

ANAWS Project

DiffServ Tool

Dario Figliuzzi, Giuliano Peraz, Valerio Tanferna



Introduzione	2
Scopo del tool	2
Struttura della Rete	2
Protocolli utilizzati	3
Funzionalità del Tool	4
Classi Standard	5
Classi Custom	5
Implementazione e classi Java	6
Tool	6
Functions	6
OSPFRouter	6
Manuale d'uso	7
Conclusioni	12
Bibliografia	13

Introduzione

Il progetto si compone di un software denominato DiffServ Tool che permette ad un amministratore di rete di configurare DiffServ su uno o più router Cisco. Il tool comprende uno script di inizializzazione delle interfacce di rete, **runtool.sh**, ed un eseguibile scritto in Java che si occupa della connessione verso i router, della configurazione di questi e lascia la possibilità all'utente di definire i propri comandi per la classificazione dei pacchetti.

Il tool ha come obiettivo di automatizzare il più possibile il processo di configurazione dei router Cisco, ma richiede all'utente le informazioni necessarie al setup di DiffServ quali l'indirizzo dei router a cui connettersi, le interfacce di rete sulle quali applicare le service-policy e alcuni parametri nel caso in cui l'utente voglia definire una classe custom piuttosto che utilizzare una delle classi, dette "classi standard", che si possono trovare nella directory *classes/std* della root dell'eseguibile.

Scopo del tool

Il traffico generato dalla rete deve essere in qualche modo classificato per rispettare la Quality of Service per determinati tipi di servizio come le applicazioni multimediali quali, per esempio, VoIP e streaming Video. Si necessita quindi di un paradigma alternativo al best effort che non garantisce un uso ottimale della banda per tipologia di traffico, un limite superiore alla latenza e un uso più efficiente delle risorse di rete.

DiffServ alloca risorse non ai singoli flussi, ma per forwarding classes. La classificazione ed il marking dei pacchetti viene svolto dai Border Router. Il PHB (Per Hop Behaviour) dei Core Router si basa sul forwarding prioritario dei pacchetti, priorità definita dall'utente o dalle istruzioni presenti nei file delle classi standard che sono create dallo script di avvio. La priorità si basa sul campo DSCP dei pacchetti IP (il vecchio ToS per IPv4) che è impostato dai BR e controllato dai Core Router. In base ad esso il Core Router sa quali risorse riservare per quale valore del DSCP.

