H2020 i**P** **O**ver **I**c**N**- the bet**T**er IP (POINT)

Examples

Deploy an IP/HTTP-over-ICN POINT Network

List of Authors: Sebastian Robitzsch

1. Overview

The main objective of this document is to guide the reader through required steps to set up an IP-to-IP communication an ICN network made up of two nodes. It is considered that the reader has access to POINT's D3.1 deliverable which describes the NAP in further detail.

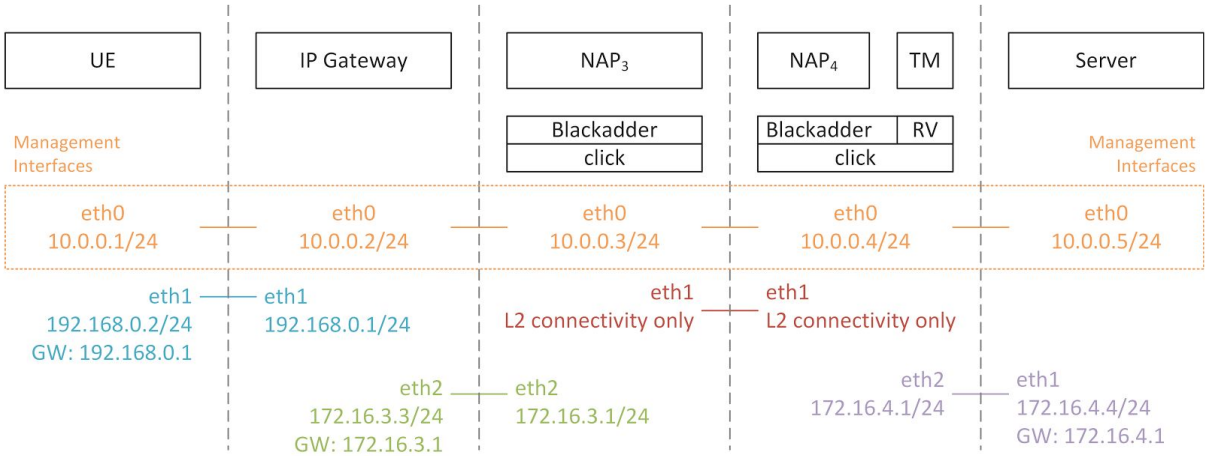

Figure 1: IP-over-ICN set-up described in this section

If not mentioned otherwise, it is expected that the user configures the interfaces in all nodes (UE, IP gateway, NAPs and Server) according to Figure 1. All eth0 interfaces coloured in orange are managed interfaces which allows to communicate among all five nodes using standard IP tools, e.g., ssh, scp, telnet or ping.

2. Configurations

2.1 IP Endpoint Server

The server can be configured either way (static/automatic address assignment). Just ensure that the default GW is 172.16.4.1.

To make this setting permanent add the following line to you interface configuration file on the server machine (either /etc/network/interfaces or /etc/network/interfaces.d/eth1.conf):

```
iface eth1 inet static
      address 172.16.4.4/24
```

```
    netmask 255.255.255.0
    gateway 172.16.4.1
```

2.2 IP Gateway

The IP gateway must provide NAT with eth2 as the outgoing (public) interface and eth1 as the internal one facing the local IP endpoints.

2.2.1 NAT

To set up the NAT between eth1 and eth2 invoke the following commands:

```
$ echo 1 > /proc/sys/net/ipv4/ip_forward
$ iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE
$ iptables -A FORWARD -i eth2 -o eth1 -m state --state
RELATED,ESTABLISHED -j ACCEPT
$ iptables -A FORWARD -i eth1 -o eth2 -j ACCEPT
```

To make the NATting a permanent setting which applies after reboot simply add the lines above to /etc/rc.local before the exit 0 statement.

2.2.2 DHCP Server

It is recommended that the IP assignment of IP endpoints connected to the IP gateway should be conducted automatically via DHCP. The default DHCP server in Debian-based repositories can be installed via:

```
~$ apt install isc-dhcp-server
```

Now tell the software to bind on eth1 for DHCP requests by changing the value of INTERFACES in /etc/default/isc-dhcp-server to:

```
INTERFACES=”eth1”
```

Now the subnet range has to be configured used by the DHCP server to respond to DHCP request from IP endpoints seeking for an IP. To do so add the following lines to /etc/dhcp/dhcpd.conf:

```
subnet 192.168.0.0 netmask 255.255.255.0 {
   range 192.168.0.2 192.168.0.254;
   option routers 192.168.0.1;
}
```

Before restarting the DHCP server to apply the changes above, ensure that eth1 has been set up according to Figure 1 (192.168.0.1/24). Now restart the DHCP server:

```
~$ service isc-dhcp-server restart
```

Sometimes the DHCP server cannot be started automatically when rebooting the system due to some delay in configuring the network interfaces. To overcome this issue add the line above to /etc/rc.local.

