Serviços de Rede 1 –
Lesson 11 - Practices

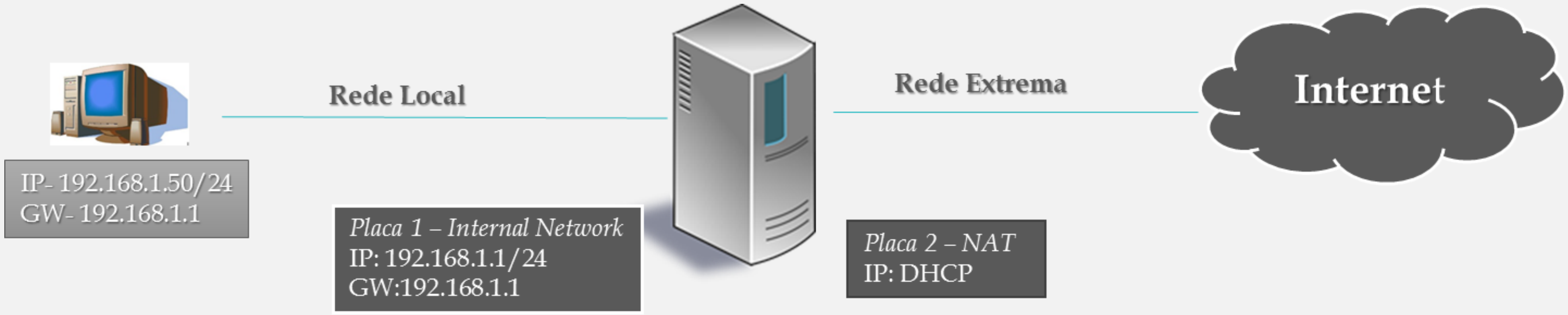
2019-2020

Instituto Politécnico de Coimbra

Departamento de Engenharia Informática

Pre - Requirements -Exercise 1

- Have the previous class topology installed and working.
- In Moodle is the Squid configuration file.



IP- 192.168.1.50/24
GW- 192.168.1.1

Rede Local

Placa 1 – Internal Network
IP: 192.168.1.1/24
GW:192.168.1.1

Rede Extrema

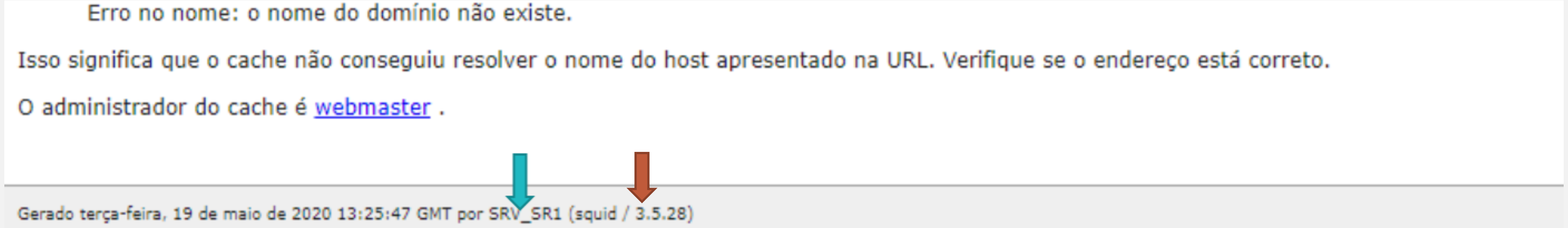
Placa 2 – NAT
IP: DHCP

Internet

Exercise 1 - Squid advanced configuration

Exercise

- Remove the version of the squid that is running in case of an error. Generate an error (for example access to www.sr1.pt) and see if that name appears.

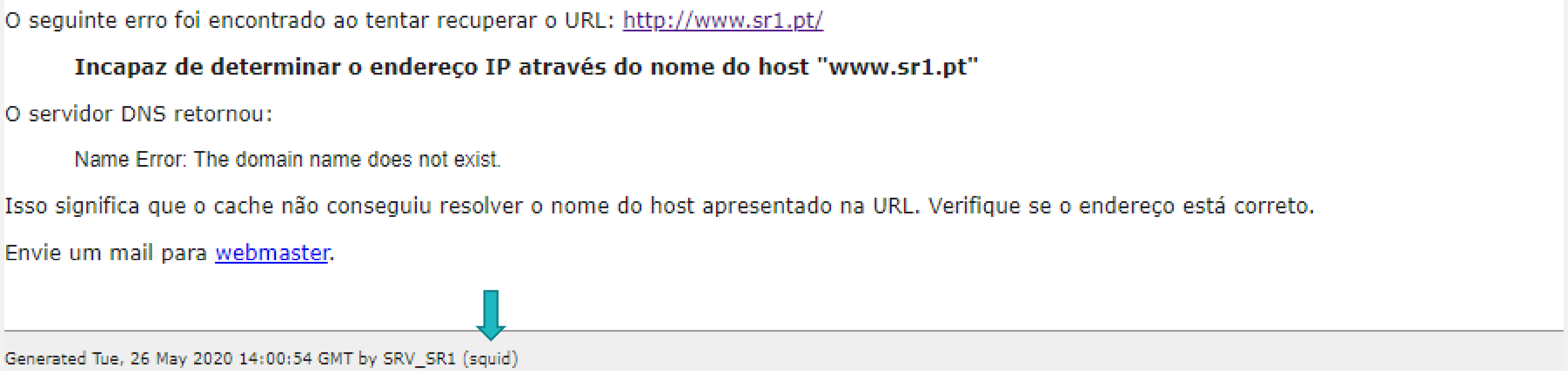


Erro no nome: o nome do domínio não existe.

Isso significa que o cache não conseguiu resolver o nome do host apresentado na URL. Verifique se o endereço está correto.

O administrador do cache é webmaster .

