

Relazione esercitazione 5 Laboratorio di reti

Configurazione di uno switch CISCO

Franco Masotti Danny Lessio

May 3, 2015

Contents

I	Configurazione dello switch	2
1	Consegna	2
2	Collegamento allo switch	2
3	Configurazione di base	2
4	Salvataggio e ripristino della configurazione	3
5	VLAN	4
6	Porte trunk	4
7	Prove	5
II	Listati	6
8	startup-config	6
9	tftp	8

Part I

Configurazione dello switch

1 Consegna

L'obiettivo di questa esercitazione é quello di collegarsi ad uno switch CISCO e di configurarlo in modo da creare delle VLAN. Inoltre é richiesto di collegare insieme piú switch definendo alcune delle porte come trunk.

2 Collegamento allo switch

Per collegarsi inizialmente allo switch é necessario uno speciale adattatore dotato di attacco seriale da attaccare alla porta console presente nella parte posteriore dell'apparato di rete. Poiché i computer recenti non hanno un attacco seriale, é necessario di un secondo adattatore da seriale a USB.

Una volta collegati tutti i cavi bisogna individuare la periferica. Questo lo si può fare con il classico comando `dmesg`. Nel nostro caso la periferica é `ttyUSB0`.

A questo punto bisogna installare un programma per mettere in comunicazione il computer con lo switch. Noi abbiamo usato `minicom`¹, che é presente nei repository ufficiali². Iniziamo la configurazione con il comando `sudo minicom -s`³, dove `-s` significa *setup*. A questo punto impostiamo i valori come indicato dalla guida⁴. Se il collegamento va a buon fine ci troviamo il terminale con la scritta `Switch>`. A questo punto inizia la configurazione vera e propria.

3 Configurazione di base

Prima di poter configurare qualunque cosa é necessario entrare in modalità amministratore. Per farlo basta eseguire il comando `enable`. A questo punto il *prompt* cambia e viene visualizzato `Switch#`⁵.

Impostiamo la data e l'ora corrente con `clock set hh:mm:ss dd mmm6 yyyy`.

In questa modalità possiamo anche vedere tutta la configurazione corrente con `show running-config`.

Adesso si entra in una modalità ancora piú particolare con il comando `configure terminal` dove abbiamo configurato i parametri fondamentali. Il *prompt* cambia ancora, così: `Switch(config)#`. Qui si imposta il fuso orario con `clock timezone GMT +1`. A questo punto abbiamo cambiato l'hostname: `hostname hostname7`, e cambiamo la password di amministrazione con `enable password newpassword8`. Abilitiamo ora il servizio di cifratura delle password con `service password-encryption`. Infine controlliamo se la password risulta criptata con `show running-config`.

⁰networks-lab Copyright (C) 2016 frnmst (Franco Masotti), dannylessio (Danny Lessio). This document comes with ABSOLUTELY NO WARRANTY. This is free software, and you are welcome to redistribute it under certain conditions; see LICENSE file for details.

¹<https://alioth.debian.org/projects/minicom>

²<https://www.archlinux.org/packages/?name=minicom>

³É necessario essere utente `root` per aprire la periferica `ttyUSB0`.

⁴<https://help.ubuntu.com/community/CiscoConsole>

⁵Simile a quando si diventa utente `root` con `su` in un terminale.

⁶Il mese può andare col nome completo (es: `April`) oppure abbreviato (es: `Apr`)

⁷Nel nostro caso `hostname gruppo2`.

⁸Nel nostro caso `enable password labreti`. Le nostre password, da questo punto saranno tutte `labreti`.

Se si vuole accedere allo switch da una postazione remota bisogna abilitare l'amministrazione. Questa si trova di default nella VLAN 1. Rimaniamo nella modalità `configure terminal` e diamo in sequenza i seguenti comandi:

- `interface vlan1`⁹
- `ip address switch-address switch-subnet-mask`¹⁰
- `no shut`¹¹

Collegiamo i due apparecchi con un cavo ethernet e testiamo il tutto dal computer con un `ping switch-address`¹². Anche lo switch ha il comando `ping`.

Dopo aver configurato la rete si può abilitare l'amministrazione con telnet (quindi non c'è più bisogno del cavo seriale) per controllare lo switch da una postazione remota. Nonstrate telnet sia abilitato di default questo non funzionerà correttamente finché non impostiamo una password. È sufficiente eseguire questi tre comandi, dopo essere entrati nella modalità `configure terminal`:

- `line vty 0 4`¹³
- `login`
- `password telnet-password`¹⁴

4 Salvataggio e ripristino della configurazione

Dopo aver configurato la rete siamo pronti per un salvataggio preliminare delle impostazioni. Per salvare la configurazione dello switch abbiamo usato un server tftp chiamato `xinetd`¹⁵ presente nei repository ufficiali¹⁶. Qui è sufficiente cambiare una riga¹⁷ nel file di configurazione ed avviare il servizio¹⁸. Creiamo la cartella di destinazione del server¹⁹ e creiamo anche il file di destinazione²⁰ (che deve esistere prima dell'invio del file stesso).

