

Appunti primo approccio a JunOS CLI

by Andrea Barontini, Marzo 2018

- Accesso al dispositivo tramite prompt
 - livelli della CLI e passaggio da uno all'altro
- Visualizzazione configurazione
 - nuove e vecchie modalità di visualizzazione
 - semplici comandi della Pipe
- Navigazione nella configurazione gerarchica
 - come muoversi all'interno della configurazione
 - analogie con i file-system
- Gestione delle configurazioni
 - semplici editing e help in linea
 - salvataggi e recupero versioni precedenti
 - caricamento tramite tftp

Accesso al dispositivo tramite prompt

Junos ha 3 livelli di prompt:

- Shell (*prompt %*): accesso al sistema operativo;
- Operational Mode (*prompt >*): CLI con possibilità di configurazione limitate;
- Configuration Mode (*prompt #*): modalità analoga a *conf t* / *system-view*.

Accedendo come root si viene proiettati in modalità Shell, con gli altri utenti invece in Operational.

Comandi per passare da una modalità all'altra:

DA ↓ A →	Shell %	Operational >	Configuration #
Shell %	-	<i>cli</i>	-
Operational >	<i>exit</i> (solo root) <i>start shell</i>	-	<i>configure</i> <i>edit</i>
Configuration #	<i>run start shell</i>	<i>exit</i>	-

In modalità Operational e Configuration sono attivi il consueto completamento comandi tramite <TAB>, e la visualizzazione dell'elenco dei comandi utilizzabili al livello corrente tramite il tasto ?.

In modalità Configuration è possibile eseguire comandi della modalità Operational premettendo la keyword *run* (analogo al *do* di IOS). Es.:

```
root@provaJunos# run show configuration
```

Visualizzazione configurazione

Il comando base è *show configuration*. Applicato sul dispositivo Juniper vMX14.1R4.8-1 del test ritorna:

```
root@provaJunos> show configuration
## Last commit: 2018-03-01 15:56:17 UTC by es_reti
version 14.1R4.8;
system {
    host-name provaJunos;
    root-authentication {
        encrypted-password "$1$wjS3tl5M$7DMNEuaB1e/lTml7K/vz2."; ## SECRET-
DATA
    }
    name-server {
        192.0.2.10;
        192.0.2.20;
    }
    login {
        user baro {
            full-name "Andrea Barontini";
            uid 2000;
        }
    }
}
```

```

        class super-user;
        authentication {
            encrypted-password "$1$ZQaCQHsi$bvl0zCiPPulaM0KaJ/tXV1"; ##
SECRET-DATA
        }
    }
    user es_reti {
        uid 2001;
        class super-user;
        authentication {
            encrypted-password "$1$rAj.zeOD$yiXoQDMig004yykhEZDAX1"; ##
SECRET-DATA
        }
    }
}
services {
    ssh;
    telnet;
}
syslog {
    user * {
        any emergency;
    }
    file messages {
        any notice;
        authorization info;
    }
    file interactive-commands {
        interactive-commands any;
    }
}
}
interfaces {
    ge-0/0/9 {
        unit 0 {
            family inet {
                address 192.0.2.135/24;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 192.0.2.254;
    }
}
}

```

Tale visualizzazione evidenzia la logica gerarchica della configurazione costituita da sezioni nidificate racchiuse tra parentesi graffe, ma è possibile ottenere la configurazione flat e compatta (che per altro rappresenta esplicitamente i comandi da dare in modalità Configuration) tramite l'utilizzo della pipe / (che in Junos è estremamente potente in termini di filtri/modificatori impostabili):

```

root@provaJunos> show configuration | display set
set version 14.1R4.8
set system host-name provaJunos
set system root-authentication encrypted-password
"$1$wjS3tl5M$7DMNEuaB1e/lTml7K/vz2."
set system name-server 192.0.2.10

