Static routing – ruting statyczny

- W ramach zajęć należy stworzyć prostą sieć składającą się z czterech maszyn wirtualnych – dwóch komputerów (PC1 I PC2) oraz dwóch ruterów (R1 i R2).
  - Adresy IP poszczególnych interfejsów urządzeń zostały już skonfigurowane, należy sprawdzić poprawność tej konfiguracji
  - Należy zbadać łączność pomiędzy urządzeniami końcowymi z użyciem polecenia **`ping`** i sprawdzić zawartość tablic rutingu z użyciem polecenia **`route`**
  - Należy uzupełnić konfigurację urządzeń w taki sposób, aby ruting statyczny działał poprawnie
  - Należy użyć programu **`tcpdump`** w celach diagnostycznych
- Działania te powinny umożliwić komunikację komputerów za pośrednictwem sieci

Static routing – ruting statyczny

- **Głównym celem zajęć jest**
  - skonfigurowanie rutingu statycznego
  - analiza działania rutingu statycznego
  - lepsze zapoznanie się ze środowiskiem **`kathara`**

Università degli Studi Roma Tre
Dipartimento di Ingegneria
Computer Networks Research Group

kathara lab

static-routing

Version	1.1
Author(s)	L. Ariemma, G. Di Battista, M. Patrignani, M. Pizzonia, F. Ricci, M. Rimondini
E-mail	contact@kathara.org
Web	https://www.kathara.org/
Description	an example of configuration of static routes – kathara version of netkit lab static-routing vers. 2.2