2.3 IP Endpoint UE

This IP endpoint has a single interface which must be physically connected to the gateway's eth1 interface. As this how-to guides the user through the set-up of a DHCP server on the IP gateway, the UE's interface should be configured to automatically obtain its IP configuration. Simply verify that /etc/network/interfaces (or in case a dedicated configuration file per NIC is created in /etc/network/interfaces.d/*) has the following entry:

```
iface eth1 inet dhcp
```

and for eth0:

```
iface eth0 inet static
   address 10.0.0.1
   netmask 255.255.255.0
```

Furthermore, ensure that UE's kernel has the appropriate routing table entry to reach the server on 172.16.4.4 via the IP gateway instead of the management interface eth0. To do so, run the route command and ensure your routing table is identical to the one below.

```
~$ route -n
~$ Kernel IP routing table
Destination Gateway     Genmask  Flags Metric Ref      Use Iface
0.0.0.0     192.168.0.1 0.0.0.0  UG    1024   0        0 eth1
10.0.0.0    0.0.0.0   255.255.255.0   U     1000   0   0 eth0
192.168.0.0 0.0.0.0   255.255.255.0   U     0     0     0 eth1
```

2.4 Network Attachment Point

As the Kernel is not aware of the NAP, ICMP destination unreachable responses can be turned off:

```
~$ sudo iptables -I OUTPUT -p icmp --icmp-type
destination-unreachable -j DROP
```

In order to make this a permanent setting simply add the line to /etc/rc.local, before the exit 0 command and make /etc/rc.local executable:

```
~$ sudo chmod +x /etc/rc.local
```

3. Deploy the Network

3.1 The ICN Core

The doc directory of the nap folder comes with an example Blackadder configuration file which reflects the scenario illustrated in Figure 3.1. Only change this file if interface names or IP addresses have been changed.

First, deploy the network of two ICN nodes where the TM and the RVZ are running on the same machine as the NAP. To do so Blackadder comes with a deployment tool which requires sudo privileges to run the underlying click platform. Please ensure that the sudo command does not require a password input on the machines where the NAPs are running (see Section 2.1.1 in the HowTo document) and that SSH keys were distributed:

```
NAP3~$ ssh-copy-id –i .ssh/id_rsa.pub point@10.0.0.3
NAP3~$ ssh-copy-id –i .ssh/id_rsa.pub point@10.0.0.4

NAP4~$ ssh-copy-id –i .ssh/id_rsa.pub point@10.0.0.4
NAP4~$ ssh-copy-id –i .ssh/id_rsa.pub point@10.0.0.3
```

Now the topology can be deployed by running the deployment tool:

```
$ cd blackadder/deployment
$ ./deploy –c examples/ip-over-icn.cfg
```

Note, this starts the RV and TM.

3.2 The Network Attachment Points (NAPs)

As both ICN nodes have Blackadder running and RV and TM are up, the NAPs can be started to serve their IP endpoints. Note, it does not matter whether the NAP facing the UE or the one facing the server is started first.

On NAP_3 invoke:

```
~$ sudo nap –c ~/blackadder/apps/nap/doc/examples/nap3.cfg
```

On NAP_4 invoke:

```
~$ sudo nap –c ~/blackadder/apps/nap/doc/examples/nap4.cfg
```

3.3 Verify the Network

This completes the setup and IP traffic can be sent from the UE to the server and vice versa. Test this by issuing a ping from the UE towards the server over the deployed ICN network:

```
~$ ping 172.16.4.4 –c 3
```

Or by SSHing into the server from the UE:

```
~$ ssh 172.16.4.4
```

4. HTTP-over-ICN

This example provides the required steps to set up an HTTP-over-ICN example. In order to make this work it is assumed that the previous IP-over-ICN scenario has been successfully completed and all nodes are still operational. This example will enable the client to issue an HTTP request to web.point which is delivered via HTTP-over-ICN to the web server behind the server NAP (sNAP).

4.1 Prerequisites

4.1.1 Server

On the server, simply install a webserver, e.g., Apache:

```
~$ sudo apt install apache2
```

And test that it runs correctly by opening a browser on the server machine and go to http://localhost. If there has been no GUI installed, use the command line browser w3m:

```
~$ w3m localhost
```

4.1.2 Client

As there is no DNS running in the network, the client requires a local DNS entry for the FQDN web.point. To do so, open /etc/hosts as root and add

```
172.16.4.4 web.point
```

Before the IPv6 related entries.

4.2 Running the Software

As for the IP-over-ICN example the network must be deployed and the NAPs started manually. Please follow the steps in Section 3.2.

Eventually, open a web browser or wget on the client and go to web.point. This gives you back the default Apache index.html page saying that the web server works.