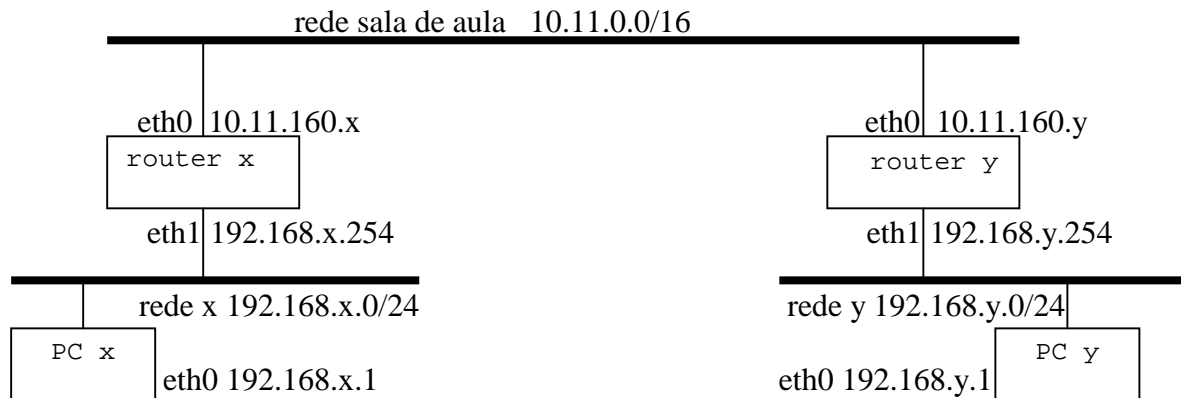


LAB04

Configuração de um router

Pretende-se construir a seguinte rede:



NOTA: Para se realizar este laboratório os alunos tem que se organizar em **dois grupos: o grupo x e o grupo y**. Os números x e y estão assinalados nos routers do laboratório.

O router x (server x) tem 2 placas de rede e vai fazer de router entre a rede da sala (que pertence à rede das salas de aula) e a rede 192.168.x.0/24 composta por um PC que **é o portátil do aluno**.

De igual modo o router y faz o encaminhamento entre a rede da sala e a rede 192.168.y.0/24

Os passos a seguir indicados referem-se ao router x. **O grupo que controla o router y deve fazer os mesmos passos.**

1. Liga o portátil com 1 cabo cruzado à placa eth1 do router x (substitui x pelo número do teu grupo). O router x já deve encontrar-se ligado à rede da sala através da interface eth0.
2. Inserir (se necessário) os módulos dos device drivers no kernel do router. Normalmente este passo já foi realizado na fase de arranque do sistema operativo, se não:
(a ordem é importante!)
#insmod 8139too
#insmod tulip

verifica que os drivers estão no kernel
#lsmod

3. Configuração dos interfaces no router x

Numero:

Nome:

Data:

```
#ifconfig eth0 down
#ifconfig eth1 down
#ifconfig eth0 _____ netmask _____ broadcast_____ up
```



Substitui pelo que está escrito no computador!

```
#ifconfig eth1 _____ netmask _____ broadcast_____ up
```

verifica a configuração:

```
#ifconfig
```

4. Configuração da tabela de routing no router.

Normalmente este passo já foi realizado automaticamente pelo sistema operativo.

Verifica a tabela de routing actual

```
#route -n
```

Configura, se necessário, as entradas para as redes às quais o router está directamente ligado:

```
#route add -net _____ netmask_____ dev eth0
```

```
#route add -net_____ netmask_____ dev eth1
```

Configura, se necessário, a entrada de defeito:

```
#route add default gw _____
```

Obrigatório: configura a entrada para a rede y no router x (e vice-versa):

```
#route add -net_____ netmask_____ gw_____
```

5. Configuração da interface *no teu portátil* (PC x)

- em Linux:

Utiliza a janela correspondente à tua versão do Linux. Em alternativa, desliga o serviço Network Manager

```
#service network stop
```