```

```

set system name-server 192.0.2.20
set system login user baro full-name "Andrea Barontini"
set system login user baro uid 2000
set system login user baro class super-user
set system login user baro authentication encrypted-password
"$1$ZQaCQHsi$bvl0zCiPPulaM0KaJ/tXV1"
set system login user es_reti uid 2001
set system login user es_reti class super-user
set system login user es_reti authentication encrypted-password
"$1$rAj.zeOD$yiXoQDMig004yykhEZDAX1"
set system services ssh
set system services telnet
set system syslog user * any emergency
set system syslog file messages any notice
set system syslog file messages authorization info
set system syslog file interactive-commands interactive-commands any
set interfaces ge-0/0/9 unit 0 family inet address 192.0.2.135/24
set routing-options static route 0.0.0.0/0 next-hop 192.0.2.254

```

Altri modificatori di `/ display` possono essere ovviamente elencati, come premesso sopra, tramite ?:

```

root@provaJunos> show configuration | display ?
Possible completions:
  changed          Tag changes with junos:changed attribute (XML only)
  commit-scripts   Show data after commit scripts have been applied
  detail           Show configuration data detail
  inheritance       Show inherited configuration data and source group
  max-depth        Maximum depth of configuration data
  omit             Emit configuration statements with the 'omit' option
  rfc5952          Display IPv6 addresses as per RFC 5952 specifications
  set              Show 'set' commands that create configuration
  xml              Show output as XML tags

```

segnalo:

show configuration | display detail

che visualizza la configurazione in modalità gerarchica arricchendola però di commenti che spiegano i comandi delle varie righe a mo' di help in linea.

Tra i molti altri filtri invece:

show configuration | find <testo_da_cui_partire> analogo al *| begin* di IOS
show configuration | match <testo_da_cercare> analogo a *| include*

Navigazione nella configurazione gerarchica

Utilizzando la configurazione dell'interfaccia è possibile fare pratica nelle operazioni di spostamento all'interno della configurazione.

Dalle indentazioni della configurazione gerarchica visualizzata in precedenza ne deriva che al primo livello ci sono 3 sezioni:

- *system*
- *interfaces*
- *routing-options*

Concentrandosi sulle interfacce e filtrando l'output di *show configuration* (modalità **Operational**):

```
root@provaJunos> show configuration | find interfaces
interfaces {
  ge-0/0/9 {
    unit 0 {
      family inet {
        address 192.0.2.135/24;
      }
    }
  }
}
routing-options {
  static {
    route 0.0.0.0/0 next-hop 192.0.2.254;
  }
}
```

Un output analogo ma circoscritto a uno specifico livello gerarchico può essere ottenuto con il comando *show <sezione>* in modalità **Configuration**, anche scendendo nel nesting:

```
root@provaJunos> configure
Entering configuration mode

[edit]
root@provaJunos# show interfaces
ge-0/0/9 {
  unit 0 {
    family inet {
      address 192.0.2.135/24;
    }
  }
}

[edit]
root@provaJunos# show interfaces ge-0/0/9
unit 0 {
  family inet {
    address 192.0.2.135/24;
  }
}

[edit]
root@provaJunos# show interfaces ge-0/0/9 unit 0
family inet {
  address 192.0.2.135/24;
}

[edit]
root@provaJunos#
```

Si noti che il prompt in modalità Configuration è sempre preceduto da una riga:

[edit]

Tale riga indica la sezione in cui ci si trova dentro, o semplicemente *[edit]* come fin'ora se si è al livello più alto, quello dell'intera configurazione nel suo complesso.

Per scendere a un livello sottostante si usa il comando *edit <sottosezione>* (per inciso si noti che dal contesto descritto nelle ultime righe appare ragionevole che per passare dalla modalità Operational a quella Configuration si possa usare il comando *edit* oltre che *configure*, come elencato nella tabella a inizio documento). Il comando *show* senza alcun parametro mostra la sezione corrente.

