Otto-Friedrich-University Bamberg

Professorship for Computer Science,
Communication Services, Telecommunication,
Systems and Computer Networks

Foundation of Internet Communication

Assignment-07 :IPv6

Submitted by:
Group J

Azar Ghadami
Manjunath B Marigoudar
Shivasharan Reddy
Reem Eslam Mohamed Mekky Khalil
Reema Miranda

Supervisor: Prof. Dr. Udo Krieger

Bamberg, August 2, 2020
Summer Term 2020

Contents

List of Figures

Chapter 1

IPv6 - A first Glance

1.1 Setting Kathara Configuration files

lab.conf

```
thorin[0]="A"
thorin[image]="unibaktr/alpine:ipv6"


fili[0]="A"
fili[image]="unibaktr/alpine:ipv6"

```

Figure 1.1: Configuration of labConf file

1.1.1 Startup files for Thorin and fili

```
#!/bin/sh
ip -6 addr add 2010:db8:1::beef/64 dev eth0
```

Figure 1.2: Fili startup file.

```
#!/bin/sh
ip -6 addr add 2010:db8:1::dead/64 dev eth0
```

Figure 1.3: Thorin startup file.

1.2 Congiguring Kathara settings to in enable IPv6

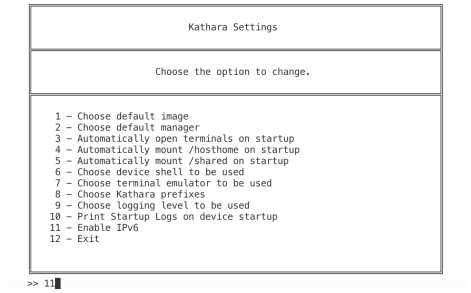


Figure 1.4: Enabling IPv6 in Kathara

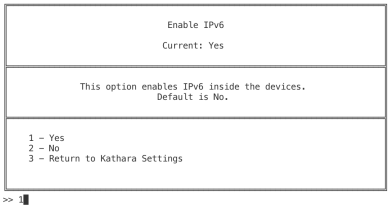


Figure 1.5: Saving settings

1.3 checking connectivity between the hosts

In this section we check the connectivity between host by using IPV6 commands.

- ping6
- traceroute6

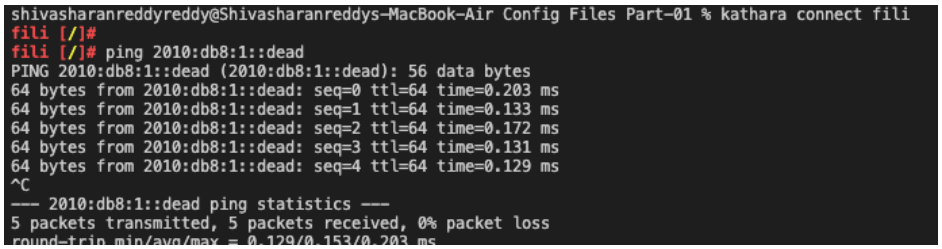


Figure 1.6: ping from thorin to fili

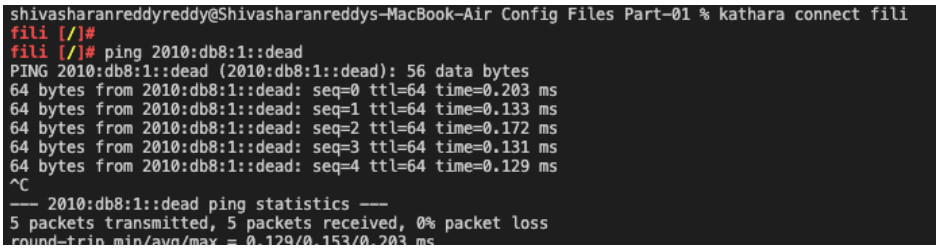


Figure 1.7: ping from thorin to fili

1.4 Describing features in IPv6 not in IPv4

IPv6 includes the following features that fix most of the limitations of IPv4

- **New Header Format.**
  The IPv6 header has a new format designed to minimize header overhead and this optimization is achieved by moving both non-essential fields and optional fields to extension headers that appear after the IPv6 header. IPv4 headers and IPv6 headers do not interoperate. IPv6 is not a superset of functionality that is backward compatible with IPv4. A host or router must use an implementation of both IPv4 and IPv6 to recognize and process both header formats. The IPv6 header is only twice as large as the IPv4 header, even though IPv6 addresses are four times as large as IPv4 addresses.