6. #ifconfig eth0_____ netmask_____ broadcast_____ up

7. Configuração da tabela de routing no PC x:

```
#route add default gw _____
```

verifica a tabela de routing:

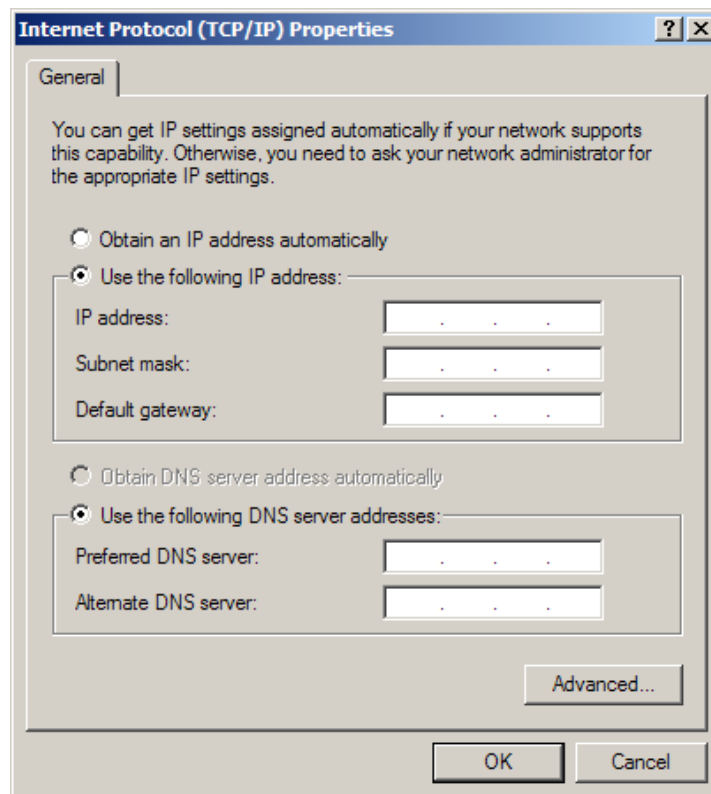
```
#route -n
```

Numero:

Nome:

Data:

- em Windows debes utilizar a janela correspondente à tua versão do sistema operativo para configurares a interface com um **IP estático**:



8. Neste momento o portátil x na rede x já sabe como comunicar com a rede da sala.

No portátil 192.168.x.1 faz:

```
#ping 192.168.x.254
```

que é a gateway da rede x. Recebes o eco? _____

```
#ping 10.11.160.x
```

que é o endereço do router x na rede da sala. Recebes o eco? _____
Porquê? _____

9. Activa agora a funcionalidade de router (IP forwarding) no router x:

```
#echo 1 > /proc/sys/net/ipv4/ip_forward
```

confirma que a funcionalidade de router ficou activada no router x:

```
#cat /proc/sys/net/ipv4/ip_forward
```

10. No portátil 192.168.x.1 faz ping novamente:

```
#ping 10.11.160.x
```

Numero:

Nome:

Data:

Recebes o eco? _____

11. No portátil 192.168.x.1 faz ping ao router y:

```
#ping 10.11.160.y
```

Recebes o eco? _____

12. No portatil 192.168.x.1 faz ping ao portatil y

```
#ping 192.168.y.1
```

Recebes o eco? _____

13. No portatil 192.168.x.1 faz ping ao router z (10.11.160.z), isto é faz ping a um router da sala que não seja o router x nem o router y. Recebes o eco? _____
Porquê? _____

14. No portatil 192.168.y.1 verifica que há um serviço ssh a correr (em alternativa um servidor web que abre a porta 80)

```
#netstat -an | grep ':22'
```

15. No portatil 192.168.x.1 faz uma sessão ssh para o serviço que está a correr no portatil y:

```
#ssh 192.168.y.1
```

Recebes o login prompt? _____

16. Fim deste laboratório. Devolve o cabo cruzado. Desliga o router e o monitor.