). We took it today in lecturer, so to be more specific the type flag is 135.

Source:

1- http://www.fredrikholmberg.com/2012/05/ipv6-autoconfiguration-with-slaac-and-ndp-how-does-it-work/

Task 15. Stop the radvd service and see if the network breaks. Explain why.

Answer:

Stop the radvd service, on HostR:

```
root@host2:~# service radvd stop
```

And let's stop it manually to make sure:

```
root@host2:~# ps -ef | grep radv
root      2850     1  0 12:17 ?        00:00:00 radvd
root      2851  2850  0 12:17 ?        00:00:00 radvd
root      3139  3095  0 12:44 ?        00:00:00 grep --color=auto radv
root@host2:~# pkill -9 radvd
root@host2:~# ps -ef | grep radv
root      3142  3095  0 12:45 ?        00:00:00 grep --color=auto radv
```

Check if network breaks:

The network should break, but its not in my case because I test it immediately after I stop it:

```
root@host1:~# ping6 2001:db8:200:e2a::3
PING 2001:db8:200:e2a::3(2001:db8:200:e2a::3) 56 data bytes
64 bytes from 2001:db8:200:e2a::3: icmp_seq=1 ttl=63 time=0.902 ms
64 bytes from 2001:db8:200:e2a::3: icmp_seq=2 ttl=63 time=0.130 ms
^C
```

```
--- 2001:db8:200:e2a::3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.130/0.516/0.902/0.386 ms

root@host1:~# ip -6 route
2001:db8:200:e00::/64 dev eth12  proto kernel  metric 256  pref medium
fe80::/64 dev eth12  proto kernel  metric 256  pref medium
default via fe80::216:3eff:fe16:75a5 dev eth12  proto ra  metric 1024
expires 1606sec hoplimit 64 pref medium
```

The reason why its still working is that NDP timer “expires 1606sec”. So once a timer reaches zero, the configuration is removed from the client. In this, case if I wait for 1606 second, the network will break.

I will check after 1606 seconds to make sure.

After 1606 seconds :D, let's check again:

On HostA try to ping to HostB:

```
root@host1:~# ip -6 route
2001:db8:200:e00::/64 dev eth12  proto kernel  metric 256  pref medium
fe80::/64 dev eth12  proto kernel  metric 256  pref medium

root@host1:~# ping6 2001:db8:200:e2a::3
connect: Network is unreachable
```

Now, It's not working, so that proof my answer.

Analysis

Task 16. Check the tcpdump path on the host system. It should contain pcap files of the last pogo run. These dumps should contain all the relevant packets regarding auto-configuration.

Answer:

I checked the configuration file on Mick gitlab page, in simple.cfg file (that we already used as structure for my configuration file), it states that:

tcpdump_path= /tmp, so let's check:

```
root@Guest-Z:/tmp# ls
ogo-ovs0.pcap
ogo-ovs1.pcap
```

```
systemd-private-beb18be1cf87468d83daeedbe6d9c0eb-systemd-timesyncd.service-
ynHJQB
```

Let's use tcpdump to read these files:

* **/tmp/ogo-ovs0.pcap file:**

```
root@Guest-Z:/tmp# tcpdump -r /tmp/ogo-ovs0.pcap

reading from file /tmp/ogo-ovs0.pcap, link-type EN10MB (Ethernet)
21:42:57.518459 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:42:58.226686 IP6 :: > ff02::1:ff06:ce1e: ICMP6, neighbor solicitation,
who has fe80::216:3eff:fe06:ce1e, length 32
21:42:58.322598 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:42:58.962388 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:42:58.986504 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
2 group record(s), length 48
21:42:58.986906 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:42:59.118549 IP6 :: > ff02::1:ff00:2: ICMP6, neighbor solicitation, who
has 2001:db8:200:e00::2, length 32
21:42:59.250356 IP6 fe80::216:3eff:fe06:ce1e > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:42:59.250500 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:42:59.602484 IP6 :: > ff02::1:ff16:75a5: ICMP6, neighbor solicitation,
who has fe80::216:3eff:fe16:75a5, length 32
21:42:59.858358 IP6 fe80::216:3eff:fe06:ce1e > ff02::16: HBH ICMP6,
multicast listener report v2, 1 group record(s), length 28
21:42:59.986354 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:43:00.178391 IP6 fe80::216:3eff:fe06:ce1e > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:00.470346 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
2 group record(s), length 48
21:43:00.626391 IP6 fe80::216:3eff:fe16:75a5 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:00.626523 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:01.010340 IP6 fe80::216:3eff:fe16:75a5 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:01.106492 IP6 :: > ff02::1:ff00:3: ICMP6, neighbor solicitation, who
has 2001:db8:200:e00::3, length 32
21:43:01.362906 IP6 fe80::216:3eff:fe16:75a5 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:03.250351 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:04.530428 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
```

```
solicitation, length 16
21:43:11.442411 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:12.210625 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:28.082676 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:28.082713 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:44:00.850623 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:44:02.898635 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:45:06.386629 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:45:10.482726 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:47:33.842732 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
21:47:33.842780 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:51:55.986663 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:52:12.370670 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
22:00:40.274634 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
22:01:45.810740 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
22:15:00.688334 IP6 2001:db8:200:e00::2 > ff02::1:ff00:3: ICMP6, neighbor
solicitation, who has 2001:db8:200:e00::3, length 32
22:15:00.688444 IP6 2001:db8:200:e00::3 > 2001:db8:200:e00::2: ICMP6,
neighbor advertisement, tgt is 2001:db8:200:e00::3, length 32
22:15:00.688504 IP6 2001:db8:200:e00::2 > 2001:db8:200:e00::3: ICMP6, echo
request, seq 1, length 64
22:15:00.688519 IP6 2001:db8:200:e00::3 > 2001:db8:200:e00::2: ICMP6, echo
reply, seq 1, length 64
22:15:05.810424 IP6 fe80::216:3eff:fe16:75a5 > 2001:db8:200:e00::2: ICMP6,
neighbor solicitation, who has 2001:db8:200:e00::2, length 32
22:15:05.810463 IP6 2001:db8:200:e00::2 > fe80::216:3eff:fe16:75a5: ICMP6,
neighbor advertisement, tgt is 2001:db8:200:e00::2, length 24
22:15:10.930428 IP6 fe80::216:3eff:fe06:ce1e > fe80::216:3eff:fe16:75a5:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fe16:75a5, length 32
22:15:10.930467 IP6 fe80::216:3eff:fe16:75a5 > fe80::216:3eff:fe06:ce1e:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fe16:75a5, length 24
22:15:16.050418 IP6 fe80::216:3eff:fe16:75a5 > fe80::216:3eff:fe06:ce1e:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fe06:ce1e, length 32
22:15:16.050461 IP6 fe80::216:3eff:fe06:ce1e > fe80::216:3eff:fe16:75a5:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fe06:ce1e, length 24
22:18:08.850629 IP6 fe80::216:3eff:fe16:75a5 > ip6-allrouters: ICMP6, router
solicitation, length 16
```

```
22:20:19.922655 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
22:21:40.510712 IP6 fe80::216:3eff:fe16:75a5 > ff02::16: HBH ICMP6,
multicast listener report v2, 3 group record(s), length 68
22:21:41.362411 IP6 fe80::216:3eff:fe16:75a5 > ff02::16: HBH ICMP6,
multicast listener report v2, 3 group record(s), length 68
22:57:28.146634 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
23:56:27.090608 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
00:53:14.962864 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
01:58:47.122736 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
02:57:46.066761 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
03:56:45.010700 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
04:55:43.954664 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
05:59:05.042670 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