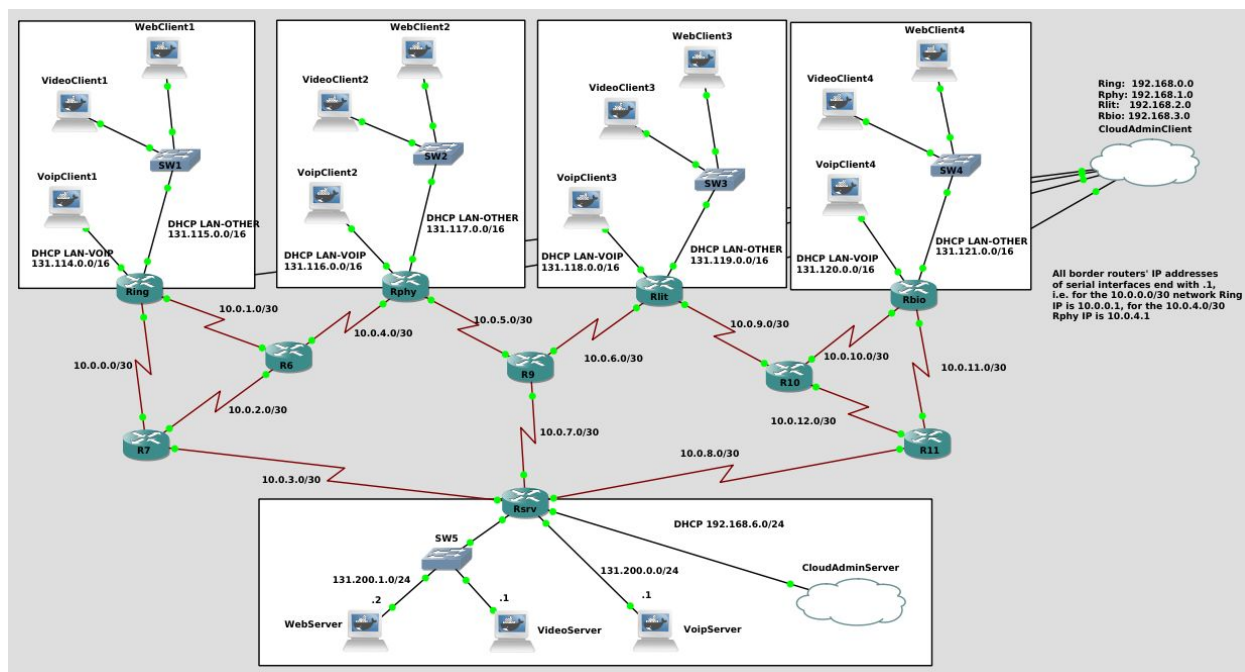
Struttura della Rete

Il tool è stato testato su una rete simulata costruita tramite il popolare software di emulazione di rete GNS3 (versione 2.1.3). Abbiamo pensato di creare una rete simile a quella universitaria, ma semplificata per i nostri scopi, in cui sono presenti 4 dipartimenti con dei client che si connettono a dei server remoti attraverso una rete di router Cisco. Ogni router è configurato in modo da avere un processo OSPF per il routing del traffico.

Gli indirizzi IP dei router sono configurati manualmente così come quelli dei server. I client si connettono tramite richieste DHCP al router di riferimento del proprio dipartimento (un dipartimento è un rettangolo della rete a cui corrisponde un router denominato R#### dove #### è la sigla del dipartimento).

La rete è stata suddivisa in 3 aree:

1. **Area 0:** identifica la backbone della rete universitaria che si occupa di smistare il traffico dai router di dipartimento al router dei server. I router che appartengono a questa area sono definiti Core Router;
2. **Area 1:** identifica l'area a cui sono connessi i client. Il Border Router dei client è configurato in modo da avere un DHCP server in esecuzione;
3. **Area 2:** identifica l'area a cui sono connessi i server dell'università; gli indirizzi dei server sono configurati manualmente poiché non c'è un server DNS configurato.



Il tool si connette alla rete attraverso l'interfaccia di rete **tap0**. All'avvio del tool richiederà un indirizzo al router connesso via DHCP. Il tool è pensato per essere connesso a un router con DHCP abilitato e configurato.

Ogni router utilizza il protocollo OSPF per fare routing del traffico di rete, anche la nuvola che rappresenta l'host che esegue il tool è configurato come router OSPF. La configurazione dell'host avviene tramite l'uso delle utility di frrouting.org come richiesto da specifiche.

Protocolli utilizzati

Il tool usa diversi protocolli:

- **SSH:** la connessione ai router avviene attraverso una libreria Java chiamata **SSHJ**. Su ogni router è stata impostata una chiave univoca per permettere la connessione;
- **OSPF:** è il protocollo di routing della rete basato su un algoritmo di tipo link state per determinare se un collegamento tra due router si è interrotto o meno;

- **IPv4:** garantisce la connettività a livello 3. Abbiamo usato la “vecchia” versione di IP perché le utility di frrouting.org garantiscono il funzionamento corretto solo con questa versione di IP;
- **DiffServ:** non è propriamente un protocollo di rete ma un modello scalabile multiservizio, cioè un insieme di tecniche e tecnologie pensate per classificare e gestire il traffico di rete. Permette di implementare il Quality of Service (QoS) e di utilizzare al meglio le risorse di rete disponibili.

