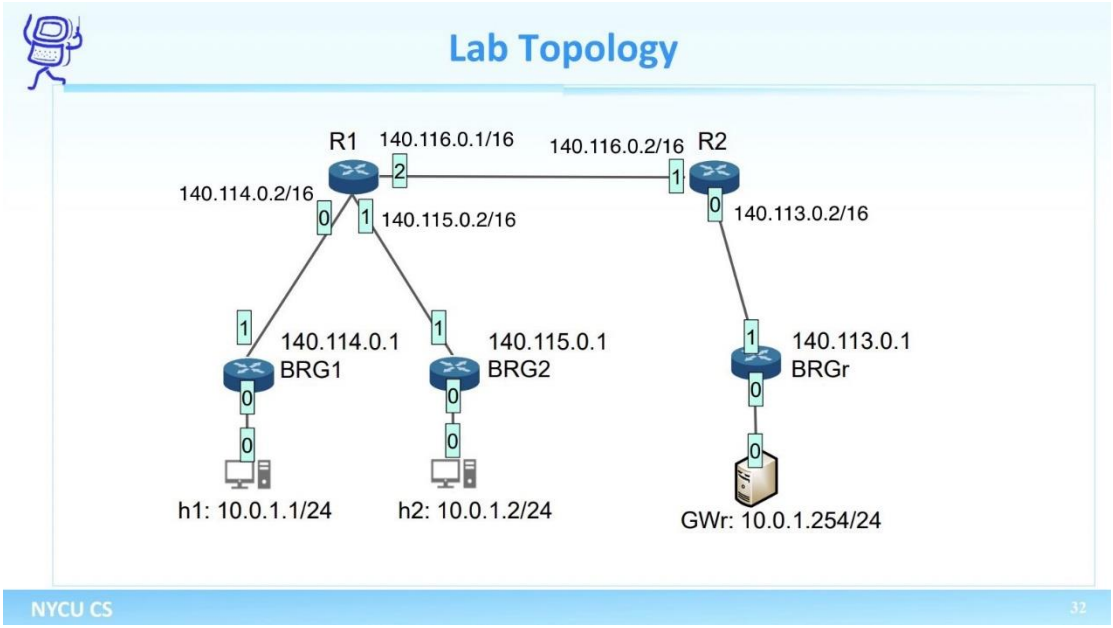
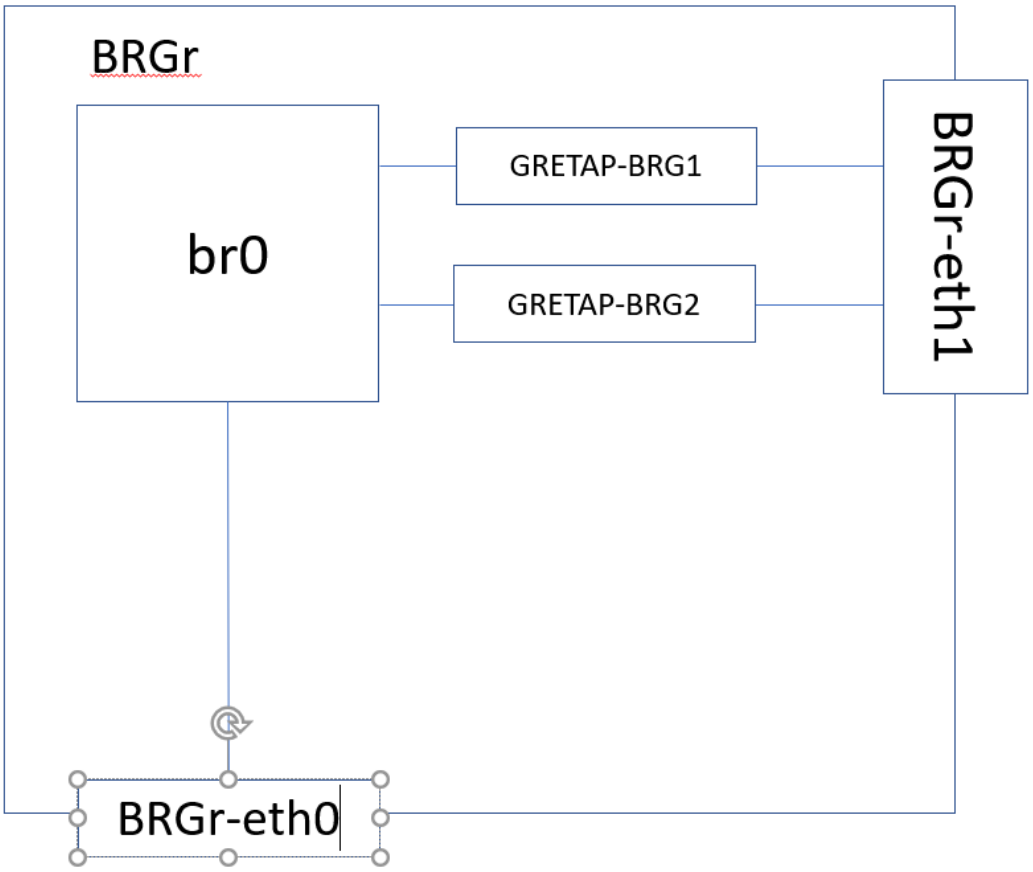


Above is my configuration of this lab.