Gerado terça-feira, 19 de maio de 2020 13:25:47 GMT por SRV_SR1 (squid / 3.5.28)



O seguinte erro foi encontrado ao tentar recuperar o URL: http://www.sr1.pt/

Incapaz de determinar o endereço IP através do nome do host "www.sr1.pt"

O servidor DNS retornou:

Name Error: The domain name does not exist.

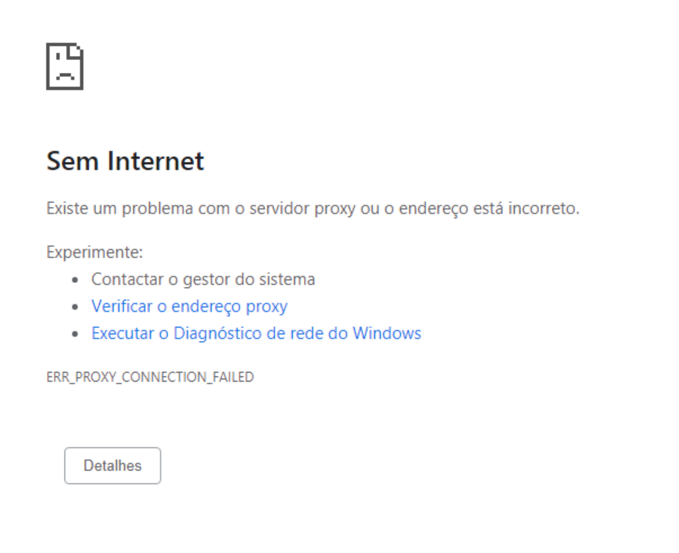
Isso significa que o cache não conseguiu resolver o nome do host apresentado na URL. Verifique se o endereço está correto.

Envie um mail para webmaster.

Generated Tue, 26 May 2020 14:00:54 GMT by SRV_SR1 (squid)

Exercise

- In the acl you created in the previous class, change your name to extern. Restart the squid services and see what happens. You must be without access to the network.
- Through the terminal window, identify which line the error is occurring on.
- Correct the error and check that everything is running as you had configured.
- Also in the terminal window, identify which version of the SQUID you are using.
- See your log files ... They must have grown very significantly. Erase the access logs.



Sem Internet

Existe um problema com o servidor proxy ou o endereço está incorreto.

Experimente:

- Contactar o gestor do sistema
- Verificar o endereço proxy
- Executar o Diagnóstico de rede do Windows

ERR_PROXY_CONNECTION_FAILED

Exercise

- In squid / var / cache create a new directory called SR1. This directory will be used to save the files of your server's cache function.
- You must activate the cache function with the following characteristics:
  - Board / var / cache / sr1
  - Type - USF
  - 512MB of space, 128 directories and 256 subdirectories
- Proceed to create this function.
- Validate in the SR1 directory that all folders were created as defined in the command.

How To

- One of the most important items in the squid.conf configuration file are access control lists, or ACLs.
- It is possible to create ACLs with different patterns of restriction and access, such as, for example, releasing and / or blocking internet access for a specific computer, computer network or unwanted websites. We have thus:
  - **Source ACLs**: are the rules that define the IPs that may have access or restriction to the defined rules;
  - **Destination ACLs:** define which destination may or may not be accessible. The control can be per site, for a network or using regular expressions;
  - **Time ACLs:** allow you to release or block access according to times defined in the rules;
  - **ACLs using blacklist:** allows controlling granularly the blocking of accesses through specific words;
  - **ACLs using whitelist:** allows granular control of the release of domains containing words from the blacklist.

- The command is:

acl [*nome*] [*tipo*] [*argumento*]

- Where:
- nome - is the name of the ACL
- tipo - it is the type of acl that could be created (see the possible types on the next slide)
- argumento - options that can be added

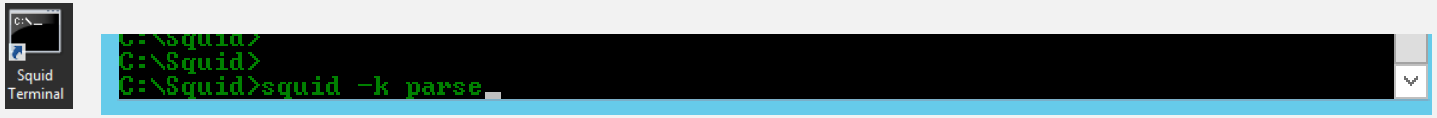
src	
time	
urlpath_regex	
url_regex	
dstdomain	
proxy_auth	
arp	
maxconn	
proto	
port	

- ACLs define the types of controls we want to do. However, even at this stage, requests are not processed, and it is necessary to configure the http_access command.

- Once the source and destination ACLs have been created, it is necessary to define what will be released (allow) and / or what will be blocked (deny), with the command:

http_access [*alow/denny*] [*nome da ACL*]

- As the SQUID configuration is done in the squid.conf file, it is sometimes not easy to diagnose an error. One hypothesis is the command **squid –k** parse in squid's terminal window and analyzing the errors.



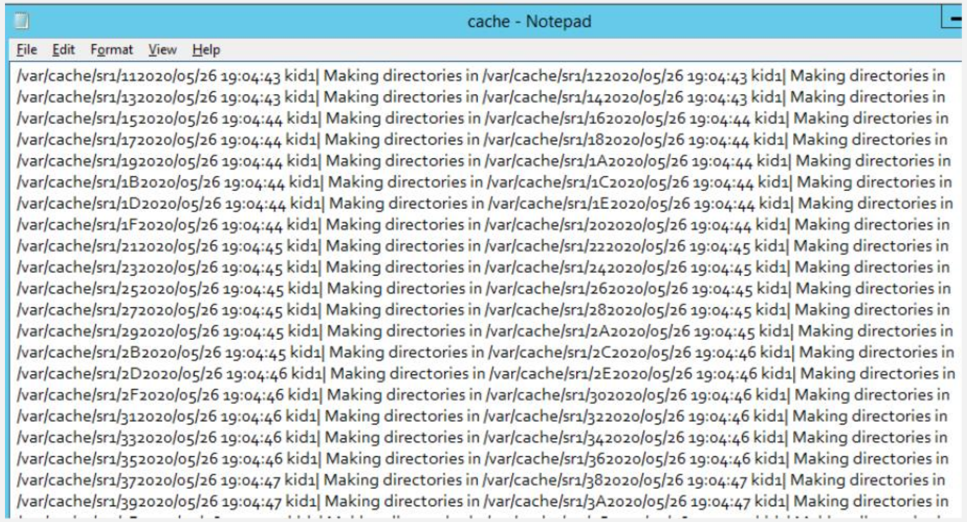
```
C:\squid>
C:\Squid>
C:\Squid>squid -k parse_
```

- You can find out what commands are available, you must use the **squid - v** command.