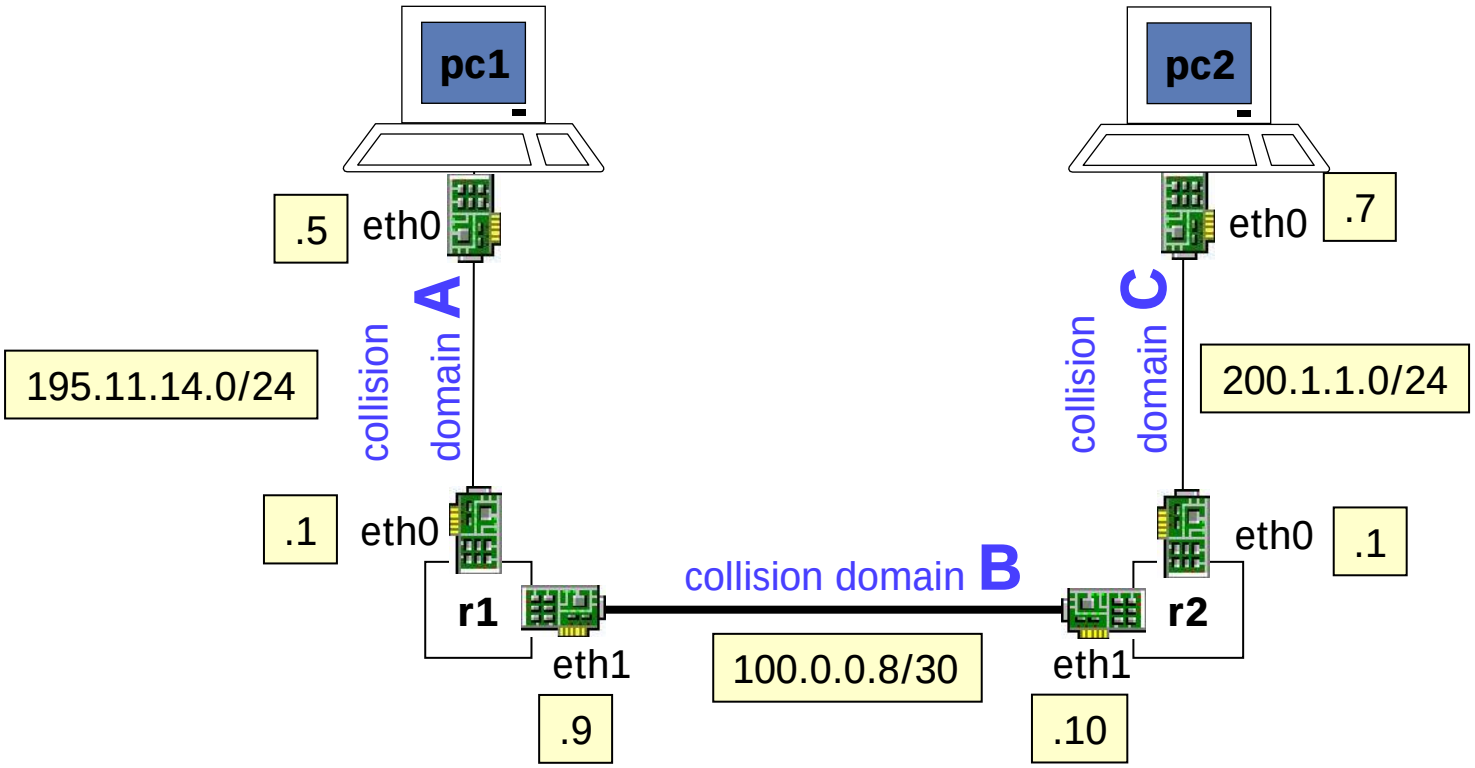
Copyright notice

- All the pages/slides in this presentation, including but not limited to, images, photos, animations, videos, sounds, music, and text (hereby referred to as “material”) are protected by copyright.
- This material, with the exception of some multimedia elements licensed by other organizations, is property of the authors and/or organizations appearing in the first slide.
- This material, or its parts, can be reproduced and used for didactical purposes within universities and schools, provided that this happens for non-profit purposes.
- Information contained in this material cannot be used within network design projects or other products of any kind.
- Any other use is prohibited, unless explicitly authorized by the authors on the basis of an explicit agreement.
- The authors assume no responsibility about this material and provide this material “as is”, with no implicit or explicit warranty about the correctness and completeness of its contents, which may be subject to changes.
- This copyright notice must always be redistributed together with the material, or its portions.

Lab Scenario Personalization

- Please modify the default scenario in the following way
  - In the next slide change the name of `r1` to `r<LAB-ID>` for each configuration file, where LAB-ID is your personal ID assigned by the lab instructor
- **Note well**: from now-on
  - Command-line commands should reflect the change in naming, therefore there can be differences in the outputs shown in the manual

Step 1 – Network topology



Step 2 – The lab

- lab directory hierarchy
  - lab.conf
  - pc1.startup
  - pc2.startup
  - r1.startup
  - r2.startup

Step 2 – The lab

lab.conf

```
r1[0]=A
r1[1]=B

r2[0]=C
r2[1]=B

pc1[0]=A
pc2[0]=C
```

pc1.startup

```
ifconfig eth0 195.11.14.5/24 up
```

pc2.startup

```
ifconfig eth0 200.1.1.7/24 up
```

r1.startup

```
ifconfig eth0 195.11.14.1/24 up
ifconfig eth1 100.0.0.9/30 up
```

r2.startup

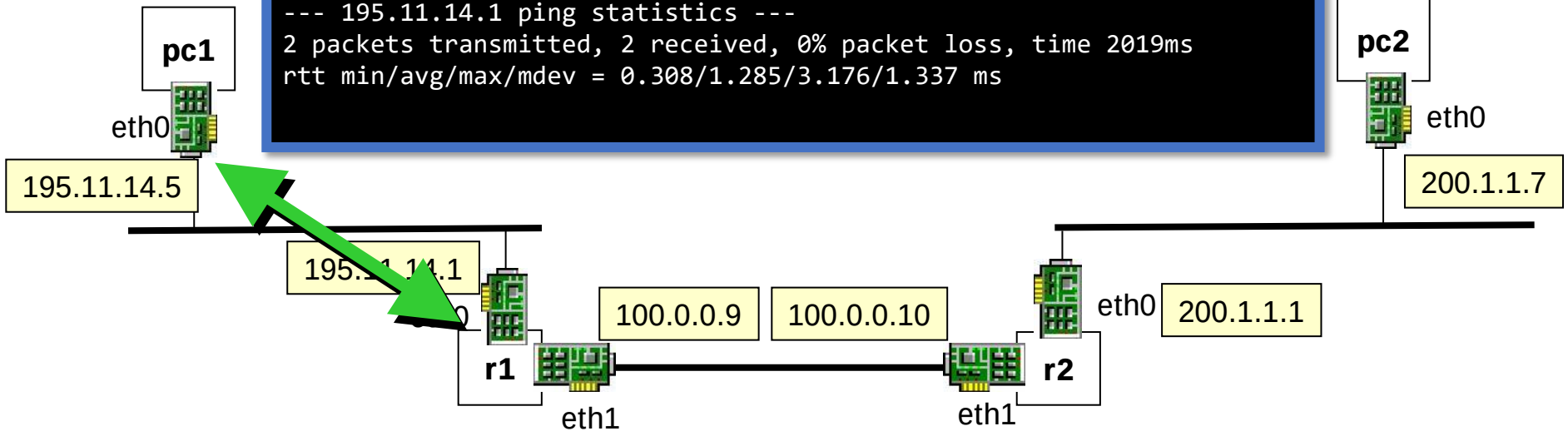
```
ifconfig eth0 200.1.1.1/24 up
ifconfig eth1 100.0.0.10/30 up
```

