



Selenium Grid

Ing. Simone Giustetti
www.giustetti.net

Selenium Grid

Selenium Grid permette di eseguire i test scritti per **Selenium Web Driver** o **Selenium IDE** su una moltitudine di macchine, via rete, in parallelo.

Funziona come un server proxy nascondendo il dominio di origine dei test e simulando la “Same Origin Policy”.

Non è necessario installare software sui server del dominio oggetto di analisi.



Obiettivi di Selenium Grid

Selenium Grid ha lo scopo di:

- Fornire uno strumento semplice da usare e configurare per eseguire test in parallelo su più macchine.
- Consentire di eseguire test su versioni differenti del medesimo browser.
- Consentire di eseguire lo stesso test su browser diversi e sistemi operativi diversi.



Selenium Grid 4 è costituito da 6 componenti distinti.

La suddivisione di compiti specifici tra servizi / macchine distinti garantisce:

- Alta scalabilità.
- Possibilità di sfruttare i contenitori.
- Migliore bilanciamento dei carichi di lavoro.

Più componenti possono essere installati sulla medesima macchina.



I ruoli che costituiscono **Selenium Grid 4** sono:

- **Router**;
- **Session Queue**;
- **Distributor**;
- **Session Map**;
- **Event Bus**;
- **Node**.

I componenti colorati in blu costituiscono un **Hub**.



Il Router

Il **router** è il punto di accesso a Selenium Grid. Ha il compito di:

- Ricevere tutte le richieste esterne e indirizzarle al componente adatto.
- Passare le nuove richieste a Session Queue.
- Passare al nodo corretto comandi per le richieste in esecuzione.
- Bilanciare il carico di lavoro tra le macchine.



Il Distributore

Il **distributor** svolge 2 compiti:

- Registrare i nodi e tenere traccia delle loro caratteristiche.
- Interrogare la coda delle nuove sessioni ed inoltrarle ai nodi liberi più adatti. Quando assegna una sessione segnala l'associazione sessione ↔ nodo alla mappa delle sessioni.



La **session map** registra e tiene traccia delle sessioni di test attive e delle macchine su cui girano.

La **session queue** è una lista FIFO in cui sono registrate le sessioni di test in attesa di essere assegnate ad un nodo.



Il Bus degli Eventi

Lo **event bus** mette in comunicazione tutti gli altri componenti di Selenium Grid.

La comunicazione tra gli elementi di Selenium Grid avviene attraverso messaggi piuttosto che chiamate HTTP onde garantire prestazioni migliori ed un carico inferiore.

Lo **event bus** è il primo componente che dovrebbe essere avviato in una infrastruttura distribuita tra più macchine distinte.



I Nodi

I **nodi** sono le macchine su cui vengono eseguiti i test. Selenium Grid è costituita da uno o più nodi.

Ogni nodo si registra presso il distributore attraverso lo event bus. Durante la registrazione invia informazioni circa la sua configurazione e le risorse hardware di cui dispone.

I nodi eseguono i comandi senza apportare alcuna verifica.



Selenium Grid viene distribuito nella forma di un archivio **jar** completo che include tutti i componenti.

Verificare che sulla macchina sia disponibile una Java Virtual Machine compatibile con Java \geq **11**.

Installare i browser su cui verranno eseguiti i test.

Installare i driver per i browser corrispondenti.



Aggiungere il percorso ove sono installati i driver di Selenium al **PATH** del proprio utente.