- **cache.log**
  - This file contains informational messages, formatted for humans, about Squid's operation. The file name is defined by the cache_log command. Under normal conditions, the file grows about 10 to 100 KB per day.
- **access.log**
  - This file contains an entry for all HTTP and (optionally) ICP transactions made by Squid clients. The file name is defined by the cache_access_log command. Under normal conditions the file grows at a rate of 100-200 bytes per transaction.

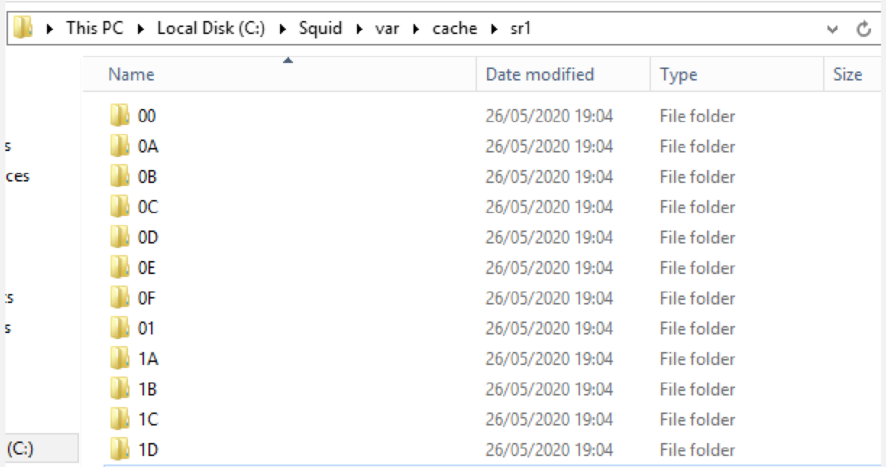
- To activate the cache function, use the command:
  - cache_dir [type] [path] [disk size] [number of directories] [number of subdirectories]
- You must also define the memory space you will use for the cache, using the command
  - cache_mem [value]
- Then run the command squid –k parse in squid's terminal window and analyze the possible errors.
- Then you must run the command squid –z so that the program creates in the directory defined for the cache the swap dir. You only need to do this the first time you are activating the cache function.

- You can validate the creation of these directories in the log file.



```
/var/cache/sr1/112020/05/26 19:04:43 kid1| Making directories in /var/cache/sr1/122020/05/26 19:04:43 kid1| Making directories in
/var/cache/sr1/132020/05/26 19:04:43 kid1| Making directories in /var/cache/sr1/142020/05/26 19:04:43 kid1| Making directories in
/var/cache/sr1/152020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/162020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/172020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/182020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/192020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/1A2020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/1B2020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/1C2020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/1D2020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/1E2020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/1F2020/05/26 19:04:44 kid1| Making directories in /var/cache/sr1/202020/05/26 19:04:44 kid1| Making directories in
/var/cache/sr1/212020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/222020/05/26 19:04:45 kid1| Making directories in
/var/cache/sr1/232020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/242020/05/26 19:04:45 kid1| Making directories in
/var/cache/sr1/252020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/262020/05/26 19:04:45 kid1| Making directories in
/var/cache/sr1/272020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/282020/05/26 19:04:45 kid1| Making directories in
/var/cache/sr1/292020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/2A2020/05/26 19:04:45 kid1| Making directories in
/var/cache/sr1/2B2020/05/26 19:04:45 kid1| Making directories in /var/cache/sr1/2C2020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/2D2020/05/26 19:04:46 kid1| Making directories in /var/cache/sr1/2E2020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/2F2020/05/26 19:04:46 kid1| Making directories in /var/cache/sr1/302020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/312020/05/26 19:04:46 kid1| Making directories in /var/cache/sr1/322020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/332020/05/26 19:04:46 kid1| Making directories in /var/cache/sr1/342020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/352020/05/26 19:04:46 kid1| Making directories in /var/cache/sr1/362020/05/26 19:04:46 kid1| Making directories in
/var/cache/sr1/372020/05/26 19:04:47 kid1| Making directories in /var/cache/sr1/382020/05/26 19:04:47 kid1| Making directories in
/var/cache/sr1/392020/05/26 19:04:47 kid1| Making directories in /var/cache/sr1/3A2020/05/26 19:04:47 kid1| Making directories in
```