Step 3 – Testing connectivity

interfaces on the same domain can reach each other

```
root@pc1:~$ ping 195.11.14.1
PING 195.11.14.1 (195.11.14.1) 56(84) bytes of data.
64 bytes from 195.11.14.1: icmp_seq=1 ttl=64 time=3.17 ms
64 bytes from 195.11.14.1: icmp_seq=2 ttl=64 time=0.371 ms

--- 195.11.14.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2019ms
rtt min/avg/max/mdev = 0.308/1.285/3.176/1.337 ms
```

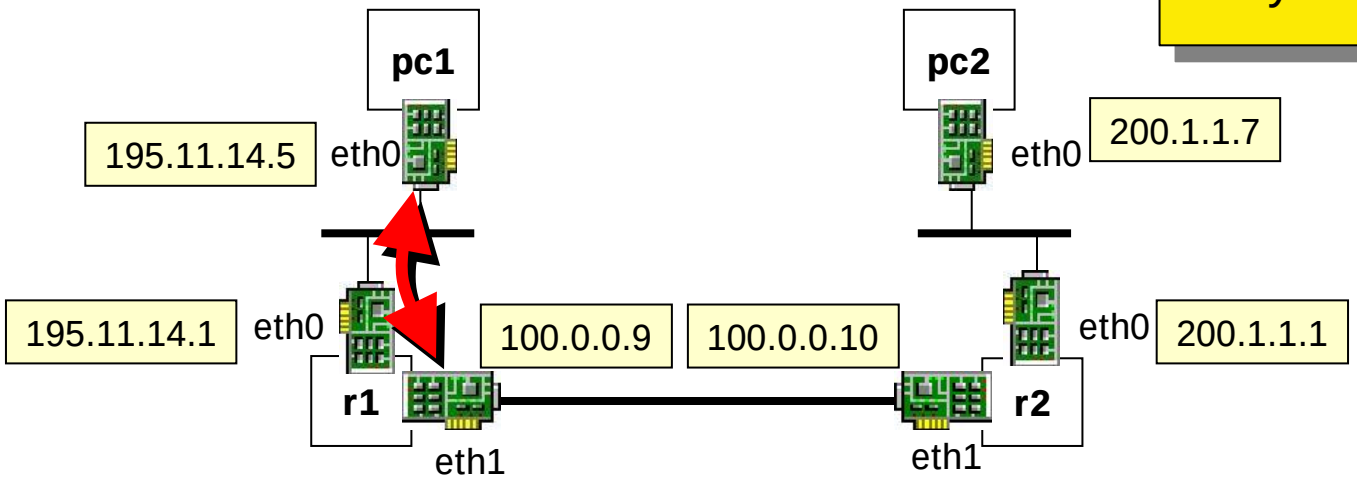


Step 3 – Testing connectivity

```
root@pc1:~$ ping 100.0.0.9
connect: Network is unreachable
```

interfaces on different domains cannot be reached

can you tell why?