Gli esempi sotto mostrano quanto appena detto (si noti che si può scendere in un colpo solo anche al di sotto delle sezioni figlio):

```
root@provaJunos> edit
Entering configuration mode

[edit]
root@provaJunos# edit interfaces

[edit interfaces]
root@provaJunos# show
ge-0/0/9 {
    unit 0 {
        family inet {
            address 192.0.2.135/24;
        }
    }
}

[edit interfaces]
root@provaJunos# edit ge-0/0/9 unit 0

[edit interfaces ge-0/0/9 unit 0]
root@provaJunos# show
family inet {
    address 192.0.2.135/24;
}

[edit interfaces ge-0/0/9 unit 0]
root@provaJunos#
```

In pratica, volendo fare un paragone un po' forzato con un file system:

- *edit <sezione>* ↔ *cd <cartella>*
- *show* ↔ *ls o dir*

Per tornare al livello immediatamente superiore si usa il comando: *up*

Per risalire in un colpo solo al livello più alto (intera configurazione nel suo complesso): *top* o *exit*

Ovviamente molto utile in tutti i comandi visti il completamento tramite <TAB>, ancora una volta in maniera analoga alla modalità a cui si è abituati dai file system.

Gestione delle configurazioni

Ovviamente non ha senso entrare qua nelle peculiarità delle varie istruzioni di configurazione, ma si possono comunque dare indicazioni di utilità generale.

In modalità Configuration si interviene sulle varie sezioni tramite il comando *set* (che infatti compariva nella vista compatta della configurazione a inizio documento).

Per quanto riguarda la configurazione dell'interfaccia di rete, per esempio:

```
[edit]
root@provaJunos# set interfaces ge-0/0/9 unit 0 family inet address
192.0.2.135/24
```

Si noti che dopo *set* compaiono le sezioni nidificate, a scendere fino all'istruzione ultima (nel nostro caso *address 192.0.2.135/24*). Va per logica che se ci fossimo trovati non nella radice della configurazione ma già in una sottosezione l'istruzione riporterebbe il "percorso nelle sezioni" RELATIVO alla sezione corrente:

```
[edit]
root@provaJunos# edit interfaces ge-0/0/9

[edit interfaces ge-0/0/9]
root@provaJunos# set unit 0 family inet address 192.0.2.135/24
```

Per cancellare un'istruzione si usa il comando *delete* (con la stessa logica appena descritta per il comando *set*), OCCHIO però che senza alcuna opzione cancella tutta la sezione corrente (chiedendone comunque conferma prima).

Oltre al già citato onnipresente comando *?*, ulteriori informazioni possono essere ricavate con il comando *help* che costituisce un vero e proprio manuale in linea:

```
[edit]
root@provaJunos# help ?
Possible completions:
<[Enter]>      Execute this command
apropos        Find help information about a topic
reference      Reference material
syslog         System log error messages
tip            Tip for the day
topic          Help for high level topics
|             Pipe through a command
```

Per rendersi conto provare, per esempio, il comando:

help topic routing-options static

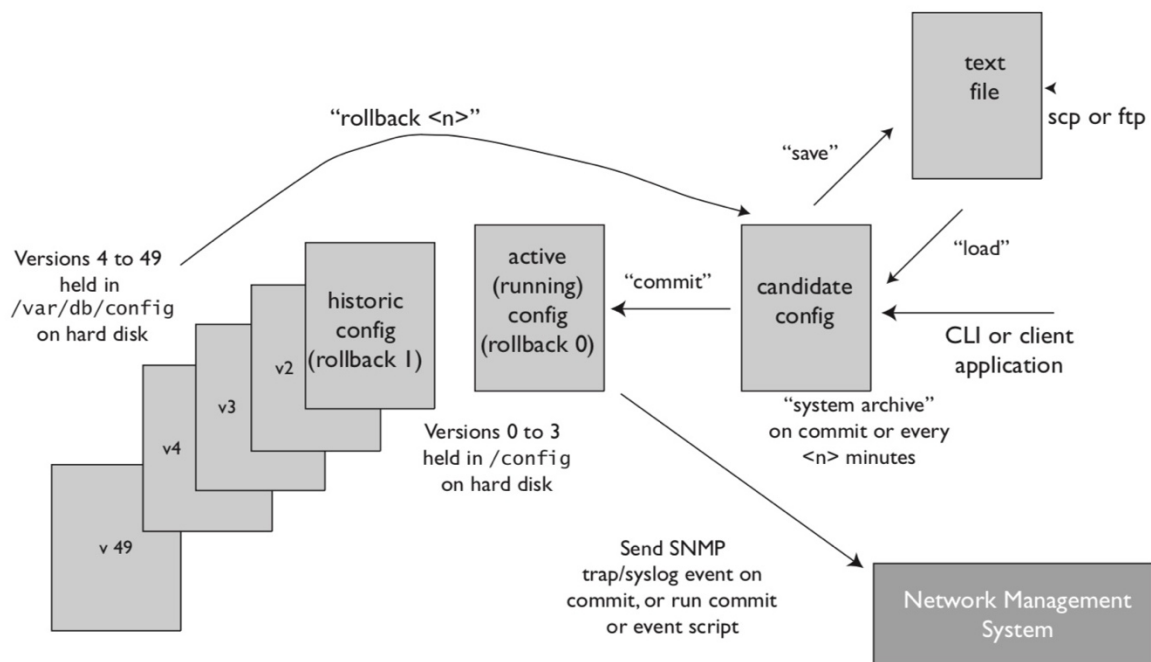
o

help apropos routing-options

In Junos I comandi impartiti non sono immediatamente operativi. Abbiamo infatti:

- **Active Configuration** che è la configurazione in funzione;
- **Candidate Configuration** NON ATTIVA, è la configurazione caricata da un file o la Active Configuration modificata dai comandi impartiti da CLI;
- **Rollback Configurations** configurazioni precedenti automaticamente archiviate.

La relazione che le lega è ben riassunta dallo schema sotto (tratto da “JUNIPER DAY ONE: EXPLORING THE JUNOS CLI” pag 49):



Quindi per rendere attivi i cambiamenti si usa il comando *commit* (ovviamente in modalità Configuration), con eventualmente diversi parametri che ne modificano il comportamento:

- *commit* copia la Candidate Configuration sulla Active Configuration;
- *commit at <orario>* esegue il *commit* a un determinato orario;
- *commit check* fa soltanto un controllo sintattico (non attiva la configurazione);
- *commit confirmed* fa il *commit*, ma dopo 10 minuti torna indietro se non si conferma il cambiamento ridando nuovamente il comando *commit* (ovviamente MOLTO UTILE per cambi di configurazione a rischio)

L'archivio delle configurazioni precedenti, al di là della funzione di primo backup, è utile per verificare eventuali cambiamenti effettuati e, in generale confrontare varie versioni della configurazione. A questo riguardo è importante sottolineare che:

- i comandi *show* (nelle varie accezioni che abbiamo incontrato) visualizzano la Candidate Configuration;
- le Rollback Configuration sono numerate in ordine di vecchiaia (Rollback 5 precedente a Rollback 4, per esempio);

- la Rollback Configuration **0** è la Active Configuration

È quindi possibile confrontare la Candidate con le Rollback/Active in modalità Operational con:

show configuration / compare rollback <numero>

oppure in modalità Configuration (nel qual caso con il comando sotto il confronto interesserà SOLO la sezione corrente, e quindi l'intera configurazione solo se siamo alla radice delle sezioni come nell'esempio sotto):

show / compare rollback <numero>

```
[edit]
root@provaJunos# show / compare rollback ?
Possible completions:
 0      2018-03-01 15:56:17 UTC by es_reti via cli
 1      2018-03-01 15:53:06 UTC by es_reti via cli
 2      2018-02-28 19:41:55 UTC by es_reti via cli
 3      2018-02-28 19:41:47 UTC by es_reti via cli
 4      2018-02-28 19:41:42 UTC by es_reti via cli commit
confirmed, rollback in 10mins
 5      2018-02-28 19:41:21 UTC by es_reti via cli
 6      2018-02-28 19:26:48 UTC by es_reti via cli
 7      2018-02-28 19:07:55 UTC by es_reti via cli
 8      2018-02-28 19:03:42 UTC by baro via cli
 9      2018-02-28 18:51:18 UTC by baro via cli
10      2018-02-28 18:41:34 UTC by baro via cli
11      2018-02-28 18:40:11 UTC by baro via cli
12      2018-02-28 18:30:01 UTC by baro via cli
13      2018-02-28 18:27:08 UTC by baro via cli
14      2018-02-28 18:26:49 UTC by baro via cli
15      2018-02-28 18:24:13 UTC by baro via cli
16      2018-02-28 18:06:48 UTC by baro via cli
17      2018-02-28 17:56:02 UTC by root via cli
18      2018-02-28 17:51:50 UTC by root via cli
19      2018-02-27 16:45:47 UTC by root via other
[edit]
root@provaJunos# show / compare rollback 0
[edit]
root@provaJunos#
```