Part 1

1. Show all interfaces of node BRGr and draw the interconnection diagram of interfaces and Linux bridge on BRGr. Explain your diagram with the interface list of BRGr.

```
eric070021@ubuntu:~$ docker exec BRGr ifconfig
BRGr-eth0 Link encap:Ethernet  HWaddr 7e:33:8d:d6:14:7c
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1161 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1163 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:109130 (109.1 KB)  TX bytes:109238 (109.2 KB)

BRGr-eth1 Link encap:Ethernet  HWaddr 6a:70:93:ff:83:03
          inet addr:140.113.0.1  Bcast:0.0.0.0  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1232 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1248 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:156598 (156.5 KB)  TX bytes:158588 (158.5 KB)

GRETAP-BRG1 Link encap:Ethernet  HWaddr ba:02:ad:c2:f6:25
          UP BROADCAST RUNNING MULTICAST  MTU:1462  Metric:1
          RX packets:359 errors:0 dropped:0 overruns:0 frame:0
          TX packets:368 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:32830 (32.8 KB)  TX bytes:28416 (28.4 KB)

GRETAP-BRG2 Link encap:Ethernet  HWaddr 32:b5:8c:58:e3:77
          UP BROADCAST RUNNING MULTICAST  MTU:1462  Metric:1
          RX packets:802 errors:0 dropped:0 overruns:0 frame:0
          TX packets:814 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:76300 (76.3 KB)  TX bytes:65936 (65.9 KB)

br0       Link encap:Ethernet  HWaddr 32:b5:8c:58:e3:77
          UP BROADCAST RUNNING MULTICAST  MTU:1462  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:56 (56.0 B)  TX bytes:108 (108.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```



BRGr-eth1 connects to R2. BRGr-eth0 connects to GWr. GRETAP-BRG1 connect to BRGr-eth1 so that it can decapsulate the gre packet from BRG1. GRETAP-BRG2 connect to BRGr-eth1 so that it can decapsulate the gre packet from BRG2. Br0 connects to both GRETAP-BRG1 and GRETAP-BRG2.

2. Let h1 and h2 ping GWr and take screenshot of ping results.

```
eric070021@ubuntu:~$ docker exec h1 ping 10.0.1.254 -c 3
PING 10.0.1.254 (10.0.1.254) 56(84) bytes of data.
64 bytes from 10.0.1.254: icmp_seq=1 ttl=64 time=1.99 ms
64 bytes from 10.0.1.254: icmp_seq=2 ttl=64 time=0.189 ms
64 bytes from 10.0.1.254: icmp_seq=3 ttl=64 time=0.119 ms

--- 10.0.1.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2010ms
rtt min/avg/max/mdev = 0.119/0.767/1.993/0.867 ms
```

```
eric070021@ubuntu:~$ docker exec h2 ping 10.0.1.254 -c 3
PING 10.0.1.254 (10.0.1.254) 56(84) bytes of data.
64 bytes from 10.0.1.254: icmp_seq=1 ttl=64 time=0.092 ms
64 bytes from 10.0.1.254: icmp_seq=2 ttl=64 time=0.177 ms
64 bytes from 10.0.1.254: icmp_seq=3 ttl=64 time=0.138 ms

--- 10.0.1.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2032ms
rtt min/avg/max/mdev = 0.092/0.135/0.177/0.037 ms
```

3. Can h1 ping h2? Take screenshots to explain why or why not

Yes, h1 will forward packet to GWr, and GWr will forward packet to h2.