Torniamo sullo switch in modalità `enable` e copiamo le impostazioni così: `copy running-config tftp:` avendo cura di specificare l'indirizzo del PC e il nome del file sorgente e destinazione²¹. Se non ci sono errori viene indicato il numero di byte copiati.

Per verificare il salvataggio è sufficiente fare `cat /var/tftpboot/running-config` e confrontarlo con l'output di `show running-config` presente nello switch.

È anche possibile ripristinare e rendere persistente le impostazioni sullo switch. Per farlo basta copiare `running-config` nel file `startup-config` all'interno dell'*nvr*am dello switch²² (che è la memoria non volatile. Il file `startup-config` infatti viene caricato all'avvio.

⁹Seleziona l'interfaccia.

¹⁰Nel nostro caso `ip address 192.168.2.100 255.255.255.0`. Ovviamente è necessario cambiare la sottorete dell'interfaccia cablata del computer in modo che sia la stessa di quella dello switch.

¹¹Abilita l'interfaccia di rete.

¹²Nella nostra configurazione con: `ping 192.168.2.100`

¹³Abilita il login fino a 5 sessioni consecutive.

¹⁴Setta la password per il telnet. Nel nostro caso è `labreti`

¹⁵`xinetd` = the extended Internet services daemon.

¹⁶<https://www.archlinux.org/packages/?name=xinetd>

¹⁷Questo verrà spiegato nella parte dei listati

¹⁸Con `systemd: sudo systemctl start xinetd`.

¹⁹`mkdir /var/tftpboot && chmod 666 /var/tftpboot` in modo che tutti gli utenti abbiano permessi di lettura e scrittura.

²⁰`touch /var/tftpboot/running-config && chmod 666 /var/tftpboot/running-config`

²¹Sempre `running-config`.

²²`copy tftp: nvram:/startup-config`

5 VLAN

La seconda parte dell'esercitazione consiste nella creazione di alcune VLAN (Virtual LAN) in modo da collegarsi anche agli switch degli altri gruppi. Usando le VLAN infatti é possibile creare molte reti virtualmente separate ma fisicamente collegate. In questo modo si risparmia il numero dei cablaggi e degli apparati di rete.

Per identificare univocamente le VLAN vengono usati numeri interi da 1 a *vlan-max*. Inoltre ad ogni VLAN può essere associato un nome sotto forma di stringa. Quindi, per prima cosa, abbiamo creato la VLAN 2 di nome `vlan2`, appunto. Entriamo nella modalità `enable` poi:

- `vlan database`²³
- `vlan 2 name vlan2`²⁴
- `exit`²⁵

Per controllare che la configurazione sia andata a buon fine usiamo il comando `show vlan`, il quale ci mostrerà lo stato di tutte le VLAN.

Lo switch in dotazione ha 12 porte, quindi possiamo assegnare almeno due porte per ognuna delle quattro VLAN. Ora noteremo che tutte le interfacce fisiche sono assegnate alla VLAN 1: dobbiamo assegnarne 3 alla VLAN 2. Entriamo prima nella modalità `configure terminal` poi:

- `interface Fa0/2`²⁶
- `switchport access vlan 2`²⁷
- `exit`

Ripetiamo queste operazioni anche per le porte 3 e 4. Ora rendiamo la VLAN 2 amministrativa nello stesso modo dell'VLAN 1. A questo punto l'interfaccia amministrativa della VLAN 1 si disattiva automaticamente.

Con lo stesso procedimento per creare la VLAN 2 (esclusa la parte di amministrazione) abbiamo creato anche le VLAN 11 e 3 (Con due porte assegnate ciascuna).

Per collegarci fisicamente agli switch degli altri gruppi abbiamo usato dei cavi di tipo crossover²⁸ poiché sono apparati dello stesso tipo, quindi é necessario invertire ricezione e trasmissione.

6 Porte trunk

Per poter mettere in comunicazione stesse VLAN che si trovano su switch diversi i pacchetti vengono incapsulati in modo che abbiano un'identificativo con il numero della VLAN. Tutto questo si ottiene assegnando una/piú porta/e come trunk.

Trovandoci fisicamente adiacenti sia al gruppo 1 sia al gruppo 3, abbiamo configurato le porte 1 e 12 come porte trunk. Per impostare una porta trunk entriamo nella modalità `configure terminal`, poi:

- `interface Fa0/1`

²³Entra nella modalità di configurazione delle VLAN.