Funzionalità del Tool

Il tool lavora sotto alcune supposizioni:

- *I router sono configurati per permettere l'accesso tramite SSH*
- *I router sono accessibili via SSH con username = “cisco” e password = “cisco”*
- *L'host su cui si esegue il tool è connesso a un router su cui è in esecuzione un server DHCP*
- *Sull'host sono installati i demoni di Frrouting e la JVM versione 8 o superiore*
- *Sull'host sono installati Bash e sudo*

Le funzionalità del tool sono le seguenti:

- *Connect to router:* consente la connessione a un router presente nella rete. Inizialmente viene creata una lista con i router che sono presenti nella rete, usando le informazioni sulla tipologia fornite dal database di OSPF. Successivamente si richiede all'utente a quale router connettersi e il tool si occupa della connessione verso il router presentando il prompt standard dei router cisco.
- *Show topology:* richiede al demone OSPFD di frrouting.org di fornire la topologia della rete che comprende le informazioni contenute nel database OSPF.
- *Configure DiffServ:* lascia all'utente la possibilità di configurare DiffServ utilizzando comandi di classificazione dei pacchetti presenti in dei file, chiamati “classi standard” presenti nella directory *classes/std*, oppure usando dei file che sono stati definiti dall'utente attraverso la funzionalità *Define New Class*, questi file sono detti *classi custom*. Il tool richiede se configurare le classi su tutti i router o su uno solamente, in questo caso mostrerà una lista di router da scegliere, chiederà le interfacce al router scelto e applicherà di conseguenza i comandi di configurazione.
- *Define New Class:* chiede all'utente di definire una nuova classificazione da salvare in un file e permette all'utente di essere guidato attraverso un wizard che richiede delle informazioni sui parametri della classificazione: quale valore DSCP inserire nei pacchetti che corrispondono alla classificazione, il limite di banda, la lunghezza della coda dei pacchetti, se abilitare o meno Random Early Detection (RED) e altro. Alla fine il tool

richiede all'utente se vuole modificare i comandi inseriti e salva i comandi su un file che ha lo stesso nome della classe dato dall'utente..

- *Show Router Configurations*: la funzionalità è pensata per connettersi ai router e mostrare il contenuto della loro *running-config*, cioè la configurazione con la quale sta lavorando il router salvata in memoria.

Classi Standard

Le classi standard sono file che contengono le istruzioni per configurare DiffServ sui Core Router.

Abbiamo definito 4 classi standard per automatizzare il processo di configurazione di DiffServ sui Border Router (BR) e sui Core Router (CR). In base a queste classi il tool automaticamente imposta le access-list, le policy-map e le class-map discriminando tra BR e CR.

Le classi standard possono essere modificate dall'utente tramite editing diretto, ma questo comportamento non è consigliato per l'uso di questo tool. Abbiamo previsto la funzionalità *define new class* nel caso l'utente voglia utilizzare delle classi definite da sé.

Le 4 classi standard usate sono:

1. **VoIP**: per traffico di telefonia o traffico che richiede bassa latenza e ha la garanzia di avere la più alta probabilità di essere spedito prima di altri pacchetti in coda;
2. **Video**: per il traffico che richiede una certa garanzia di banda ma che può sopportare la perdita di pacchetti;
3. **Web**: garantisce delle risorse a coloro che navigano Internet, evitando che altri tipi di traffico consumino tutte le risorse;
4. **Excess**: garantisce una minima percentuale di banda a tutto il traffico che non ricade nelle classi precedenti.

Classi Custom

Il tool prevede la possibilità che l'utente definisca una lista di comandi che comporranno la cosiddetta *classe custom*. Ogni classe custom è salvata nella directory *classes/new* in un file chiamato secondo il nome digitato dall'utente nel tool.

Le classi custom non necessitano di definire una access-list, una class-map o una policy-map che sono configurate automaticamente dal tool. All'utente si richiede solamente di specificare il tipo di protocollo (TCP o UDP) a cui applicare la access-list.

I comandi scritti dall'utente non hanno un meccanismo automatico di verifica, quindi è responsabilità dell'utente scrivere i comandi in modo che il router sia configurato correttamente.

Implementazione e classi Java

DiffServ Tool si appoggia su due librerie esterne:

1. **SSHJ**, scritta da Jeroen van Erp, è una libreria che gestisce le connessioni SSH, dalla connessione alla disconnessione basandosi sulla configurazione dell'host.
2. **ExpectIT**, scritta da Alexey Gavrilov, è una libreria usata per inviare i comandi verso i router e riceverne l'output. Mima il funzionamento di Expect, un tool pensato per creare script di comandi che devono essere eseguiti in remoto.