```
eric070021@ubuntu:~$ docker exec h1 ping 10.0.1.2 -c 3
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data.
64 bytes from 10.0.1.2: icmp_seq=1 ttl=64 time=1.73 ms
64 bytes from 10.0.1.2: icmp_seq=2 ttl=64 time=0.115 ms
64 bytes from 10.0.1.2: icmp_seq=3 ttl=64 time=0.144 ms

--- 10.0.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2011ms
rtt min/avg/max/mdev = 0.115/0.663/1.730/0.754 ms
```

```
eric070021@ubuntu:~$ docker exec -it h1 bash
root@f85be9122bee:/# tcpdump -i h1-eth0 icmp -XX -n -c 2
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h1-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
07:17:01.605964 IP 10.0.1.1 > 10.0.1.2: ICMP echo request, id 272, seq 1, length 64
        0x0000:  fa63 2408 a228 1e26 86b6 014b 0800 4500  .c$..(.&...K..E.
        0x0010:  0054 c0cc 4000 4001 63da 0a00 0101 0a00  .T..@.@.c.......
        0x0020:  0102 0800 06a5 0110 0001 edd5 5362 0000  ............Sb..
        0x0030:  0000 e73e 0900 0000 0000 1011 1213 1415  ...>............
        0x0040:  1617 1819 1a1b 1c1d 1e1f 2021 2223 2425  ..........!"#$%
        0x0050:  2627 2829 2a2b 2c2d 2e2f 3031 3233 3435  &'()*+,-./012345
        0x0060:  3637                                     67
07:17:01.607236 IP 10.0.1.2 > 10.0.1.1: ICMP echo reply, id 272, seq 1, length 64
        0x0000:  1e26 86b6 014b fa63 2408 a228 0800 4500  .&...K.c$..(..E.
        0x0010:  0054 19d4 0000 4001 4ad3 0a00 0102 0a00  .T....@.J.......
        0x0020:  0101 0000 0ea5 0110 0001 edd5 5362 0000  ............Sb..
        0x0030:  0000 e73e 0900 0000 0000 1011 1213 1415  ...>............
        0x0040:  1617 1819 1a1b 1c1d 1e1f 2021 2223 2425  ..........!"#$%
        0x0050:  2627 2829 2a2b 2c2d 2e2f 3031 3233 3435  &'()*+,-./012345
        0x0060:  3637                                     67
2 packets captured
2 packets received by filter
0 packets dropped by kernel
```

4. Explain how Linux kernel of BRGr determines which gretap interface to forward packets from GWr to hosts (h1 or h2)? Describe your answer with appropriate screenshots.

There is a MAC learning table on the bridge. When it receives a frame that the destination MAC address is not in its MAC table, it floods arp frames to every LAN port of the same VLAN except for the port that the frame received. When the destination station responds, the bridge adds the MAC address and port ID to the MAC table.

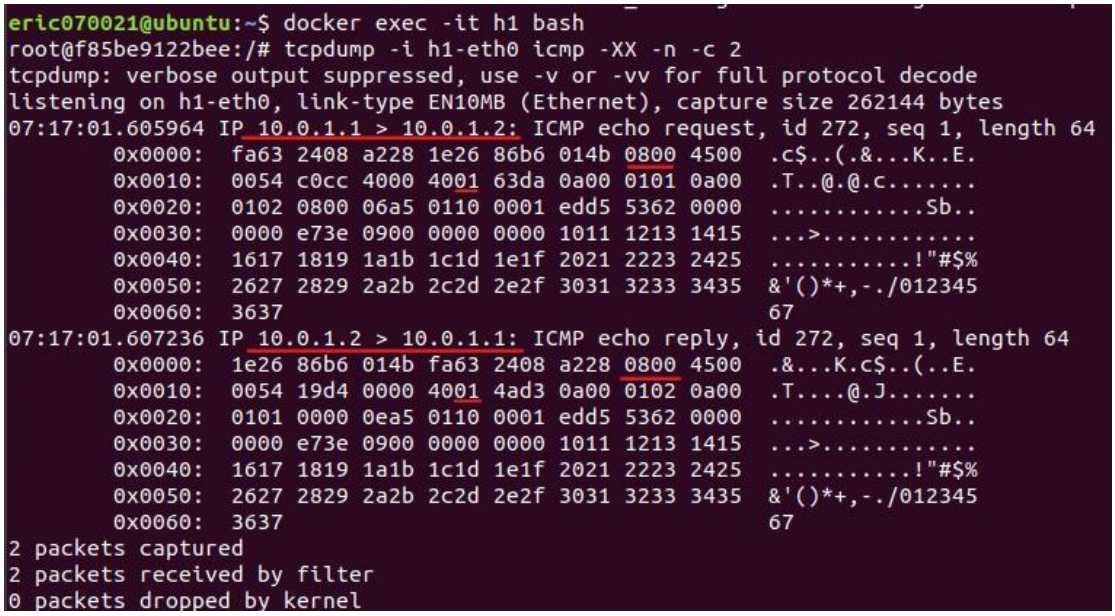
```
eric070021@ubuntu:~$ docker exec -it h1 ifconfig
h1-eth0   Link encap:Ethernet  HWaddr 1e:26:86:b6:01:4b
          inet addr:10.0.1.1  Bcast:0.0.0.0  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:381 errors:0 dropped:0 overruns:0 frame:0
          TX packets:372 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:34618 (34.6 KB)  TX bytes:33880 (33.8 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

```
eric070021@ubuntu:~$ docker exec -it h2 ifconfig
h2-eth0   Link encap:Ethernet  HWaddr fa:63:24:08:a2:28
          inet addr:10.0.1.2  Bcast:0.0.0.0  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:828 errors:0 dropped:0 overruns:0 frame:0
          TX packets:815 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:78480 (78.4 KB)  TX bytes:77350 (77.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:504 (504.0 B)  TX bytes:504 (504.0 B)
```

```
eric070021@ubuntu:~$ docker exec -it BRGr bash
root@7669b8cdaf82:/# bridge fdb show br0
33:33:00:00:00:01 dev gretap0 self permanent
33:33:00:00:00:01 dev erspan0 self permanent
1e:26:86:b6:01:4b dev GRETAP-BRG1 master br0
ba:02:ad:c2:f6:25 dev GRETAP-BRG1 vlan 1 master br0 permanent
ba:02:ad:c2:f6:25 dev GRETAP-BRG1 master br0 permanent
33:33:00:00:00:01 dev GRETAP-BRG1 self permanent
01:00:5e:00:00:01 dev GRETAP-BRG1 self permanent
fa:63:24:08:a2:28 dev GRETAP-BRG2 master br0
32:b5:8c:58:e3:77 dev GRETAP-BRG2 vlan 1 master br0 permanent
32:b5:8c:58:e3:77 dev GRETAP-BRG2 master br0 permanent
33:33:00:00:00:01 dev GRETAP-BRG2 self permanent
01:00:5e:00:00:01 dev GRETAP-BRG2 self permanent
33:33:00:00:00:01 dev br0 self permanent
01:00:5e:00:00:6a dev br0 self permanent
33:33:00:00:00:6a dev br0 self permanent
01:00:5e:00:00:01 dev br0 self permanent
4e:75:d0:0b:ff:b8 dev BRGr-eth0 master br0
7e:33:8d:d6:14:7c dev BRGr-eth0 vlan 1 master br0 permanent
7e:33:8d:d6:14:7c dev BRGr-eth0 master br0 permanent
33:33:00:00:00:01 dev BRGr-eth0 self permanent
01:00:5e:00:00:01 dev BRGr-eth0 self permanent
33:33:00:00:00:01 dev BRGr-eth1 self permanent
01:00:5e:00:00:01 dev BRGr-eth1 self permanent
```

