

ESCOLA POLITÈCNICA  
SUPERIOR



XARXES I COMUNICACIONS

PRÀCTICA 3

---

## Full Redundant SLB & SNMP

---

*Students:*

Nil Agut Marín  
Jaume Giralt Barbé

*Professor:*

Fernández Camon, Cèsar

11 de juny de 2017

## Índex

<b>1</b>	<b>Objectius</b>	<b>3</b>
<b>2</b>	<b>Full Redundant SLB</b>	<b>3</b>
2.1	Topologia de la xarxa . . . . .	3
2.2	Principals problemes de configuració i com ho hem solucionat . . . . .	3
<b>3</b>	<b>Proves de connexió</b>	<b>4</b>
<b>4</b>	<b>SNMP</b>	<b>7</b>
4.1	Introducció . . . . .	7
4.2	Executar el programa . . . . .	7
4.3	Estructura del programa . . . . .	7
4.4	Procediments . . . . .	7

## Índex de figures

1	SLB - Topologia de la xarxa a efectuar l'exercici . . . . .	3
2	Connexions mitjançant el protocol HTTP - R1 . . . . .	5
3	Connexions mitjançant el protocol HTTP - R2 . . . . .	5
4	Configuració GLBP SLB1 . . . . .	6
5	Configuració GLBP SLB2 . . . . .	6
6	Configuració GLBP SLB2 després caiguda SLB1 . . . . .	6

# 1 Objectius

L'objectiu principal d'aquesta pràctica és implementar **SLB** i **SNMP**.

## 2 Full Redundant SLB

### 2.1 Topologia de la xarxa

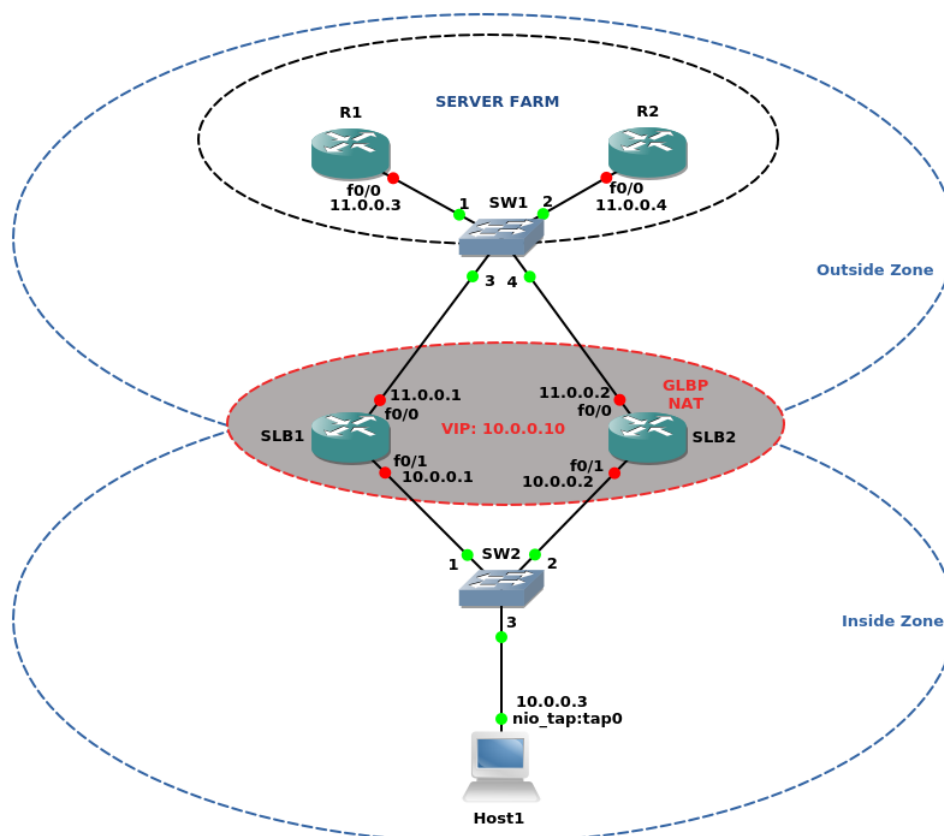


Figura 1: SLB - Topologia de la xarxa a efectuar l'exercici

Per a la realització de aquest exercici utilitzarem encaminadors **Cisco c7200**

### 2.2 Principals problemes de configuració i com ho hem solucionat

El principal problema que hem tingut en la realització d'aquesta pràctica ha sigut la connexió amb la *Server Farm* mitjançant el protocol HTTPS. Hem aconseguit realitzar connexions mitjançant el protocol HTTP al server farm. També hem aconseguit la redundància entre els dos encaminadors SLB i si un enllaç o un servidor fallava del server farm, el altre servidor responia.