- **Larger Address Space.**
  IPv6 has 128-bit (16-byte) source and destination IP addresses. Although 128 bits can express over 3.4*1038 possible combinations, the large address space of IPv6 has been designed for multiple levels of subnetting and address allocation from the Internet backbone to the individual subnets within an organization.
  With a much larger number of available addresses, address-conservation techniques, such as the deployment of NATs, are no longer necessary.

- **Efficient and Hierarchical Addressing and Routing Infrastructure.**
  IPv6 global addresses that are used on the IPv6 portion of the Internet are designed to create an efficient, hierarchical, and summarizable routing infrastructure that is based on the common occurrence of multiple levels of Internet service providers.

- **Stateless and Stateful Address Configuration.**
  To simplify host configuration, IPv6 supports both stateful address

configuration (as in the presence of a DHCP server) and stateless address configuration (as in the absence of a DHCP server).
With stateless address configuration, hosts on a link automatically configure themselves with IPv6 addresses for the link (called link-local addresses) and with addresses that they derive from prefixes that local routers advertise. Even in the absence of a router, hosts on the same link can configure themselves with link-local addresses and communicate without manual configuration.

- **Built-in Security.**
  The IPv6 protocol suite requires support for IPSec. This requirement provides a standards-based solution for network security needs and promotes interoperability between different IPv6 implementations.

- **Better Support for QoS.**
  New fields in the IPv6 header define how traffic is handled and identified. Traffic identification (using a Flow Label field in the IPv6 header) allows routers to identify and provide special handling for packets belonging to a flow, which is a series of packets between a source and a destination. Because the IPv6 header identifies the traffic, QoS can be supported even when the packet payload is encrypted through IPSec.

- **New Protocol for Neighboring Node Interaction.**
  The Neighbor Discovery protocol for IPv6 is a series of Internet Control Message Protocol for IPv6 (ICMPv6) messages that manage the interaction of nodes on the same link (known as neighboring nodes). Neighbor Discovery replaces the broadcast-based Address Resolution Protocol (ARP), ICMPv4 Router Discovery, and ICMPv4 Redirect messages with efficient multicast and unicast Neighbor Discovery messages.

- **Extensibility.**
  IPv6 can easily be extended by adding extension headers after the IPv6 header. Unlike options in the IPv4 header, which can support only 40 bytes of options, the size of IPv6 extension headers is constrained only by the size of the IPv6 packet.

1.5 Difference between stateful and stateless auto configuration

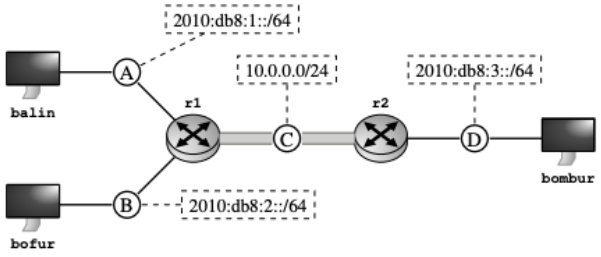
The **stateless** approach is used when user wants to configure a node without concerning a server to maintain any dynamic state for the node. In this

case, user can get prefixes and routes from router advertisements and choose its own addresses. However, the address still must be unique and properly routable. While in **stateful** DHCPv6 approach , the server handles configuration, like in IPv4 DHCP. This method is used when a site requires more precise control over address assignments.
Using **stateful** DHCP (v4 or v6) a client is more likely to keep a stable IP as long as it is on the same network, while with **stateless** method it will change frequently.

Chapter 2

IPv6 - Advanced Router Delegation

- In This Topology we use Router advertisement
- It is stateless protocol where no information is stored about the client in the routers.



Name	Docker Image	IP Address	Interface	CD
balin	unibaktr/alpine:ipv6	dhcp	eth0	A
bofur	unibaktr/alpine:ipv6	dhcp	eth0	B
bombur	unibaktr/alpine:whoami	2010:db8:3::2/64	eth0	D
r1	unibaktr/alpine:ipv6	2010:db8:1::1/64	eth0	A
		2010:db8:2::1/64	eth1	B
		10.0.0.1/24	eth2	C
r2	unibaktr/alpine:ipv6	10.0.0.2/24	eth0	C
		2010:db8:3::1/64	eth1	D

Figure 2: IPv6 - Router Advertisements

Figure 2.1: Advanced Router Delegation Topology

2.1 Configuration on R1

2.1.1 Managed router advertisement

- Managed router advertisements for CD A ranging from 2010:db8:1::100/64 to 2010:db8:1::200/64.