07:04:37.202635 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
08:01:25.074653 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
08:58:12.946720 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
10:03:45.106750 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
11:07:06.194724 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
12:03:54.066742 IP6 fe80::216:3eff:fe06:ce1e > ip6-allrouters: ICMP6, router
solicitation, length 16
12:17:42.043841 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:17:43.026664 IP6 :: > ff02::1:ff06:ce1e: ICMP6, neighbor solicitation,
who has 2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:17:58.060708 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:18:14.071674 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:23:33.232338 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:27:52.096577 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:27:52.097511 IP6 fe80::216:3eff:fe16:75a5 > ff02::1:ff06:ce1e: ICMP6,
neighbor solicitation, who has 2001:db8:200:e00:216:3eff:fe06:ce1e, length
32
12:27:52.097545 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
fe80::216:3eff:fe16:75a5: ICMP6, neighbor advertisement, tgt is
```

```
2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:27:52.097717 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:27:53.097985 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:27:53.098054 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:31:09.847384 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:36:31.395622 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:36:31.396367 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:36:32.396633 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:36:32.396708 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:36:33.426477 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:36:33.426552 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:36:36.562412 IP6 fe80::216:3eff:fe16:75a5 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, neighbor solicitation, who has
2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:36:36.562467 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
fe80::216:3eff:fe16:75a5: ICMP6, neighbor advertisement, tgt is
2001:db8:200:e00:216:3eff:fe06:ce1e, length 24
12:36:44.530866 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:36:57.380130 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:36:57.380876 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:36:58.381220 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:36:58.381289 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:36:59.410474 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:36:59.410547 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:39:55.090851 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:39:55.091408 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:39:56.091729 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:39:56.091798 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:40:00.338454 IP6 fe80::216:3eff:fe16:75a5 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, neighbor solicitation, who has
```

```
2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:40:00.338519 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
fe80::216:3eff:fe16:75a5: ICMP6, neighbor advertisement, tgt is
2001:db8:200:e00:216:3eff:fe06:ce1e, length 24
12:42:05.849816 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:43:09.470527 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:43:09.470852 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:43:10.482471 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:43:10.482544 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:43:11.506491 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:43:11.506564 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:43:14.642461 IP6 fe80::216:3eff:fe16:75a5 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, neighbor solicitation, who has
2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:43:14.642526 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
fe80::216:3eff:fe16:75a5: ICMP6, neighbor advertisement, tgt is
2001:db8:200:e00:216:3eff:fe06:ce1e, length 24
12:43:29.028394 IP6 2001:db8:200:e2a:216:3eff:fecb:4689 >
2001:db8:200:e00::2: ICMP6, echo request, seq 1, length 64
12:43:29.028919 IP6 2001:db8:200:e00::2 >
2001:db8:200:e2a:216:3eff:fecb:4689: ICMP6, echo reply, seq 1, length 64
12:43:30.029509 IP6 2001:db8:200:e2a:216:3eff:fecb:4689 >
2001:db8:200:e00::2: ICMP6, echo request, seq 2, length 64
12:43:30.029536 IP6 2001:db8:200:e00::2 >
2001:db8:200:e2a:216:3eff:fecb:4689: ICMP6, echo reply, seq 2, length 64
12:45:48.883509 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:45:48.884184 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:45:49.884432 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:45:49.884501 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:45:54.130442 IP6 fe80::216:3eff:fe16:75a5 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, neighbor solicitation, who has
2001:db8:200:e00:216:3eff:fe06:ce1e, length 32
12:45:54.130478 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
fe80::216:3eff:fe16:75a5: ICMP6, neighbor advertisement, tgt is
2001:db8:200:e00:216:3eff:fe06:ce1e, length 24
```