Com hem dit anteriorment, el gran problema d'aquest apartat han sigut les connexions HTTPs per culpa dels certificats. El primer problema amb el qual ens vam trobar va ser que el protocol scp no és podia utilitzar en aquests encaminadors. Vam intentar exportar els certificats a la memòria del router, la nvram. Tot i així, no eram capaços d'importar els certificats. També hem intentat amb altres protocols com el tftp. Hem instal·lat el paquet i ens hem intentat connectar al servidor. Hem configurat el servei tftp, afegint `-c` al camp

TFTP\_OPTIONS per tal de poder crear arxius encara que no estiguessin. Hem configurat al encaminador que utilitzen el protocol SLB les següents comandes:

- *crypto pki trustpoint TP*
- *subject-name O=Test,CN=xarxes-lab.cat*
- *rsa keypair XARXES-KEY*
- *enrollment selfsigned*
- *revocation-check none*
- *crypto key generate rsa general-keys label XARXES-KEY modulus 1024 exportable*
- *crypto pki enroll TP*
- *crypto pki export TP pem url tftp: des PASSWORD*
- *ip http secure-trustpoint TP*
- *ip http secure-server*

Tot i així, quan intentem connectar-nos al servidor tftp que tenim en el nostre ordinador, no connecta. Hem pensat que podria ser per que faltessin rutes estàtiques, les hem posat però continua sense funcionar.

### 3 Proves de connexió

En les següents dues captures podem veure que en cas que un dels dos servidors de la granja caigués, l'altre ocuparia el seu lloc i respondria a la petició del client. Per tant, tenim redundància. Ho podem veure en les captures 2 i 3 del document.

També podem veure que en les captures 4, 5 i 6, hem apagat el encaminador el qual tenia el estat Actiu i responia les peticions del client, forçant així que l'altre servidor ocupés el seu lloc i pogués respondre les peticions del client.