²⁴Assegna il nome `vlan2` alla VLAN 2.

²⁵Per applicare la configurazione e uscire.

²⁶Entra nella configurazione dell'interfaccia 2

²⁷Assegna l'interfaccia 2 alla VLAN 2

²⁸https://en.wikipedia.org/wiki/Ethernet_crossover_cable

- `switchport mode trunk` ²⁹
- `switchport trunk allowed vlan add 2,3,11` ³⁰

Ripetiamo lo stesso procedimento per la porta 12 (cioé con Fa0/12).

7 Prove

Per testare le nostre cofigurazioni abbiamo settato gli indirizzi delle interfacce di rete dei computer di tutti i gruppi nella sottorete `192.168.1.0/24` e con il comando `ping` abbiamo verificato di poter raggiungere tutti i computer della sottorete.

²⁹ Abilita l'interfaccia in modalità trunk.

³⁰ Aggiunge le VLAN 2,3,11 alla porta trunk, dove la VLAN 11 é quella del gruppo 1 per evitare conflitti con la VLAN 1 stessa.

Part II

Listati

8 startup-config

startup-config

File di configurazione principale dello switch da mettere in nvram:/.

```
!  
! running-config  
!  
! Copyright (C) 2016 frnmst (Franco Masotti) <franco.  
    masotti@student.unife.it>  
!                               dannylessio (Danny Lessio)  
!  
! This file is part of networks-lab.  
!  
! networks-lab is free software: you can redistribute it and/or  
    modify  
! it under the terms of the GNU General Public License as  
    published by  
! the Free Software Foundation, either version 3 of the License,  
    or  
! (at your option) any later version.  
!  
! networks-lab is distributed in the hope that it will be useful,  
! but WITHOUT ANY WARRANTY; without even the implied warranty of  
! MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
! GNU General Public License for more details.  
!  
! You should have received a copy of the GNU General Public  
    License  
! along with networks-lab. If not, see <http://www.gnu.org/  
    licenses/>.  
!  
  
!  
version 12.1  
no service pad  
service timestamps debug uptime  
service timestamps log uptime  
service password-encryption  
!  
hostname gruppo2  
!  
enable password 7 1042081B17120602  
!
```

```

clock timezone GMT 1
ip subnet-zero
!
!
spanning-tree mode pvst
no spanning-tree optimize bpdu transmission
spanning-tree extend system-id
!
!
!
!
interface FastEthernet0/1
    switchport mode trunk
!
interface FastEthernet0/2
    switchport access vlan 2
    switchport mode access
!
interface FastEthernet0/3
    switchport access vlan 2
    switchport mode access
!
interface FastEthernet0/4
    switchport access vlan 2
    switchport mode access
!
interface FastEthernet0/5
    switchport access vlan 11
    switchport mode access
!
interface FastEthernet0/6
    switchport access vlan 11
    switchport mode access
!
interface FastEthernet0/7
    switchport access vlan 3
    switchport mode access
!
interface FastEthernet0/8
    switchport access vlan 3
    switchport mode access
!
interface FastEthernet0/9
!
interface FastEthernet0/10
!
interface FastEthernet0/11
!
interface FastEthernet0/12

```

```
    switchport mode trunk
!
interface Vlan1
  ip address 192.168.2.100 255.255.255.0
  no ip route-cache
  shutdown
!
interface Vlan2
  ip address 192.168.2.100 255.255.255.0
  no ip route-cache
!
ip http server
!
line con 0
line vty 0 4
  password 7 04570A041D245847
  login
line vty 5 15
  login
!
!
end
```


9 tftp

/etc/xinet.d/tftp

File di configurazione del server tftp. Abbiamo semplicemente cambiato l'ultima riga da `disable = yes` a `disable = no`, in modo da abilitare il servizio.

```
#
# tftp
#
# Copyright (C) 2016 frnmst (Franco Masotti) <franco.
#   masotti@student.unife.it>
#           dannylessio (Danny Lessio)
#
# This file is part of networks-lab.
#
# networks-lab is free software: you can redistribute it and/or
#   modify
#   it under the terms of the GNU General Public License as
#   published by
#   the Free Software Foundation, either version 3 of the License,
#   or
#   (at your option) any later version.
#
# networks-lab is distributed in the hope that it will be useful,
#   but WITHOUT ANY WARRANTY; without even the implied warranty of
#   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
#   GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public
#   License
#   along with networks-lab. If not, see <http://www.gnu.org/
#   licenses/>.
#

service tftp
{
    socket_type      = dgram
    protocol         = udp
    wait             = yes
    user             = nobody
    server            = /usr/sbin/tftpd
    server_args      = /var/tftpboot
    disable          = no
}
```