Linux / MacOSX:

```
export PATH=$PATH:./<path to driver>
```

Windows:

```
setx PATH "%PATH%;C:\WebDriver\bin"
```



Avviare Selenium Grid Standalone

Per avviare tutti i componenti sul medesimo server:

```
java -jar selenium-server-<version>.jar \
standalone
```

È possibile verificare che il server sia attivo collegandosi al seguente URL:

<http://localhost:4444>



Avviare Selenium Grid Hub / Node

Altre modalità in cui Selenium Grid può girare:

```
java -jar selenium-server-<version>.jar hub
```

```
java -jar selenium-server-<version>.jar node
```

Per far girare più nodi sulla medesima macchina avviare più volte il server assegnando ad ogni sessione una porta diversa con l'opzione *--port*:

```
java -jar selenium-server-<version>.jar node \  
  --port 5555
```

```
java -jar selenium-server-<version>.jar node \  
  --port 5556
```



Avviare Selenium Grid Hub / Node

Per far puntare i nodi ad un hub installato su di una macchina distinta bisogna impostare l'opzione *--hub*:

```
java -jar selenium-server-<version>.jar node \  
  --hub http://<hub IP>:4444
```



Per avviare Selenium Grid in modalità distribuita:

1) Avviare Event Bus

```
java -jar selenium-server-<version>.jar event-bus \  
  --publish-events tcp://<event bus IP>:4442 \  
  --subscribe-events tcp://<event bus IP>:4443 \  
  --port 5557
```

2) Avviare New Session Queue

```
java -jar selenium-server-<version>.jar sessionqueue \  
  --port 5559
```



3) Avviare Session Map

```
java -jar selenium-server-<version>.jar sessions \  
  --publish-events tcp://<event bus IP>:4442 \  
  --subscribe-events tcp://<event bus IP>:4443 \  
  --port 5556
```

4) Avviare Distributor

```
java -jar selenium-server-<version>.jar distributor \  
  --publish-events tcp://<event bus IP>:4442 \  
  --subscribe-events tcp://<event bus IP>:4443 \  
  --sessions http://<sessions IP>:5556 \  
  --sessionqueue http://<new session queue IP>:5559 \  
  --port 5553 --bind-bus false
```



5) Avviare il Router

```
java -jar selenium-server-<version>.jar router \
  --sessions http://<sessions IP>:5556 \
  --distributor http://<distributor IP>:5553 \
  --sessionqueue http://<new session queue IP>:5559 \
  --port 4444
```

6) Avviare i Nodi (Cambiare la porta per ognuno)

```
java -jar selenium-server-<version>.jar node \
  --publish-events tcp://<event bus IP>:4442 \
  --subscribe-events tcp://<event bus IP>:4443 \
  --port 5555
```



Configurare i Componenti di Selenium Grid

Tutti i componenti possono essere configurati attraverso un terminale mediante appositi comandi.

L'elenco dei comandi dipende dal componente.

L'elenco dei comandi può essere visionato utilizzando l'opzione *--help*:

```
java -jar selenium-server-<version>.jar node --help  
java -jar selenium-server-<version>.jar router --help  
java -jar selenium-server-<version>.jar standalone \  
--help
```



Informazioni & Licenze

LICENZA

Salvo dove altrimenti specificato grafica, immagini e testo della presente opera sono © Simone Giustetti. L'opera può essere ridistribuita per fini non commerciali secondo i termini della licenza:

Creative Commons Attribuzione - Non commerciale - Condividi allo stesso modo 4.0 Internazionale



È possibile richiedere versioni rilasciate sotto diversa licenza scrivendo all'indirizzo: studiosg@giustetti.net

TRADEMARK

- FreeBSD è un trademark di The FreeBSD Foundation.
- Linux è un trademark di Linus Torvalds.
- Macintosh, OS X e Mac OS X sono tutti trademark di Apple Corporation.
- MariaDB è un trademark di MariaDB Corporation Ab.
- MySQL è un trademark di Oracle Corporation.
- Selenium è un trademark di Software Freedom Conservancy.
- UNIX è un trademark di The Open Group.
- Windows e Microsoft SQL Server sono trademark di Microsoft Corporation.
- Alcuni algoritmi crittografici citati nella presente opera potrebbero essere protetti da trademark.

Si prega di segnalare eventuali errori od omissioni al seguente indirizzo: studiosg@giustetti.net