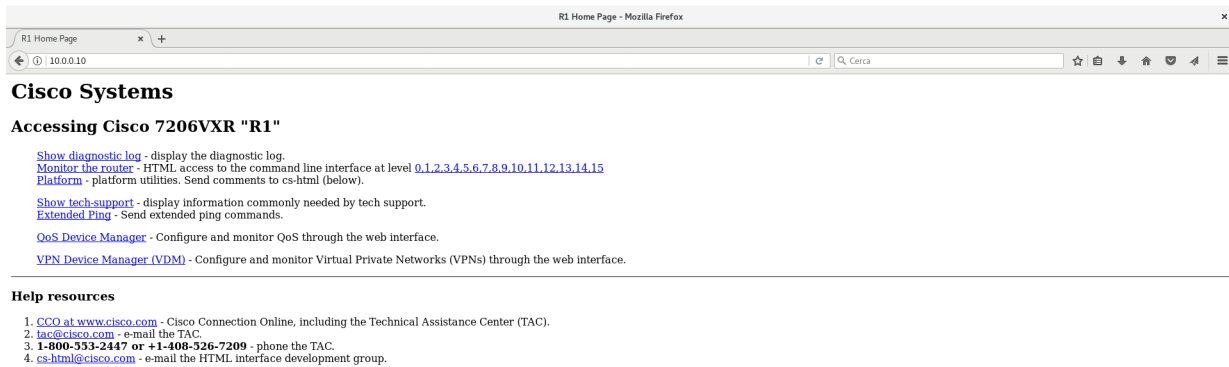


Figura 2: Connexions mitjançant el protocol HTTP - R1

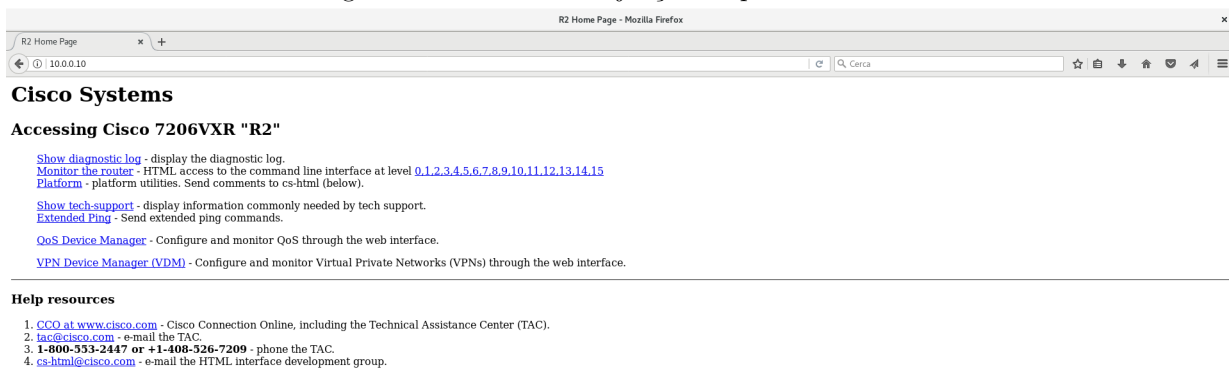
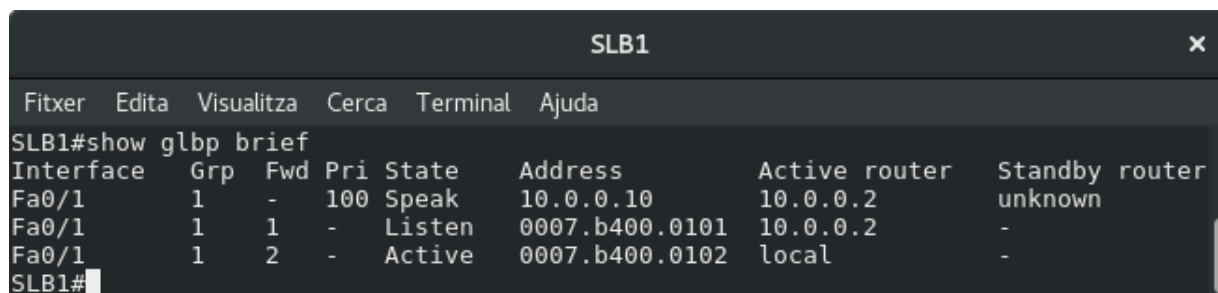


Figura 3: Connexions mitjançant el protocol HTTP - R2

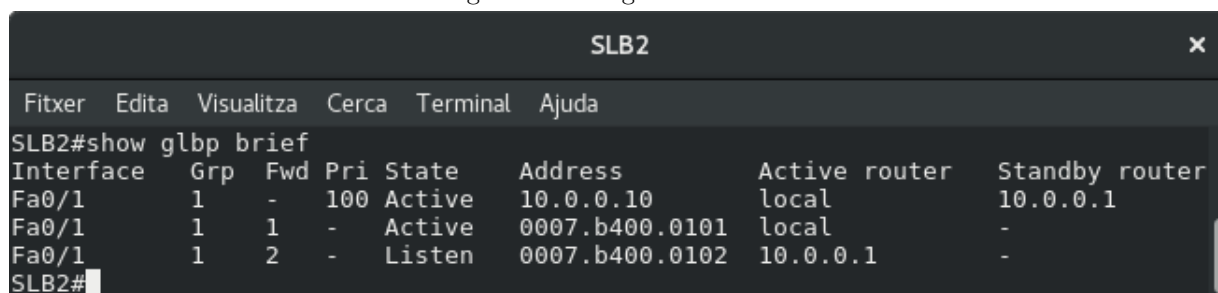


```

SLB1
Fitxer  Edita  Visualitza  Cerca  Terminal  Ajuda
SLB1#show glbp brief
Interface  Grp  Fwd  Pri  State  Address  Active router  Standby router
Fa0/1      1    -   100  Speak  10.0.0.10  10.0.0.2       unknown
Fa0/1      1    1   -   Listen 0007.b400.0101 10.0.0.2       -
Fa0/1      1    2   -   Active 0007.b400.0102 local          -
SLB1#