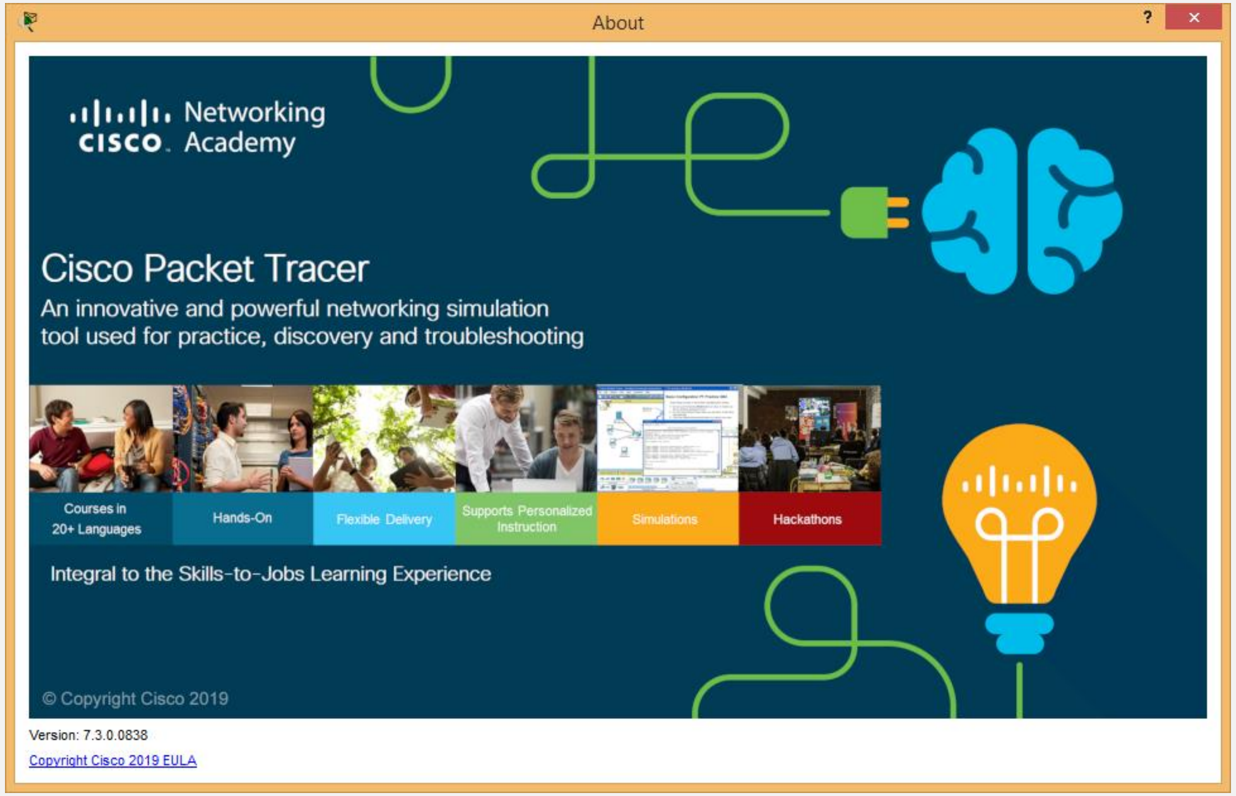
- Or validate your creation with Windows explorer.



Name	Date modified	Type	Size
00	26/05/2020 19:04	File folder	
0A	26/05/2020 19:04	File folder	
0B	26/05/2020 19:04	File folder	
0C	26/05/2020 19:04	File folder	
0D	26/05/2020 19:04	File folder	
0E	26/05/2020 19:04	File folder	
0F	26/05/2020 19:04	File folder	
01	26/05/2020 19:04	File folder	
1A	26/05/2020 19:04	File folder	
1B	26/05/2020 19:04	File folder	
1C	26/05/2020 19:04	File folder	
1D	26/05/2020 19:04	File folder	

Pre – Requisitos -Exercício 2

- Ter instalado o *Cisco Packet Tracer* versão 7.3.0



About

Networking Academy

Cisco Packet Tracer

An innovative and powerful networking simulation tool used for practice, discovery and troubleshooting