- **/tmp/ogo-ovs1.pcap file:**

```
root@Guest-Z:/tmp# tcpdump -r /tmp/ogo-ovs1.pcap
```

```
reading from file /tmp/ogo-ovs1.pcap, link-type EN10MB (Ethernet)
21:42:58.986877 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:42:59.346519 IP6 :: > ff02::1:ffeb:4279: ICMP6, neighbor solicitation,
who has fe80::216:3eff:feeb:4279, length 32
21:42:59.666382 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:43:00.370432 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 1 group record(s), length 28
21:43:00.372123 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:00.418355 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 1 group record(s), length 28
21:43:00.458834 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:43:00.530495 IP6 :: > ff02::1:ffcb:4689: ICMP6, neighbor solicitation,
who has fe80::216:3eff:fecb:4689, length 32
21:43:00.754372 IP6 :: > ff02::16: HBH ICMP6, multicast listener report v2,
1 group record(s), length 28
21:43:01.554375 IP6 fe80::216:3eff:fecb:4689 > ff02::16: HBH ICMP6,
multicast listener report v2, 1 group record(s), length 28
21:43:01.555676 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:02.226344 IP6 fe80::216:3eff:fecb:4689 > ff02::16: HBH ICMP6,
multicast listener report v2, 1 group record(s), length 28
21:43:02.394350 IP6 fe80::216:3eff:fecb:4689 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:02.498354 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:02.674489 IP6 :: > ff02::1:ff00:3: ICMP6, neighbor solicitation, who
has 2001:db8:200:e2a::3, length 32
21:43:02.866471 IP6 :: > ff02::1:ff00:2: ICMP6, neighbor solicitation, who
has 2001:db8:200:e2a::2, length 32
21:43:03.058343 IP6 fe80::216:3eff:fecb:4689 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:03.282386 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 2 group record(s), length 48
21:43:04.274354 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:05.298357 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:12.210426 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:12.722407 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:27.826639 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:30.130675 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:43:58.802688 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
```

```
21:44:04.946766 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:44:58.194613 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:45:08.434775 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:46:54.930643 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:47:15.410656 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:51:06.834766 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:51:23.218612 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:59:18.354686 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
21:59:34.738763 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
22:15:41.394661 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
22:15:57.778652 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
22:21:40.510670 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 3 group record(s), length 68
22:21:41.202403 IP6 fe80::216:3eff:feeb:4279 > ff02::16: HBH ICMP6,
multicast listener report v2, 3 group record(s), length 68
22:48:43.858684 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
23:49:53.874767 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
00:53:14.962814 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
01:58:47.122786 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
03:04:19.282643 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
04:03:18.226642 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
05:04:28.242605 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
06:05:38.258645 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
07:11:10.418650 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
08:10:09.362657 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
09:11:19.378681 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
10:12:29.394687 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
11:18:01.554662 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
```

```
solicitation, length 16
12:14:49.426639 IP6 fe80::216:3eff:fecb:4689 > ip6-allrouters: ICMP6, router
solicitation, length 16
12:17:42.042424 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:17:42.482665 IP6 :: > ff02::1:ffcb:4689: ICMP6, neighbor solicitation,
who has 2001:db8:200:e2a:216:3eff:fecb:4689, length 32
12:17:58.060448 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:18:04.256148 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:18:20.262441 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:18:31.613696 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:18:47.630102 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:19:03.634897 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:23:47.974840 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:27:52.096863 IP6 fe80::216:3eff:feeb:4279 > ff02::1:ff00:3: ICMP6,
neighbor solicitation, who has 2001:db8:200:e2a::3, length 32
12:27:52.097083 IP6 2001:db8:200:e2a::3 > fe80::216:3eff:feeb:4279: ICMP6,
neighbor advertisement, tgt is 2001:db8:200:e2a::3, length 32
12:27:52.097277 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:27:52.097315 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:27:53.098017 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:27:53.098043 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:27:57.138437 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
12:27:57.138477 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
12:28:02.258412 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fecb:4689, length 32
12:28:02.258432 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fecb:4689, length 24
12:28:39.602964 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:32:01.601225 IP6 2001:db8:200:e2a::2 > 2001:db8:200:e2a::3: ICMP6, echo
request, seq 1, length 64
12:32:01.601493 IP6 2001:db8:200:e2a::3 > ff02::1:ff00:2: ICMP6, neighbor
solicitation, who has 2001:db8:200:e2a::2, length 32
12:32:01.601537 IP6 2001:db8:200:e2a::2 > 2001:db8:200:e2a::3: ICMP6,
neighbor advertisement, tgt is 2001:db8:200:e2a::2, length 32
12:32:01.601688 IP6 2001:db8:200:e2a::3 > 2001:db8:200:e2a::2: ICMP6, echo
reply, seq 1, length 64
```

```
12:32:02.602061 IP6 2001:db8:200:e2a::2 > 2001:db8:200:e2a::3: ICMP6, echo
request, seq 2, length 64
12:32:02.602104 IP6 2001:db8:200:e2a::3 > 2001:db8:200:e2a::2: ICMP6, echo
reply, seq 2, length 64
12:33:22.328863 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:36:31.395880 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:36:31.396153 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:36:32.396669 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:36:32.396696 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:36:33.426513 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:36:33.426540 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:36:36.562443 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
12:36:36.562478 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
12:36:41.682431 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fecb:4689, length 32
12:36:41.682467 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fecb:4689, length 24
12:36:57.380515 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:36:57.380719 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:36:58.381252 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:36:58.381277 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:36:59.410509 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:36:59.410536 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:38:47.609408 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:39:55.091072 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:39:55.091253 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:39:56.091762 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:39:56.091787 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:40:00.338491 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
12:40:00.338531 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
```

```
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
12:40:05.458455 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fecb:4689, length 32
12:40:05.458499 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fecb:4689, length 24
12:43:09.470682 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:43:09.470774 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:43:10.482506 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:43:10.482532 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:43:11.506526 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 3, length 64
12:43:11.506552 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 3, length 64
12:43:14.642498 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
12:43:14.642538 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
12:43:19.762456 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fecb:4689, length 32
12:43:19.762500 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fecb:4689, length 24
12:43:29.028063 IP6 2001:db8:200:e2a:216:3eff:fecb:4689 >
2001:db8:200:e00::2: ICMP6, echo request, seq 1, length 64
12:43:29.029144 IP6 fe80::216:3eff:feeb:4279 > ff02::1:ffcb:4689: ICMP6,
neighbor solicitation, who has 2001:db8:200:e2a:216:3eff:fecb:4689, length
32
12:43:29.029189 IP6 2001:db8:200:e2a:216:3eff:fecb:4689 >
fe80::216:3eff:feeb:4279: ICMP6, neighbor advertisement, tgt is
2001:db8:200:e2a:216:3eff:fecb:4689, length 32
12:43:29.029206 IP6 2001:db8:200:e00::2 >
2001:db8:200:e2a:216:3eff:fecb:4689: ICMP6, echo reply, seq 1, length 64
12:43:30.029472 IP6 2001:db8:200:e2a:216:3eff:fecb:4689 >
2001:db8:200:e00::2: ICMP6, echo request, seq 2, length 64
12:43:30.029547 IP6 2001:db8:200:e00::2 >
2001:db8:200:e2a:216:3eff:fecb:4689: ICMP6, echo reply, seq 2, length 64
12:43:43.099337 IP6 fe80::216:3eff:feeb:4279 > ip6-allnodes: ICMP6, router
advertisement, length 56
12:45:48.883739 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:45:48.884004 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:45:49.884465 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:45:49.884491 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
12:45:54.130442 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
```

```
12:45:54.130478 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
12:45:59.250415 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor solicitation, who has fe80::216:3eff:fecb:4689, length 32
12:45:59.250449 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:fecb:4689, length 24
```