Step 3 – Inspecting routing tables

- Both routers and PCs don't know how to reach networks that are not directly connected to them

```
root@r1:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
100.0.0.8       *               255.255.255.252 U     0      0        0 eth1
195.11.14.0     *               255.255.255.0   U     0      0        0 eth0
```

- Directly connected networks are automatically inserted into the routing table when the corresponding interface is brought up
- This is a common behavior of all IP devices (even real-world routers!)

Step 4 – Default routes on PCs

- To fix the problem we could specify the default route on the pcs: “through this gateway (IP number) you can reach all the other networks”

```
root@pc1:~$ route add default gw 195.11.14.1
root@pc1:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
195.11.14.0     *               255.255.255.0   U     0      0        0 eth0
default         195.11.14.1     0.0.0.0         UG    0      0        0 eth0
```

```
root@pc2:~$ route add default gw 200.1.1.1
root@pc2:~$ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
200.1.1.0       *               255.255.255.0   U     0      0        0 eth0
default         200.1.1.1       0.0.0.0         UG    0      0        0 eth0
```

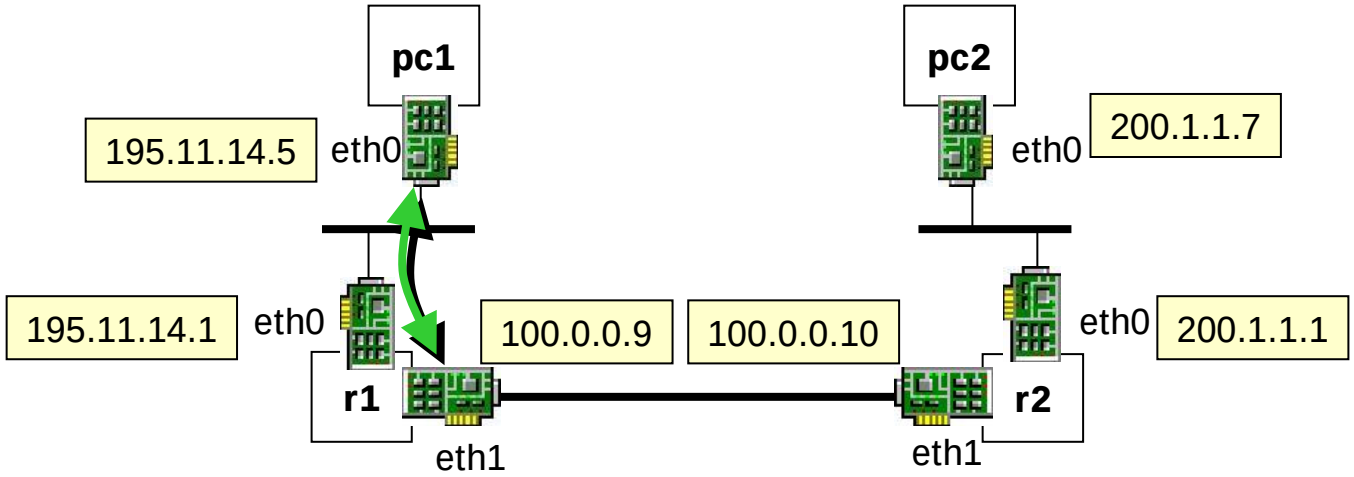
Step 4 – Default routes on PCs: test

pc1

```
pc1:~# ping 100.0.0.9
PING 100.0.0.9 (100.0.0.9) 56(84) bytes of data.
64 bytes from 100.0.0.9: icmp_seq=1 ttl=64 time=0.451 ms
64 bytes from 100.0.0.9: icmp_seq=2 ttl=64 time=0.299 ms
64 bytes from 100.0.0.9: icmp_seq=3 ttl=64 time=0.320 ms

--- 100.0.0.9 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.299/0.356/0.451/0.070 ms
pc1:~#
```