Integral to the Skills-to-Jobs Learning Experience

© Copyright Cisco 2019

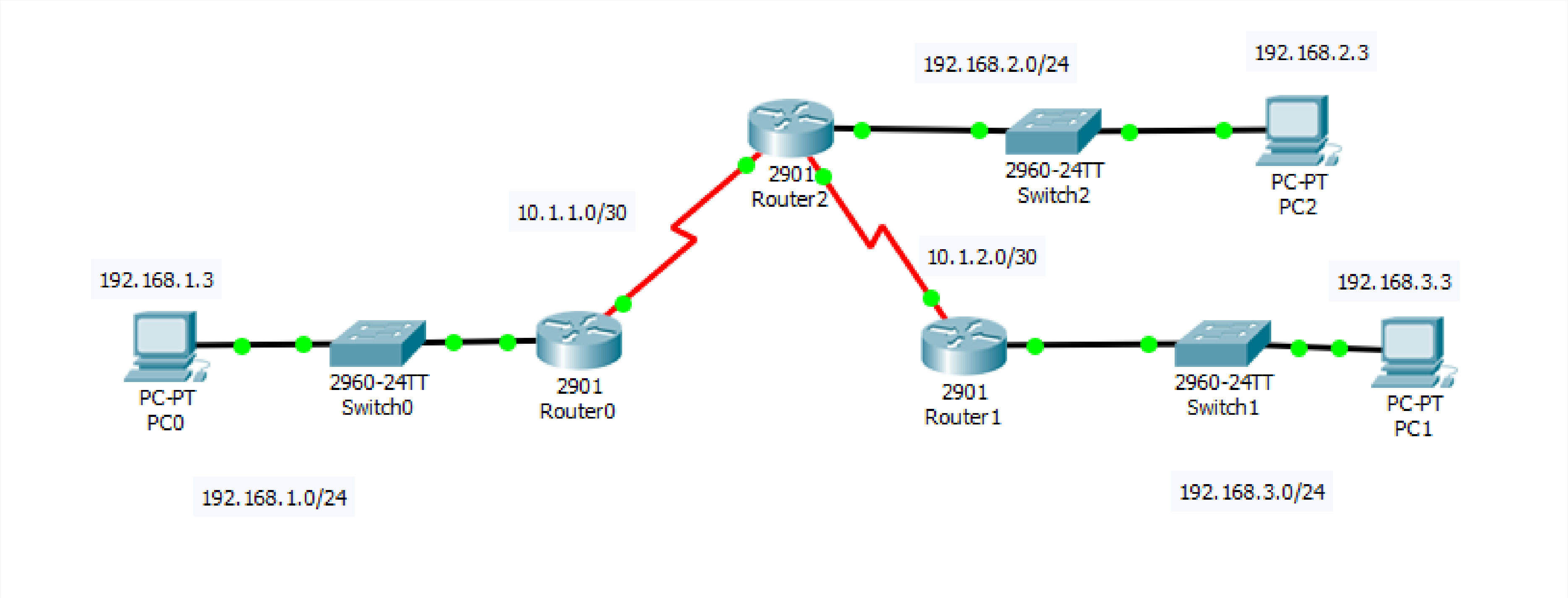
Version: 7.3.0.0838

Copyright Cisco 2019 EULA

Exercise 2 - VPN with GRE tunnel in Cisco environment

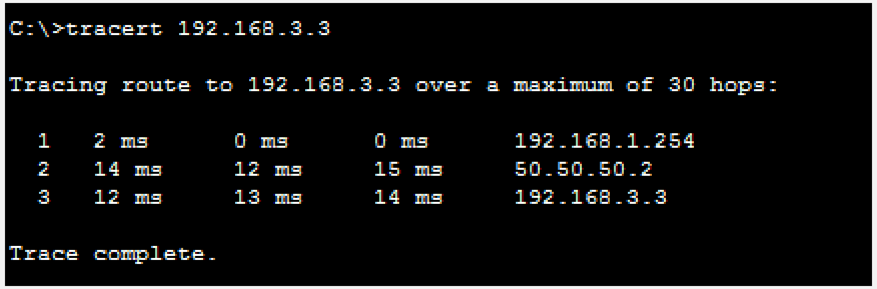
Exercise

The company SR1.SA needs to connect the headquarters network (192.168.1.0) to the delegation network (192.168.3.0) using the telecommunications connection already installed.
In a first phase, the company wants this connection to be made over a GRE tunnel.
Consider the following topology:

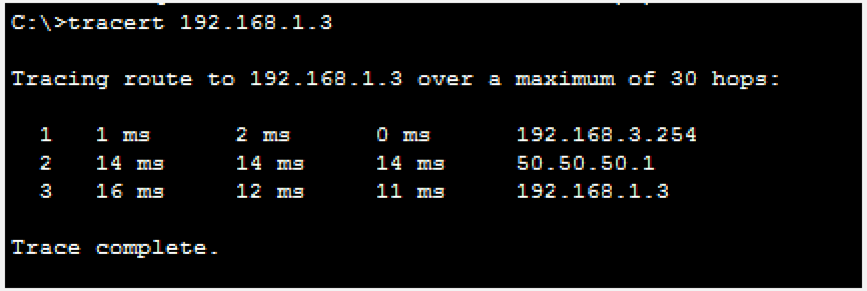


Exercise 1

- Make the topology indicated in the previous image. Save the simulation as VPN_GRE.
- Put the IP addresses of the different devices in a fixed way and according to the networks indicated in the image.
- Ensure that PC0 and PC1 are able to have connectivity to PC2.
- Create a VPN between Routers 0 and 1 based on a GRE tunnel. The IP of the tunnel will be the network 50.50.50.0/24.
- Do the **tracert** from PC0 to PC1 and vice versa.



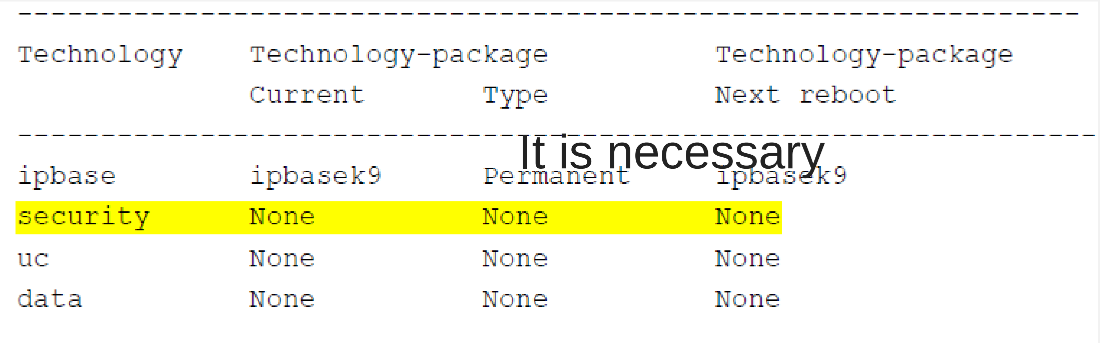
```
C:\>tracert 192.168.3.3

Tracing route to 192.168.3.3 over a maximum of 30 hops:

  1   2 ms      0 ms      0 ms      192.168.1.254
  2   14 ms     12 ms     15 ms     50.50.50.2
  3   12 ms     13 ms     14 ms     192.168.3.3

Trace complete.
```



```
C:\>tracert 192.168.1.3

Tracing route to 192.168.1.3 over a maximum of 30 hops:

  1   1 ms      2 ms      0 ms      192.168.3.254
  2   14 ms     14 ms     14 ms     50.50.50.1
  3   16 ms     12 ms     11 ms     192.168.1.3

Trace complete.
```