- on r1 create a /etc/radvd.conf file and add the managed router advertisement code

- radvd file will send route advertisements to all the nodes on the network.

- for interface 0 router 1 advertises a range of 2010:db8:1::100/64 to 2010:db8:1::200/64 address to balin ,so that it accepts the traffic entering this range of address from balin

r1 > etc > radvd.conf

```
interface eth0
{
    AdvSendAdvert on;
    AdvManagedFlag on;

};

interface eth1
{
    AdvSendAdvert on;
    prefix 2010:db8:2::/64
     {};
};
interface tun6to4
{
prefix 0:0:0:5678::/64
{
    AdvOnLink on;
    AdvAutonomous on;
    Base6to4Interface eth2;

};

};
```

Figure 2.2: Managed Advertisement

- we have added the managed advertisement for interface 0 of router 1 as highlighted in the above picture

- AdvManagedFlag : when it is set to 'on' it uses the administered stateful protocol for address auto configuration.By default this flag will be off.

2.1.2 Unmanaged router advertisement

r1 > etc > radvd.conf

```
interface eth0
{
    AdvSendAdvert on;
    AdvManagedFlag on;

};

interface eth1
{
    AdvSendAdvert on;
    prefix 2010:db8:2::/64
    {};
};
interface tun6to4
{
prefix 0:0:0:5678::/64
{
    AdvOnLink on;
    AdvAutonomous on;
    Base6to4Interface eth2;

};

};
```

Figure 2.3: Unmanaged Advertisement

- Unmanaged router advertisements for CD B 2010:db8:2::/64
- As highlighted in the above picture interface 1 of router 1 advertises 2010:db8:2::/64 address to bofur .
- In this section we have used prefix:2010:db8:2::/64 which means we only advertise this address to bofur and accept the traffic flowing through this address from bofur.

2.2 6-to-4 tunnel establishment

- To establish connection between r1 and r2 we need to enable IPv6 traffic. As r1 and r2 are assigned with IPv6 addresses and CD is assigned with IPv4, we can not establish a connection between them.
- Below screenshot shows the config/code part that helps to establish the 6-to-4 tunnelling.

r1.startup

```
#!/bin/sh
ip -6 addr add 2010:db8:1::1/64 dev eth0
ip -6 addr add 2010:db8:2::1/64 dev eth1
ip addr add 10.0.0.1/24 dev eth2

ip tunnel add tun6to4 mode sit ttl 10 remote 10.0.0.2 local 10.0.0.1
ip link set dev tun6to4 up
ip -6 addr add 2010:db8:4::1/64 dev tun6to4
ip -6 route add 2010:db8:3::/64 via 2010:db8:4::2 dev tun6to4 metric 1
radvd
dhcpd -6
```

Figure 2.4: 6-to-4 tunnelling in r1

r2.startup

```
#!/bin/sh
ip addr add 10.0.0.2/24 dev eth0
ip -6 addr add 2010:db8:3::1/64 dev eth1

ip tunnel add tun6to4 mode sit ttl 10 remote 10.0.0.1 local 10.0.0.2
ip link set dev tun6to4 up
ip -6 addr add 2010:db8:4::2/64 dev tun6to4
ip -6 route add  default via 2010:db8:4::1 dev tun6to4 metric 1
radvdexi
```

Figure 2.5: 6-to-4 tunnelling in r2

- A 6to4 tunnel interface automatically converts the 32 bits in its IPv6 address following this prefix to a global unicast IPv4 address for transport across an IPv4 network such as the public Internet.

2.3 Checking Global connectivity

- To establish global connectivity we have added assigned addresses to all the machines and provided default routers wherever required as per the given data.