```

Figura 4: Configuració GLBP SLB1

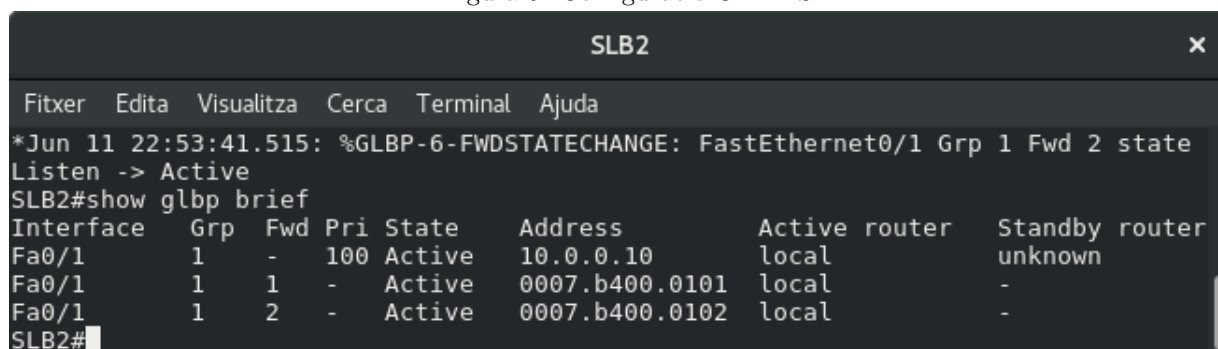


```

SLB2
Fitxer  Edita  Visualitza  Cerca  Terminal  Ajuda
SLB2#show glbp brief
Interface  Grp  Fwd  Pri  State  Address  Active router  Standby router
Fa0/1      1    -   100  Active  10.0.0.10  local          10.0.0.1
Fa0/1      1    1   -   Active 0007.b400.0101 local          -
Fa0/1      1    2   -   Listen 0007.b400.0102 10.0.0.1       -
SLB2#

```

Figura 5: Configuració GLBP SLB2



```

SLB2
Fitxer  Edita  Visualitza  Cerca  Terminal  Ajuda
*Jun 11 22:53:41.515: %GLBP-6-FWDSTATECHANGE: FastEthernet0/1 Grp 1 Fwd 2 state
Listen -> Active
SLB2#show glbp brief
Interface  Grp  Fwd  Pri  State  Address  Active router  Standby router
Fa0/1      1    -   100  Active  10.0.0.10  local          unknown
Fa0/1      1    1   -   Active 0007.b400.0101 local          -
Fa0/1      1    2   -   Active 0007.b400.0102 local          -
SLB2#

```

Figura 6: Configuració GLBP SLB2 després caiguda SLB1

## 4 SNMP

### 4.1 Introducció

A continuació es detallarà la solució implementada per a la gestió de routers mitjançant el protocol snmp. No s'han assolit tots els objectius proposats en l'enunciat de la pràctica. Hem configurat els dos primers objectius correctament i hem intentat fer el graf. Tot i així hem tingut problemes al posar les etiquetes als enllaços.

Per a la resolució d'aquest apartat hem tingut que utilitzar la llibreria de snmp-mibs per tal de poder interactuar amb els encaminadors CISCO.

### 4.2 Executar el programa

Les llibreries necessàries per l'execució del programa han sigut les següents:

- easysnmp
- graphviz
- os
- optparse

### 4.3 Estructura del programa

El programa el componen els següents documents en Python:

- part2.py: Lògica de l'executable
- utils.py: Classes per a la realització de l'execució.
- strings.py: Strings per a l'executable.

### 4.4 Procediments

Ara parlarem sobre les funcions més importants per a l'execució del programa:

- `analyze_topology()`: En aquesta classe recorrerem tota la topologia de la xarxa. Per a cada interfície que trobem, si no l'hem trobat abans, inicialitzarem un objecte Router on guardarem el nom, interfícies, veïns i rutes que té.
- `analyze_routers()`: Com hem dit en la funció anterior, aquí es crearan els diferents objectes de Router.
- `analyze_route()`: En aquesta funció s'analitzarà cada ruta que té cada encaminador. Es connectara amb el snmp i ens retornara la llista de ip's fent un walk.
- `analyze_route()`: En aquesta funció s'analitzarà cada interfície que té cada encaminador. Es connectara amb el snmp i ens retornara la llista de ip's, nom de cada interfície, velocitat.
- `graph_creator()`: En aquesta funció, es crearà un fitxer amb la topologia de la xarxa. Es creara un arxiu png i un arxiu .dot. Es important que hi hagi permisos d'escriptura en aquella carpeta si no no es podrà executar i guardar la topologia.