the "backbone interface" of **r1** is reachable



Step 4 – Default routes on PCs: test

pc1

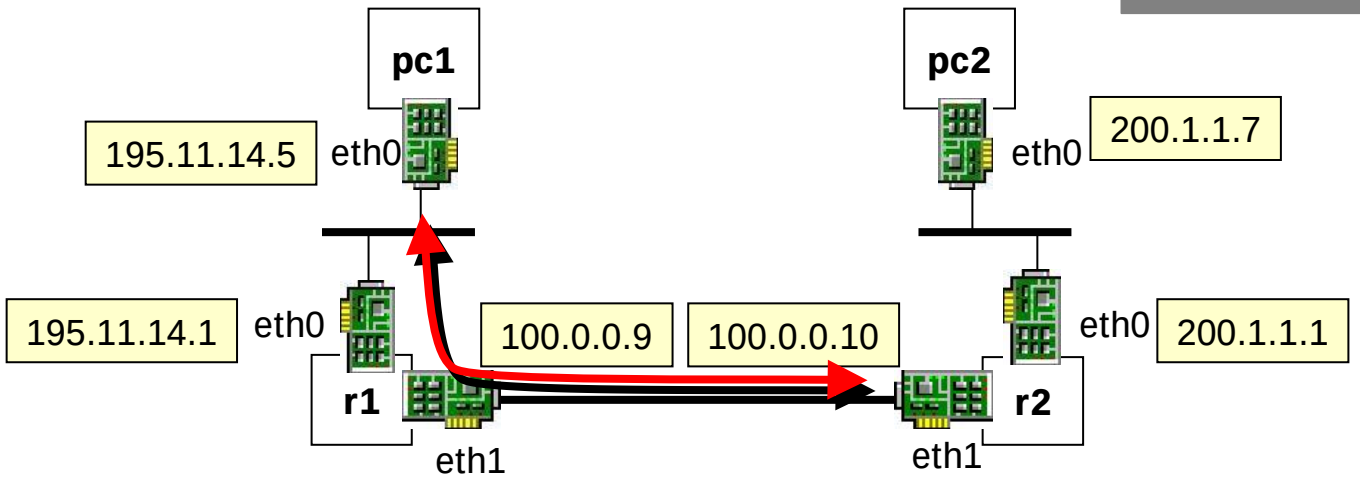
```
pc1:~# ping 100.0.0.10
PING 100.0.0.10 (100.0.0.10) 56(84) bytes of data.

--- 100.0.0.10 ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6105ms

pc1:~#
```

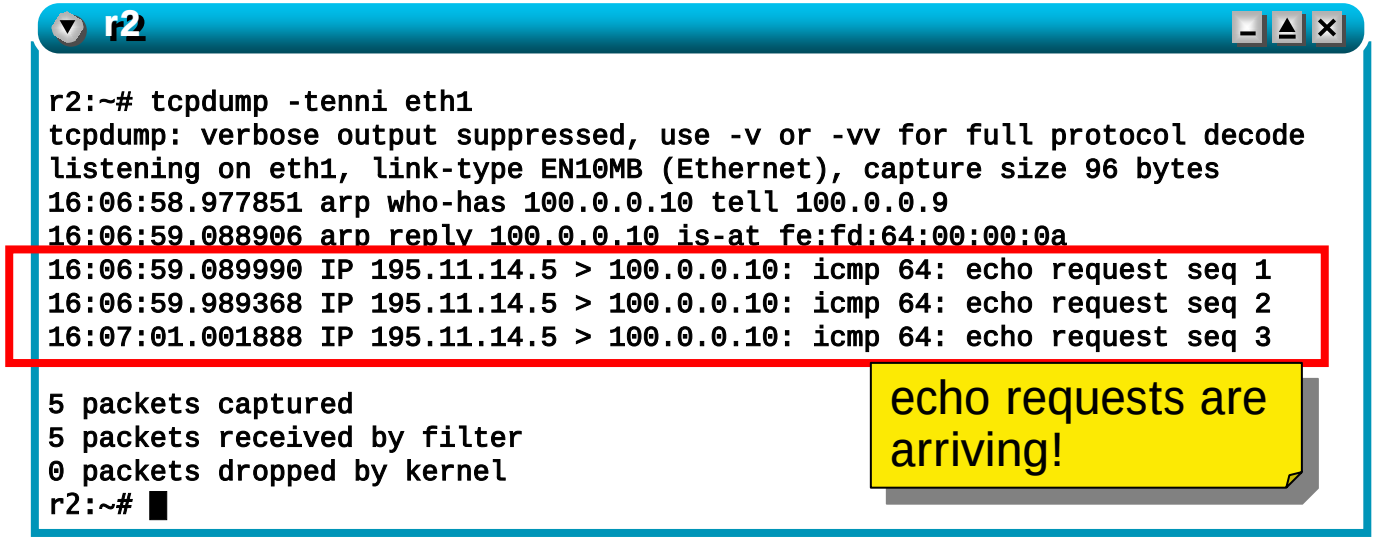
interfaces on **r2** seem unreachable!