Nel caso dell'esempio la Candidate risulta uguale alla Rollback 0 (la Active), il che significa che non ci sono state variazioni di configurazione dall'ultimo *commit*.

Per copiare una configurazione precedente nella Candidate per poi poterne eventualmente fare il *commit* si utilizza il comando:

rollback <numero di configurazione da rendere Candidate>

Per copiare la configurazione da/verso TFTP server esterni si può ricorrere alla **modalità Shell**. In realtà esistono anche altre possibilità (come per esempio la copia over ssh) maggiormente integrate nella modalità Operational e più dirette (probabilmente volutamente, per incentivarne l'uso); per ora l'idea è di documentare una modalità basica per poter garantire rapidamente un'operatività di minima.

Per copiare la **Active Configuration da Junos a un server TFTP esterno**:

```
root@provaJunos% cd /config
root@provaJunos% tftp 192.0.2.206
tftp> binary
tftp> put juniper.conf.gz junosvmx_20180305.gz
Sent 545 bytes in 0.1 seconds
tftp> quit
root@provaJunos%
```

Solo alcune note:

- è necessario accendere in modalità Shell come *root* perché sono necessari i permessi di lettura sul file della configurazione
- la Active Configuration è contenuta nel file */config/juniper.conf.gz*
- il server *192.0.2.206* e il file *junosvmx_20180305.gz* di destinazione sono ovviamente solo esempi e andranno scelti come da necessità

Per caricare una **configurazione da un server TFTP esterno e utilizzarla come Candidate Configuration**, sempre da modalità Shell:

```
root@provaJunos% cd /config
root@provaJunos% tftp 192.0.2.206
tftp> binary
tftp> get junosvmx_20180305.gz
Sent 545 bytes in 0.1 seconds
tftp> quit
root@provaJunos% cli
root@provaJunos> configure
Entering configuration mode
The configuration has been changed but not committed

[edit]
root@provaJunos# load override ?
Possible completions:
<filename>      Filename (URL, local, remote, or floppy)
db_ext          Size: 0, Last changed: Mar 05 12:19:15
db_ext-         Size: 0, Last changed: Mar 01 17:19:43
juniper.conf.1.gz Size: 544, Last changed: Mar 05 12:19:15
juniper.conf.2.gz Size: 545, Last changed: Mar 01 15:56:16
juniper.conf.3.gz Size: 545, Last changed: Mar 01 15:53:01
juniper.conf.gz  Size: 545, Last changed: Mar 05 12:24:26
juniper.conf.md5 Size: 32, Last changed: Mar 01 17:11:48
junosvmx_20180305.gz Size: 545, Last changed: Mar 05 14:29:10
terminal        Use login terminal
usage.db         Size: 1932, Last changed: Feb 27 16:46:20
```

```
vchassis/                Last changed: Feb 27 16:52:24
[edit]
root@provaJunos# load override junosvmx_20180305.gz
load complete

[edit]
root@provaJunos#
```

- ancora una volta i nomi del server e dei file coinvolti sono solo un esempio
- ancora una volta è necessario accedere come *root* (anzi a maggior ragione di prima, visto che adesso dobbiamo scrivere un file)
- nel comando *get* si sarebbe potuto specificare un nome diverso per il file di destinazione su Junos aggiungendolo come parametro in coda al comando (come fatto nell'esempio prima per la *put*)
- il comando *load* carica il file indicato come Candidate Configuration: l'opzione *override* sovrascrive completamente la precedente Candidate Configuration, altre opzioni adottano approcci meno invasivi (vedi *load ?*)