How To

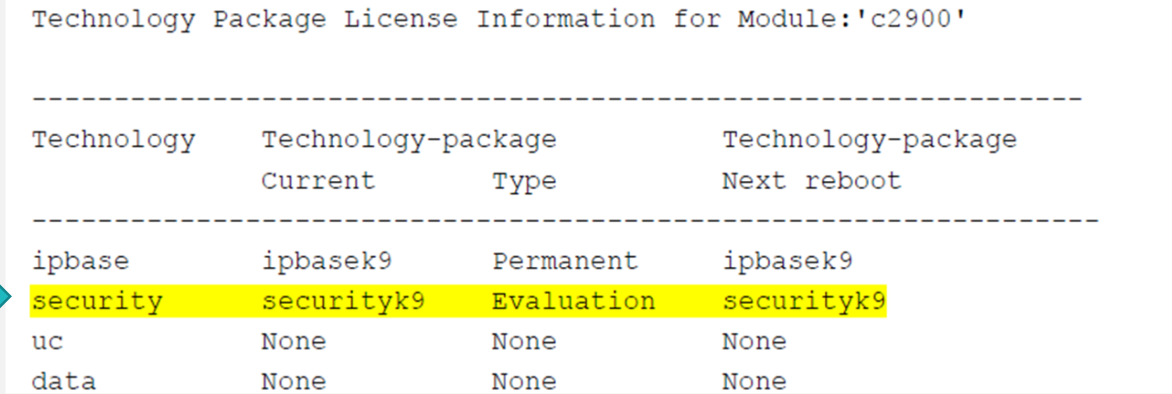
- You may have to activate Security Technology Package license on some routers.
  - For this run tej command :***Show version*** in **Enable mode**:



```
-----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      None          None           None
uc            None          None           None
data          None          None           None
```

It is necessary

It is necessary



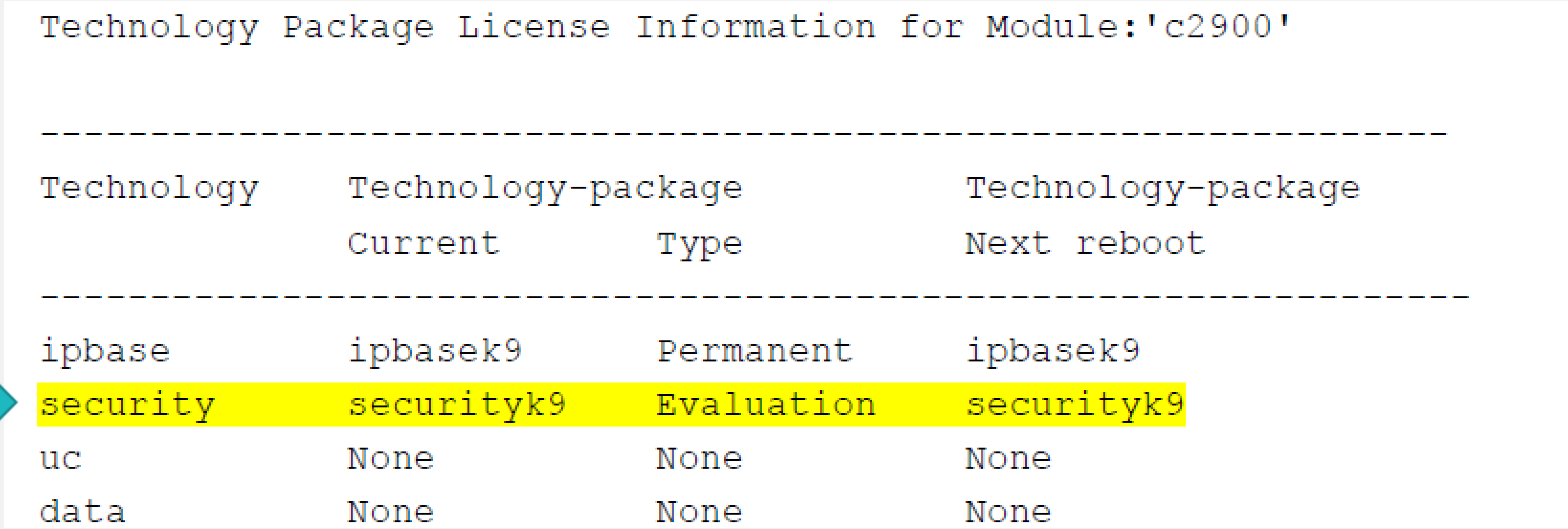
```
Technology Package License Information for Module:'c2900'

----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
-----------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      securityk9    Evaluation     securityk9
uc            None          None           None
data          None          None           None
```

It is not necessary

  - License should not be installed
  - Enter **Configuration mode** and do:

license boot module cXXXX technology-package securityk9

- Save the configuration.
- Do **reload**.
- Make ***show version*** and you must have activated Security Technology Package license.