can you tell why?



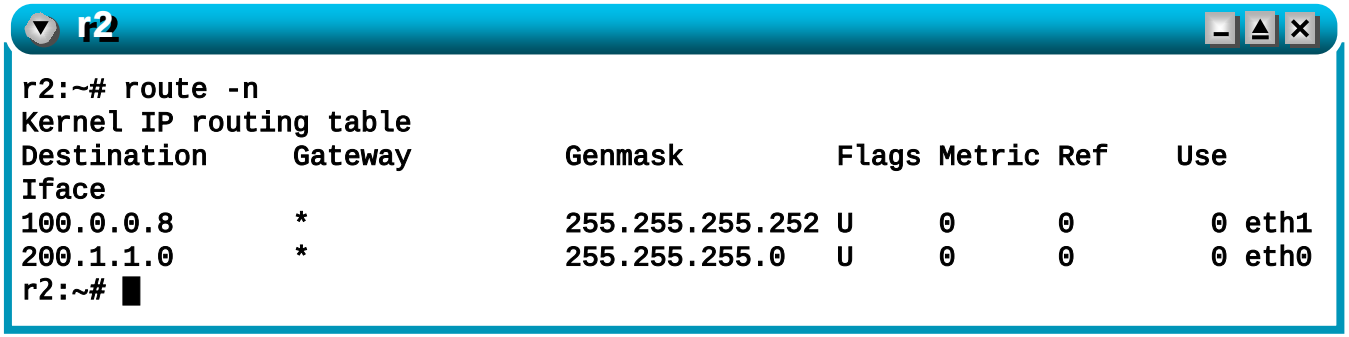
Step 4 – Let's inspect the network

- Do echo request packets reach **`r2`**?
- Let's check…
  - While pinging from **`pc1`** 100.0.0.10 sniff on interface **`eth1`** of **`r2`**