Per il funzionamento corretto di SSHJ sono necessarie altre librerie che si trovano nella directory *lib* del tool; una di queste è chiamata **BouncyCastle** che richiede l'abilitazione delle Unlimited Strength Java Cryptographic Extensions.

Per maggiori informazioni sulle classi si può fare riferimento direttamente al codice sorgente di queste.

DiffServ Tool è composto da 3 classi principali, **Tool**, **Functions** e **OSPFRouter**:

Tool

È il main del tool, si occupa della presentazione del menu di scelta delle funzionalità e richiama le funzionalità della classe Functions.

Functions

Implementa le funzionalità di DiffServ Tool e contiene al suo interno degli attributi per salvare le informazioni della rete, quali i router che ci sono, se sono o meno dei Border Router, il loro indirizzo IP e altre informazioni, in modo da evitare continue connessioni verso i dispositivi di rete.

OSPFRouter

Modella un router su cui è configurato un processo OSPF, si occupa della connessione e della disconnessione, di eseguire i comandi sui router e gestisce le informazioni che i metodi di Functions richiede, quali l'hostname, il nome delle interfacce e gli indirizzi IP di queste, la running configuration e altre.

Manuale d'uso

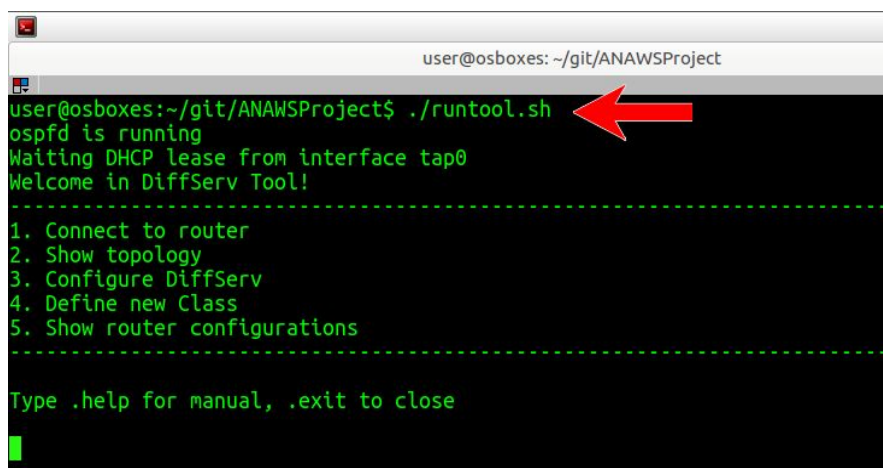
Avvio del tool

Per avviare DiffServ Tool è necessario spostarsi con il terminale nella cartella principale del tool e digitare **./runtool.sh** ed inserire la password di root.

All'avvio verrà visualizzato il menù principale.

Per chiudere il tool digitare **.exit**.

Per visualizzare il manuale con la descrizione delle funzionalità digitare **.help**; premere enter per ritornare al menù principale.



```
user@osboxes:~/git/ANAWSProject$ ./runtool.sh
ospfd is running
Waiting DHCP lease from interface tap0
Welcome in DiffServ Tool!
-----
1. Connect to router
2. Show topology
3. Configure DiffServ
4. Define new Class
5. Show router configurations
-----
Type .help for manual, .exit to close
```

Collegarsi ad un router

Per collegarsi ad un router digitare il numero **1**.

Il tool visualizzerà una lista dei router della rete alla quale è possibile connettersi.



```
--- List of available routers ---
Hostname      IP address    ABR?
R7             10.0.3.2
R6             10.0.4.2
R9             10.0.7.2
R10            10.0.12.1
R11            10.0.12.2
Ring           192.168.0.254 ABR
Rlit           192.168.2.254 ABR
Rbio           192.168.3.254 ABR
Rsrv           192.168.6.254 ABR
Choose the router IP of the list to which you want to connect to (.exit to return back):
```

Digitare l'indirizzo IP del router desiderato. Digitare **enable** ed inserire **cisco** come password.

Ora si è connessi al router ed è possibile inserire tutti i comandi riconosciuti da Cisco IOS.