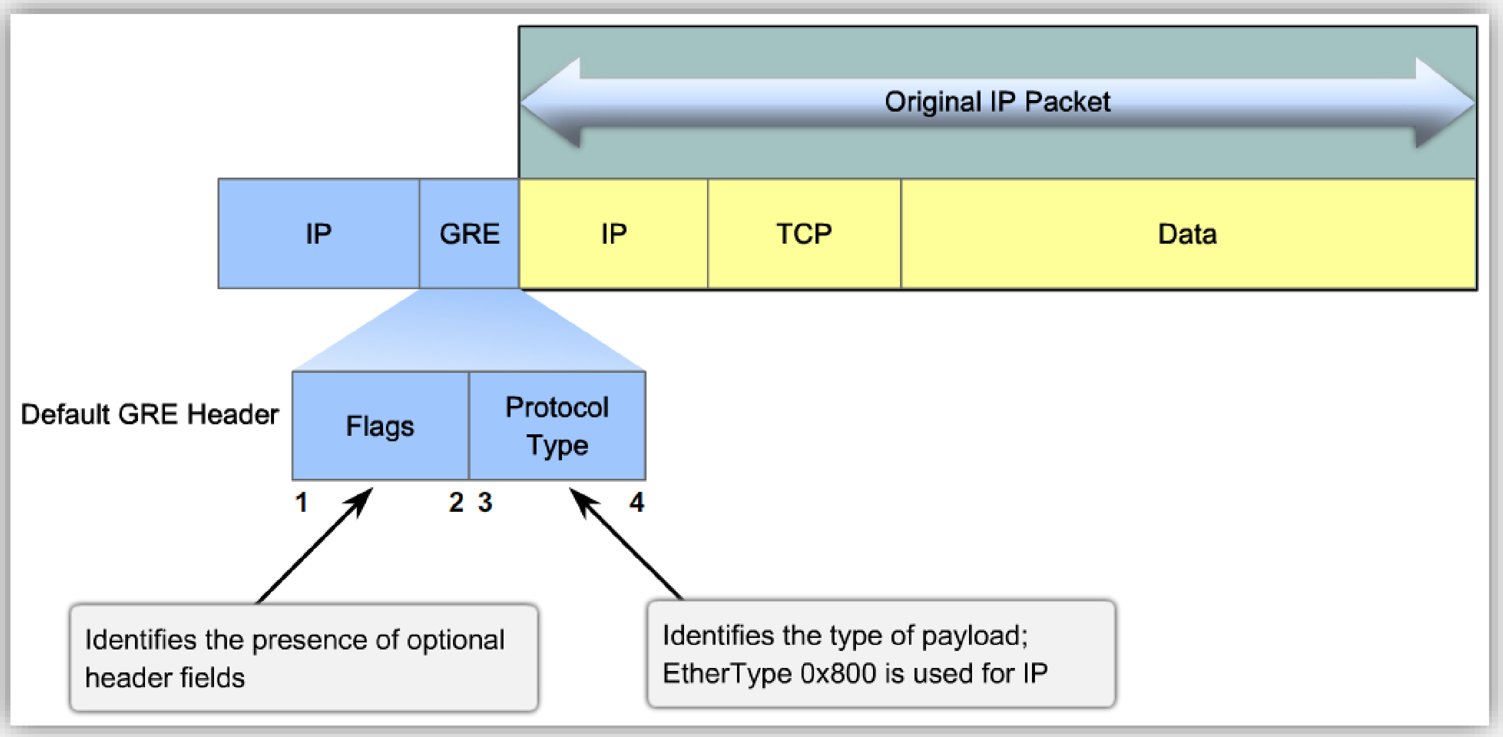
```
Technology Package License Information for Module:'c2900'

-----------------------------------------------------------------
Technology    Technology-package           Technology-package
              Current       Type           Next reboot
------------------------------------------------------------------
ipbase        ipbasek9      Permanent      ipbasek9
security      securityk9    Evaluation     securityk9
uc            None          None           None
data          None          None           None
```

Tunneling

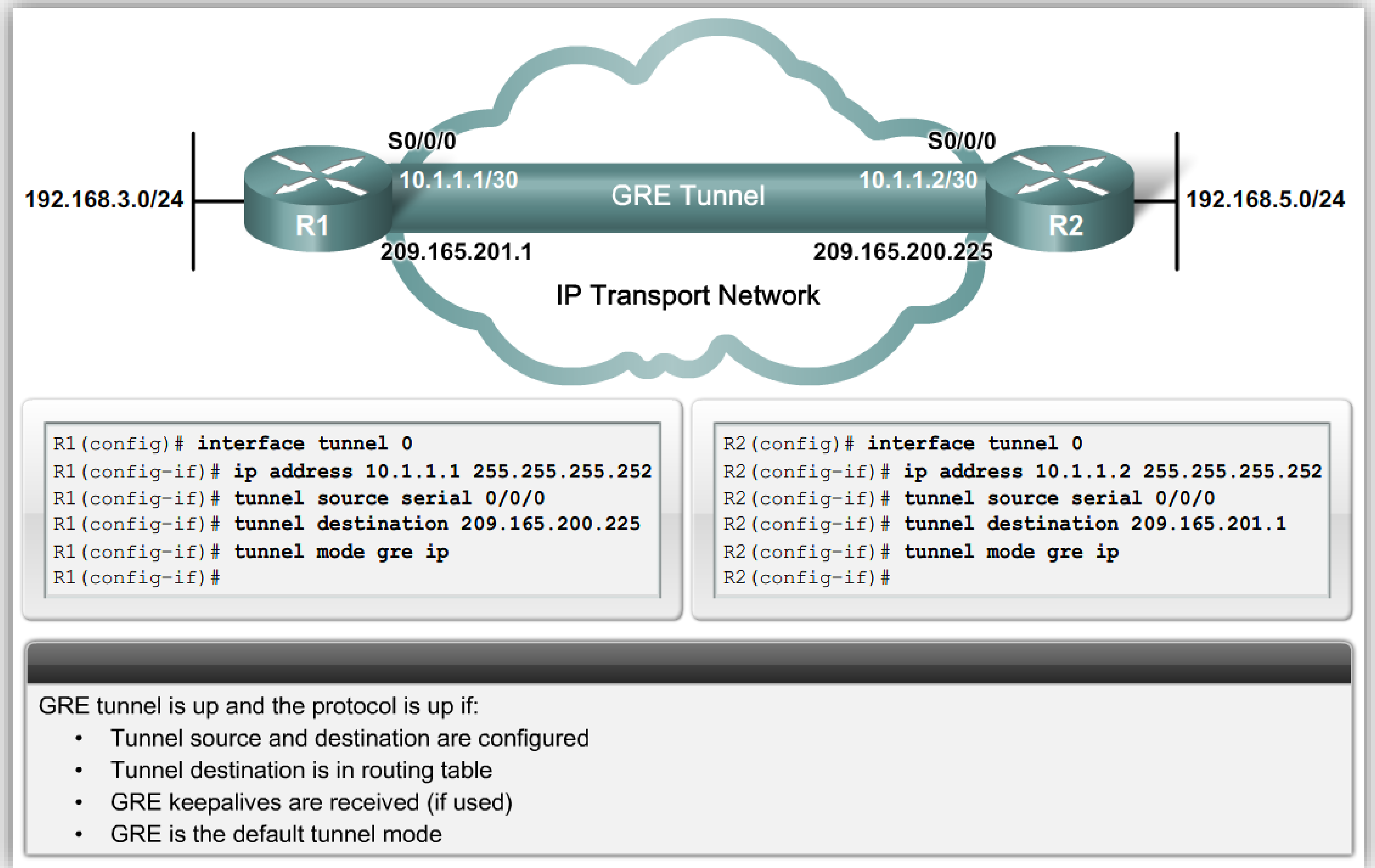
- For a tunnel to be established it is necessary that the server and client use the same protocol.
- For the establishment of the tunnel, two phases are necessary:
  - **Tunnel establishment**
    - Negotiation of variables, address, encryption and compression.
  - **Streaming**
    - Encapsulation and encryption.
    - Send.
    - Decapsulation and decryption.