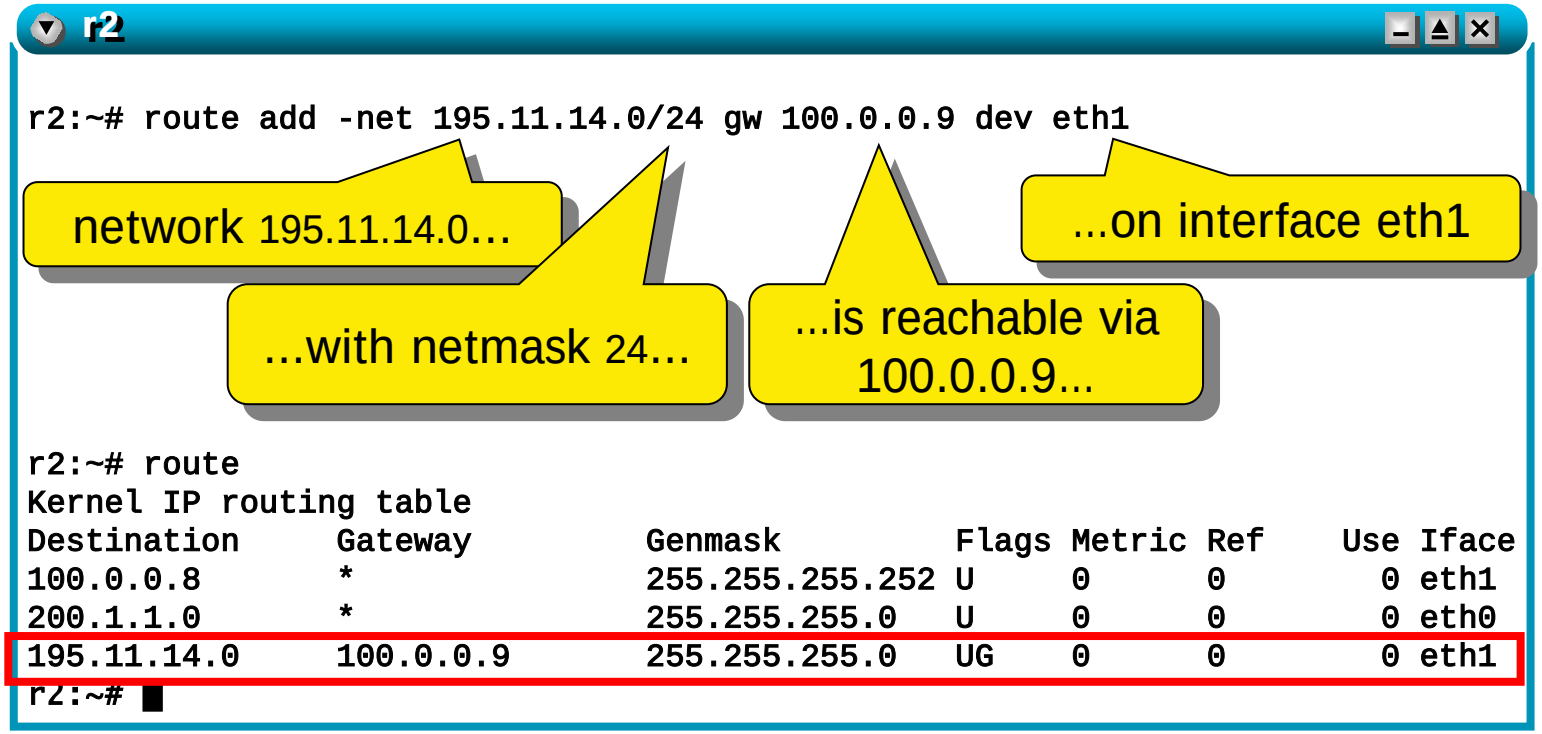


Step 4 – **r2**'s routing table

- **pc1**'s address is 195.11.14.5
- **r2** does not know how to reach such an address.
- Echo requests arrive to **r2** but **r2** does not know where echo replies should be forwarded!
- Somebody should teach **r2** how to reach **pc1**
- We may insert a static route into the routing table of **r2**