Per disabilitare l'inserimento dei comandi digitare **disable**.

Per disconnettersi dal router digitare **.exit** o **.quit**. Premere enter per tornare al menù principale.


```

--- List of available routers ---
Hostname      IP address      ABR?
R7             10.0.3.2
R6             10.0.4.2
R9             10.0.7.2
R10            10.0.12.1
R11            10.0.12.2
Ring           192.168.0.254   ABR
Rlit           192.168.2.254   ABR
Rbio           192.168.3.254   ABR
Rsrv           192.168.6.254   ABR

Choose the router IP of the list to which you want to connect to (.exit to return back):
10.0.3.2
R7>terminal length 0
terminal length 0
R7>enable
enable
enable
Password: cisco
cisco
R7#

```

Visualizzare topologia della rete

Per visualizzare la topologia della rete digitare il numero 2.

Verrà visualizzato una lista dei router della rete (sia Core Router che Border Router), il proprio indirizzo locale ed il link con il quale si è collegati ed i link dei client e dei server assegnati dal router.

```

Show Topology

      OSPF Router with ID (192.168.0.1)

      Router Link States (Area 0.0.0.0)

Link ID        ADV Router    Age Seq#          CkSum Link count
10.0.3.2       10.0.3.2      780 0x80000002 0xcb52 6
10.0.4.2       10.0.4.2      782 0x80000001 0x3aad 5
10.0.7.2       10.0.7.2      781 0x80000002 0xb5b3 5
10.0.12.1      10.0.12.1     782 0x80000002 0xb71c 6
10.0.12.2      10.0.12.2     784 0x80000001 0x626c 6
192.168.0.1    192.168.0.1   139 0x80000003 0x6b34 1
192.168.0.254  192.168.0.254 138 0x80000003 0xf60c 5
192.168.2.254  192.168.2.254 782 0x80000002 0x75c7 5
192.168.3.254  192.168.3.254 783 0x80000002 0x54d4 5
192.168.6.254  192.168.6.254 783 0x80000001 0x4724 7

      Net Link States (Area 0.0.0.0)

Link ID        ADV Router    Age Seq#          CkSum
192.168.0.254  192.168.0.254 139 0x80000001 0xb0e5

      Summary Link States (Area 0.0.0.0)

Link ID        ADV Router    Age Seq#          CkSum Route
131.114.0.0    192.168.0.254 778 0x80000001 0xdcf7 131.114.0.0/16
131.115.0.0    192.168.0.254 778 0x80000001 0xd003 131.115.0.0/16
131.118.0.0    192.168.2.254 783 0x80000001 0x9e30 131.118.0.0/16
131.119.0.0    192.168.2.254 783 0x80000001 0x923b 131.119.0.0/16
131.120.0.0    192.168.3.254 784 0x80000001 0x7f4c 131.120.0.0/16
131.121.0.0    192.168.3.254 784 0x80000001 0x7357 131.121.0.0/16
131.200.0.0    192.168.6.254 779 0x80000001 0xa6d1 131.200.0.0/24
131.200.1.0    192.168.6.254 779 0x80000001 0x9bdb 131.200.1.0/24

```

Configurare DiffServ

Per configurare DiffServ su un router della rete digitare il numero **3**.

Verrà visualizzato un elenco delle classi disponibili: quelle standard (Video, VoIP, Web, Excess) e quelle definite dall'utente.

Verrà chiesto quale di queste classi si vuole usare.

Digitare **std** per settare una classe standard, **new** per una di quelle definite dall'utente;

Digitare **all** o **one** per configurare rispettivamente tutti router oppure solamente uno.

Nel caso si digiti **one**, viene visualizzato l'elenco dei router (hostname, IP e se sono border router oppure no) e chiesto di inserire l'indirizzo IP ed il numero/i della/e classe/i da settare.