As we see in the above output, it clearly contains packet regarding the auto configuration and router advertisement “ICMP6, neighbor solicitation” etc...

Task 17. Explain the auto-negotiation process that takes place over the A-R segment using the packet trace as supporting material. Decode and explain all the interesting packets.

Answer:

I don't know to what depth you want me to explain, but I will try to give comprehensive explanation:

Packets snippets for auto-negotiation process that takes place over the A-R:

```
12:27:52.096863 IP6 fe80::216:3eff:feeb:4279 > ff02::1:ff00:3: ICMP6,
neighbor solicitation, who has 2001:db8:200:e2a::3, length 32
12:27:52.097083 IP6 2001:db8:200:e2a::3 > fe80::216:3eff:feeb:4279: ICMP6,
neighbor advertisement, tgt is 2001:db8:200:e2a::3, length 32
12:27:52.097277 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 1, length 64
12:27:52.097315 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 1, length 64
12:27:53.098017 IP6 2001:db8:200:e00:216:3eff:fe06:ce1e >
2001:db8:200:e2a::3: ICMP6, echo request, seq 2, length 64
12:27:53.098043 IP6 2001:db8:200:e2a::3 >
2001:db8:200:e00:216:3eff:fe06:ce1e: ICMP6, echo reply, seq 2, length 64
```

First, the client sends out a neighbor solicitation (NS) packet on the multicast group that belongs to the link local address in order to check if anyone in the broadcast range uses the same link-local address.

```
21:42:59.346519 IP6 :: > ff02::1:ffeb:4279: ICMP6, neighbor solicitation,
who has fe80::216:3eff:feeb:4279, length 32
```

If the client didn't receive any reply, it takes the address and now assumes the link local address is unique in the broadcast domain. Then it will send a neighbor advertisements in order to inform the others of its address.

After that, it uses the multicast group ff02::16 using router solicitation (RS) to locate a router (notice that now, it use the link local address “fe80::216:3eff:feeb:4279”).

```
21:43:00.372123 IP6 fe80::216:3eff:feeb:4279 > ip6-allrouters: ICMP6, router
solicitation, length 16
```

The link-local address of the router interface will respond with a router advertisement (RA).

```
12:17:42.043841 IP6 fe80::216:3eff:fe16:75a5 > ip6-allnodes: ICMP6, router
advertisement, length 5
```

Now, he client will send an NS to determine whether the global unicast address is already in use.

```
12:27:57.138437 IP6 fe80::216:3eff:fecb:4689 > fe80::216:3eff:feeb:4279:
ICMP6, neighbor solicitation, who has fe80::216:3eff:feeb:4279, length 32
```

Since no response from the multicast group. The client takes it on the global unicast address. And no it will send a neighbor advertisement message to inform the others of the local unicast address

```
12:27:57.138477 IP6 fe80::216:3eff:feeb:4279 > fe80::216:3eff:fecb:4689:
ICMP6, neighbor advertisement, tgt is fe80::216:3eff:feeb:4279, length 24
```