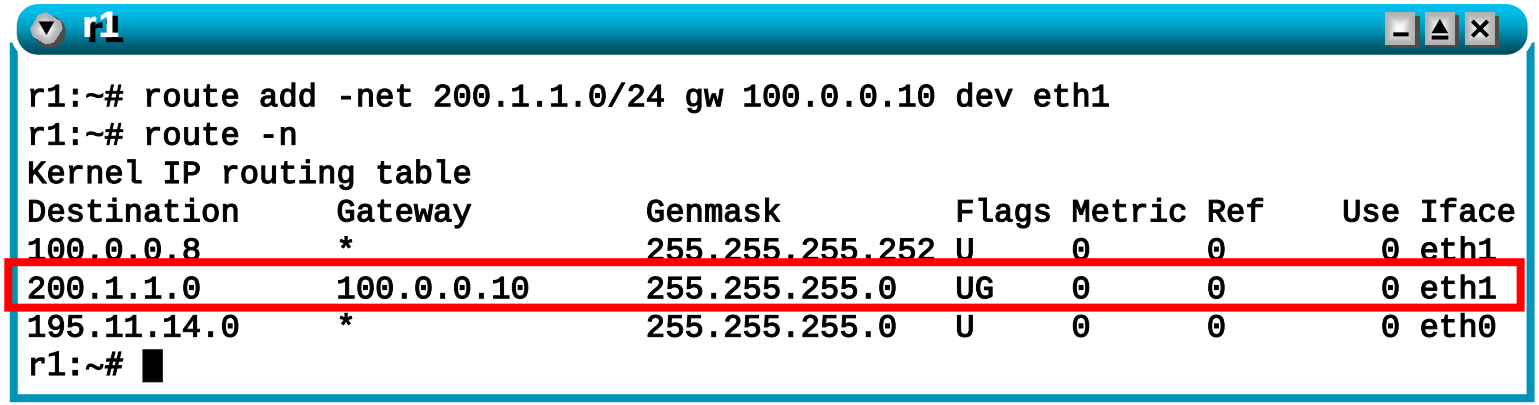


Step 5 – Configuring a static route



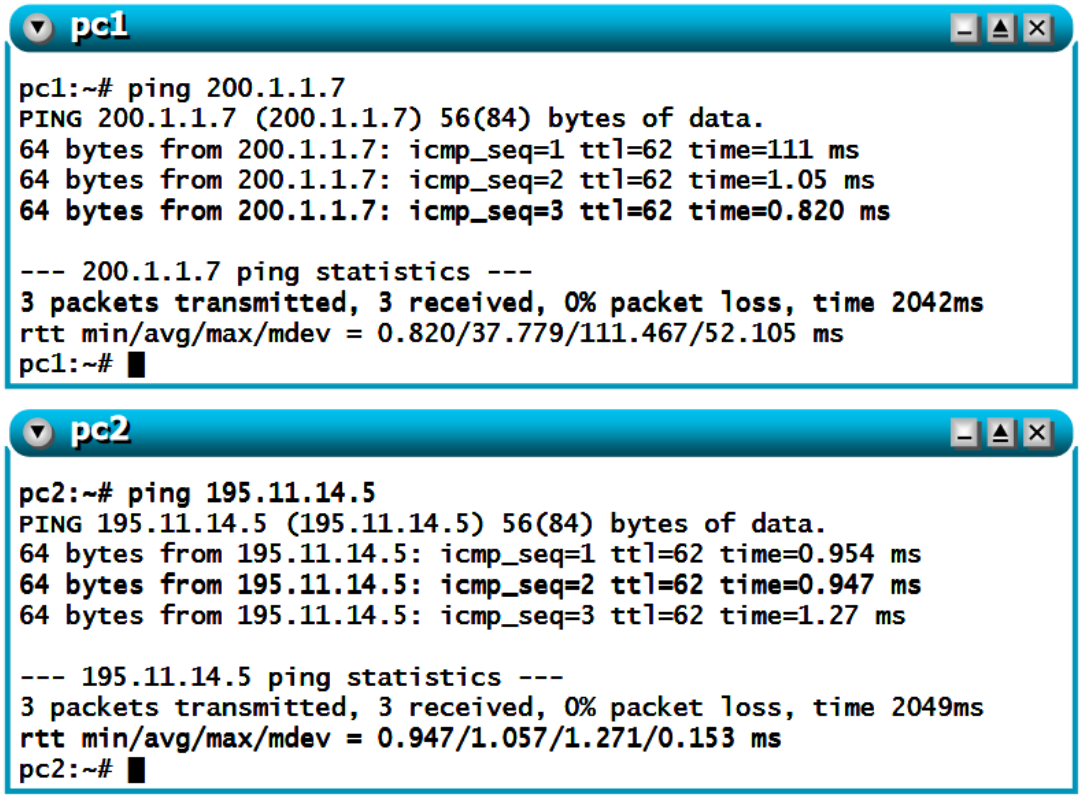
Step 5 – Configuring a static route

- A similar configuration should be deployed on **`r1`**



Step 5 – Testing static routes

- The PCs can now reach each other



Reporting

- Please deliver the following items to the UPEL system using your account
  1. A photocopy or a screenshot showing the output of the following commands:
     - **`route`** executed on router r<LAB-ID>
     - **`ping 200.1.1.7`** executed on pc1
     - **`ping 195.11.14.5`** executed on pc2

Obligatory exercises

- The default route can be statically configured by using

```
route add default gw 195.11.14.1 dev eth0
```

- Can you give a command to configure a static route that is equivalent to the default route?

```
route add -net __/__ gw __ dev __
```

Obligatory exercises

- Not all the routing tables contain a default route
- The network of this lab is so simple that routers **`r1`** and **`r2`** can be also configured to **exclusively** use default routes
- Try such a configuration and test it