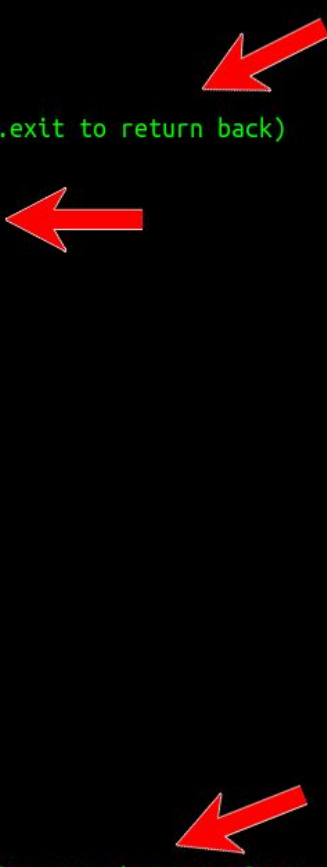
```
Starting DiffServ Configuration wizard...

STANDARD CLASSES:
-VOIP
-VIDEO
-WEB
-EXCESS
ADMIN DEFINED CLASSES:
-myclass1
-myclass2
-myclass5
-myclass3
-myclass6
-myclass4
Do you want to use standard (std) or new classes(new)? (.exit to return back)
std
Type:
<all> to configure DiffServ on every routers in the net
<one> to configure DiffServ on a single router

one
Getting router list

--- List of available routers ---
Hostname      IP address      ABR?
R7             10.0.3.2
R6             10.0.4.2
R9             10.0.7.2
R10            10.0.12.1
R11            10.0.12.2
Ring           192.168.0.254   ABR
Rlit           192.168.2.254   ABR
Rbio           192.168.3.254   ABR
Rsrv           192.168.6.254   ABR

Choose the router (enter the IP address): 10.0.3.2
Choose which class or classes to apply:
1- VoIP
2- Video
3- Web
4- Excess
You can insert a comma separated list if you want to apply more than one class
1,2,3
```



Inserire le interfacce sulla quale settare tali classi.

```
The following is the list of interfaces of 192.168.0.254 ( Ring )
# Interface      Address
1 Ethernet0/0     192.168.0.254
2 Ethernet0/1     131.114.100.254
3 Ethernet0/2     131.115.100.254
4 Serial1/0       10.0.1.1
5 Serial1/1       10.0.0.1
Choose the interface (or interfaces) to apply the diffserv classes (1-5): 1
```

Verrá chiesto se applicare la service policy al traffico di ingresso, rispondere **yes** o **no**.

```
Apply the service policy to the input traffic? (yes/no)
yes
```

Nel caso nella seconda scelta si digiti **all** verrà richiesto, ripetutamente per ogni singolo router, la classe da applicare, le interfacce e se applicare la service policy al traffico di ingresso.

Definire una nuova classe

Per definire una nuova classe digitare il numero **4**.

Verrá richiesto il nome e se si desidera un wizard per aiutare la configurazione oppure se digitare tutto i manualmente.

```
List of available classes:
1: myclass1
2: myclass2
3: myclass5
4: myclass3
5: myclass4

Enter name of the new class or those you want redefine (.exit to return back):
myclass6
Do you prefer to be guided through the class definition?
yes - the wizard will allow you to define basic features such as shaping,
      policing and enabling RED
no - you can write your own class line by line

Response (yes/no):
```

Digitanto **yes** si utilizzerá il wizard.

Ora sarà necessario rispondere ad alcune domande: settare il limite di banda in punti percentuali da **1** a **100** (**0** no limits), traffic shaping della velocità media in bps (**0** no shaping), se applicare il Random Early Detection (algoritmo di drop dei pacchetti per prevenire l'insorgere di una congestione) ed il DSCP (in valore numerico o alfanumerico).

Verrà visualizzato il contenuto della classe appena definita e sarà possibile effettuare delle modifiche manuali, riga per riga. Premendo **enter** oppure digitando **no** la classe verrà salvata e si tornerà al menu principale.

Digitando **no** sarà necessario digitare manualmente tutti i comandi Cisco IOS.