- Below screenshots shows the connectivity between the machines

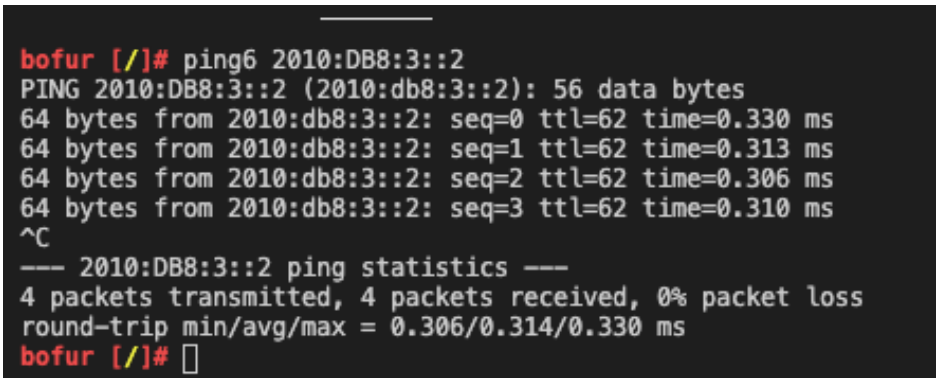


Figure 2.6: Connectivity between bofur to bombur

- To connect to balin we need get the IP assigned by dhcp. Using Ifconfig we can get the address assigned to balin as below,

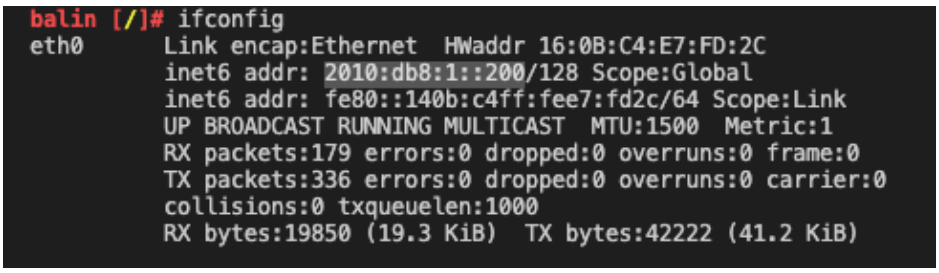


Figure 2.7: IP assigned to balin by dhcp

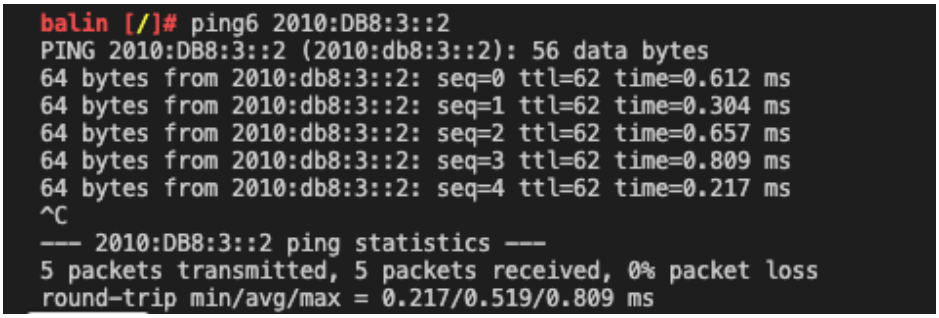


Figure 2.8: Connectivity between balin to bombur

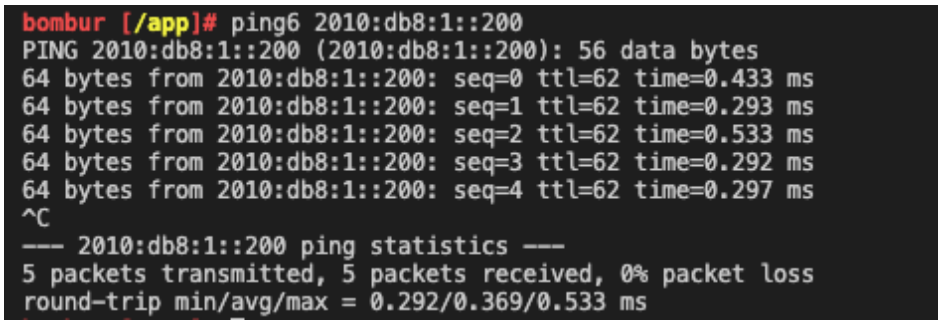


Figure 2.9: Connectivity between bombur to balin

2.4 Wireshark and 6-to-4 tunnel functionalities

- We can not capture the curl with wireshark in windows and Mac systems.

2.4.1 6-to-4 functionalities

- It enables encapsulation of IPv6 packets into IPv4 for transport across an IPv4 network.
- It allows for automatic IPv6-to-IPv4 address translation, and treats the underlying IPv4 network as one big non-broadcast multiaccess (NBMA) network, rather than a collection of independent point-to-point links.