GRE - *Generic Routing Encapsulation*

- IP packets are encapsulated in a GRE packet
  - Implies an additional payload of at least 24 bytes


Original IP Packet
IP
GRE
IP
TCP
Data
Default GRE Header
Flags
Protocol Type
1
2
3
4
Identifies the presence of optional header fields
Identifies the type of payload; EtherType 0x800 is used for IP

GRE



```
R1(config)# interface tunnel 0
R1(config-if)# ip address 10.1.1.1 255.255.255.252
R1(config-if)# tunnel source serial 0/0/0
R1(config-if)# tunnel destination 209.165.200.225
R1(config-if)# tunnel mode gre ip
R1(config-if)#
```

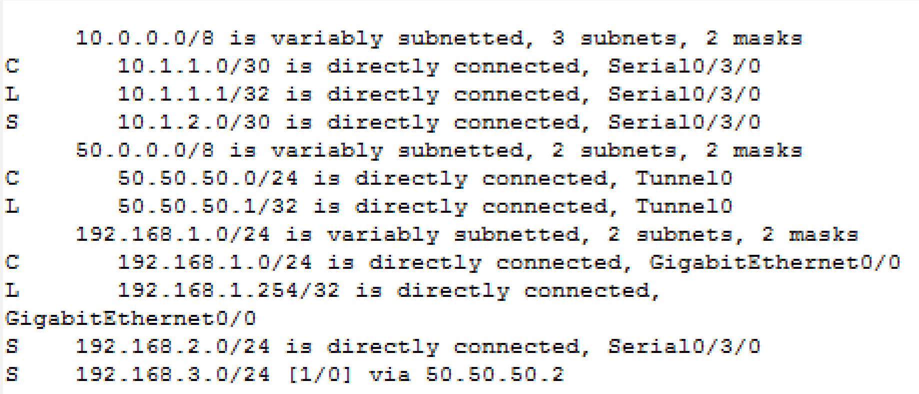
```
R2(config)# interface tunnel 0
R2(config-if)# ip address 10.1.1.2 255.255.255.252
R2(config-if)# tunnel source serial 0/0/0
R2(config-if)# tunnel destination 209.165.201.1
R2(config-if)# tunnel mode gre ip
R2(config-if)#
```

GRE tunnel is up and the protocol is up if:

- Tunnel source and destination are configured
- Tunnel destination is in routing table
- GRE keepalives are received (if used)
- GRE is the default tunnel mode

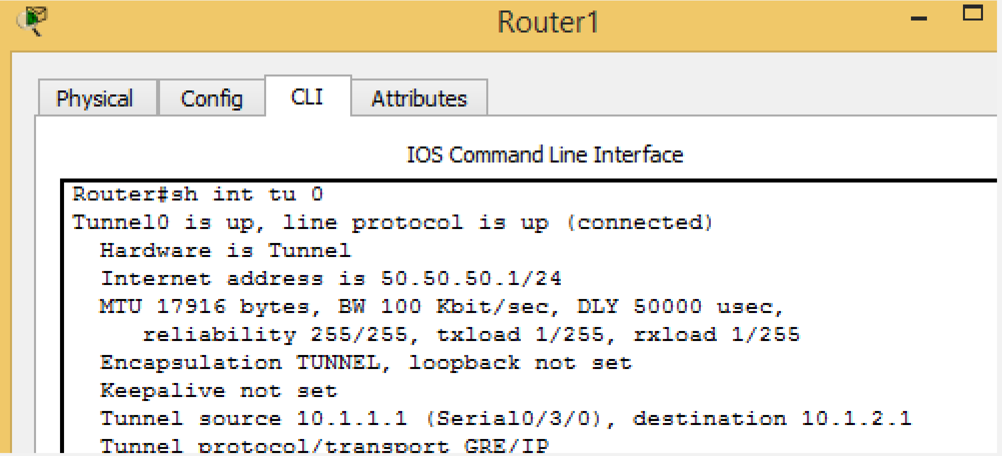
GRE

- Do not forget to make the necessary routes.



```
     10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C       10.1.1.0/30 is directly connected, Serial0/3/0
L       10.1.1.1/32 is directly connected, Serial0/3/0
S       10.1.2.0/30 is directly connected, Serial0/3/0
     50.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C       50.50.50.0/24 is directly connected, Tunnel0
L       50.50.50.1/32 is directly connected, Tunnel0
     192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.1.0/24 is directly connected, GigabitEthernet0/0
L       192.168.1.254/32 is directly connected,
GigabitEthernet0/0
S    192.168.2.0/24 is directly connected, Serial0/3/0
S    192.168.3.0/24 [1/0] via 50.50.50.2
```

- You should check if your interface is up



```
Router#sh int tu 0
Tunnel0 is up, line protocol is up (connected)
  Hardware is Tunnel
  Internet address is 50.50.50.1/24
  MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 10.1.1.1 (Serial0/3/0), destination 10.1.2.1
  Tunnel protocol/transport GRE/IP
```

Referências

- https://www.devmedia.com.br/implementacao-de-um-proxy-web-com-squid-revista-infra-magazine-8/26302 - acedido em Maio de 2020
- https://ipcisco.com/gre-tunnel-configuration-with-packet-tracer/ - acedido em Maio de 2020
- Cisco Networking Academy – Packet Tracer – Configuring VPNs