Digitare **.exit** per terminare. Verrà visualizzato il contenuto della classe appena definita e sarà possibile effettuare delle modifiche manuali, riga per riga. Premendo **enter** oppure digitando **no** la classe verrà salvata e si tornerà al menu principale.

```
Write bandwidth limit (0 = no limit, max 100): 100
Write traffic shaping on average speed of (bps, 0 = no shaping): 0
Apply random early detection to class? (yes/no) yes
Available DSCP values:
DSCP      Decimal
CS1        8
EF         46
CS3        24
BE         0
CS2        16
CS5        40
CS4        32
CS7        56
CS6        48
AF12       12
AF23       22
AF13       14
AF21       18
AF32       28
AF43       38
AF11       10
AF22       20
AF33       30
AF41       34
AF31       26
AF42       36
Set a DSCP value for this class (literal or numerical, 0 is default): EF
The content of the class myclass6 is the following:
1 class myclass6
2 bandwidth percent 100
3 random-detect
4 no random-detect precedence-based
5 set ip dscp EF
Insert the line or the lines (comma separated list) you want them to be changed (type no or RETURN if it is all right):
2
Old row value:
bandwidth percent 100
Insert new line below :
bandwidth percent 50
```

Visualizzare la configurazione di un router

Per visualizzare la configurazione di un router, digitare il numero **5**.

Verrà visualizzato un elenco dei router, inserire l'indirizzo IP del router del quale si vuole visualizzare la configurazione.

```
--- List of available routers ---
Hostname      IP address    ABR?
R7             10.0.3.2
R6             10.0.4.2
R9             10.0.7.2
R10            10.0.12.1
R11            10.0.12.2
Ring           192.168.0.254  ABR
Rlit           192.168.2.254  ABR
Rbio           192.168.3.254  ABR
Rsrv           192.168.6.254  ABR

Choose the router IP of the list to which you want the running-conf:
192.168.0.254
```

Conclusioni

Il tool è una buona dimostrazione di come si possa automatizzare il processo di configurazione dei router Cisco all'interno di una rete. Non essendo un prodotto commerciale presenta dei difetti strutturali di base:

- Credenziali di accesso: al momento sono hardcodate nella classe OSPFRouter, un software migliore richiederebbe all'utente di fornire le credenziali e non le salverebbe in memoria. Meglio sarebbe l'implementazione di un meccanismo a chiave pubblica per l'accesso in SSH tramite quelle piuttosto che usare le credenziali di login.
- Verifica dei comandi di configurazione: al momento il software non prevede l'uso di un meccanismo di verifica dei comandi di configurazione, quindi non sappiamo se un comando è valido prima di inviarlo al router. Manca una verifica dell'output del router nel caso di comando non valido. I test effettuati con il tool hanno dato la certezza che i comandi scritti da noi sono validi.
- Interfaccia utente: così com'è stata scritta ed è visibile, l'interfaccia utente non è un buon esempio di usabilità, ci sono ancora molte domande a cui l'utente deve rispondere, si potrebbe migliorare inserendo una GUI che raccolga tutte le informazioni in una unica schermata, invece che sparse in mezzo all'output del tool.
- Ottimizzazione delle connessioni: durante l'esecuzione del tool può capitare che l'utente debba attendere del tempo dovuto alla connessione verso il router, al raccoglimento delle informazioni e al successivo processamento di queste. Si potrebbe prevedere un meccanismo che riduca questi tempi cercando di salvare quante più informazioni possibile in memoria. Parte di questo è stato effettivamente implementato tramite l'uso degli attributi della classe Functions.
- Controllo della sovrascrittura della configurazione del router: ad ora le classi standard e custom vanno a sostituire eventuali classi, access-lists e policy-map già presenti sul router che hanno lo stesso nome o lo stesso numero di quelle standard o custom. Un miglioramento potrebbe essere quello di verificare che non si vada a sovrascrivere la configurazione già presente sul router.

Bibliografia

- **SSHJ**, Jeroen van Erp, <https://github.com/hierynomus/sshj>
- **ExpectIT**, Alexey Gavrilov, <https://github.com/Alexey1Gavrilov/ExpectIT>
- **Cisco IOS Configuration Fundamentals Command Reference**, Cisco Systems, Inc., https://www.cisco.com/c/en/us/td/docs/ios/fundamentals/command/reference/cf_book.html
- **OSPF**, FRRouting Project, <https://frrouting.org/user-guide/ospfd.html>