5. Is h1 aware of GRE tunneling? Take screenshot to explain why or why not

No. GRE header is encapsulated and decapsulated at BRGr, which means h1 will only get the packet content encapsulated inside GRE header. The tcpdump result can show it. The packets captured at h1 are ETHERType IPv4(0x0800) with Protocol ICMP(0x01), not GRE(0x2F).



Part 2

Run Tunnel Auto Creation Program on BRGr and show the parsed result of one captured packet.

```
0 Name: br0
1 Name: BRGr-eth0
2 Name: GRETAP-BRG1
3 Name: BRGr-eth1
4 Name: GRETAP-BRG2
5 Name: any
6 Name: lo
7 Name: nflog
8 Name: nfqueue
9 Name: usbmon1
10 Name: usbmon2
Insert a number to select interface:
3
Start listening at $BRGr-eth1
Insert BPF filter expression:
ip proto gre
filter: ip proto gre and dst host 140.113.0.1
Packet Num [1]
Outer Source MAC: 36:a9:a0:88:18:35
Outer Destination MAC: 6a:70:93:ff:83:03
Outer Ethernet type: 0800
Outer Source IP: 140.114.0.1
Outer Destination IP: 140.113.0.1
Next Layer Protocol: GRE
Protocol: 6558
Inner Source MAC: 1e:26:86:b6:01:4b
Inner Destination MAC: 4e:75:d0:0b:ff:b8
Inner Ethernet type: 0800
6a 70 93 ff 83 03 36 a9 a0 88 18 35 08 00 45 00 00 7a a1 50 40 00 3e 2f
82 1f 8c 72 00 01 8c 71 00 01 00 00 65 58 4e 75 d0 0b ff b8 1e 26 86 b6
01 4b 08 00 45 00 00 54 a2 f2 40 00 40 01 80 b8 0a 00 01 01 0a 00 01 fe
08 00 e5 99 00 d2 00 01 1a c9 53 62 00 00 00 00 e0 94 04 00 00 00 00 00
10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27
28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37
```