


















# Konfiguracja środowiska dla Selenium Grid

Selenium Grid jest doskonałym narzędziem organizującym wykonywanie testów automatycznych dla Selenium RC lub WebDriver. Znacząco minimalizuje czas potrzebny na wykonanie wielu testów poprzez testowanie równoległe i dystrybucję testów na komputery podłączone do siatki. Główną jednak cechą Selenium Grid jest zarządzanie wieloma środowiskami z centralnego punktu dzięki czemu łatwo można uruchomić testy w ogromnej kombinacji przeglądarek i środowisk operacyjnych.

Jednak aby w pełni wykorzystać dobrodziejstwa jakie daje Selenium Grid należy odpowiednio skonfigurować środowisko jego pracy. Poniżej przedstawiony zostanie sposób konfiguracji podstawowego środowiska dla testów w siatce Selenium Grid.

1. Upewniamy się, że mamy zainstalowane środowisko uruchomieniowe Javy na komputerze Java JDK/Java JRE poleceniem `java -version` w konsoli systemowej. Jeśli nie, wówczas pobieramy ostatnią jej wersję ze strony

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Product / File Description	File Size	Download
Linux ARM v6/v7 Hard Float ABI	67.7 MB	 <a href="#">jdk-7u51-linux-arm-vfp-hflt.tar.gz</a>
Linux ARM v6/v7 Soft Float ABI	67.68 MB	 <a href="#">jdk-7u51-linux-arm-vfp-sflt.tar.gz</a>
Linux x86	115.65 MB	 <a href="#">jdk-7u51-linux-i586.rpm</a>
Linux x86	132.98 MB	 <a href="#">jdk-7u51-linux-i586.tar.gz</a>
Linux x64	116.96 MB	 <a href="#">jdk-7u51-linux-x64.rpm</a>
Linux x64	131.8 MB	 <a href="#">jdk-7u51-linux-x64.tar.gz</a>
Mac OS X x64	179.49 MB	 <a href="#">jdk-7u51-macosx-x64.dmg</a>
Solaris x86 (SVR4 package)	140.02 MB	 <a href="#">jdk-7u51-solaris-i586.tar.Z</a>
Solaris x86	95.13 MB	 <a href="#">jdk-7u51-solaris-i586.tar.gz</a>
Solaris x64 (SVR4 package)	24.53 MB	 <a href="#">jdk-7u51-solaris-x64.tar.Z</a>
Solaris x64	16.28 MB	 <a href="#">jdk-7u51-solaris-x64.tar.gz</a>
Solaris SPARC (SVR4 package)	139.39 MB	 <a href="#">jdk-7u51-solaris-sparc.tar.Z</a>
Solaris SPARC	98.19 MB	 <a href="#">jdk-7u51-solaris-sparc.tar.gz</a>
Solaris SPARC 64-bit (SVR4 package)	23.94 MB	 <a href="#">jdk-7u51-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	18.33 MB	 <a href="#">jdk-7u51-solaris-sparcv9.tar.gz</a>
Windows x86	123.64 MB	 <a href="#">jdk-7u51-windows-i586.exe</a>
Windows x64	125.46 MB	 <a href="#">jdk-7u51-windows-x64.exe</a>

2. Ze strony projektu Selenium pobieramy paczkę z biblioteką serwera Selenium

<http://docs.seleniumhq.org/download/>

## Selenium Server (formerly the Selenium RC Server)

The Selenium Server is needed in order to run either Selenium RC style scripts or Remote Selenium Webdriver ones. The 2.x server is a drop-in replacement for the old Selenium RC server and is designed to be backwards compatible with your existing infrastructure.

Download version [2.39.0](#)

To use the Selenium Server in a Grid configuration [see the wiki page](#).

3. Uruchamiamy Hub Selenium Grid wydając polecenie w konsoli

```
java -jar selenium-server-standalone-xx.jar --role hub
```

gdzie xx to numer ostatniego wydania biblioteki

Hub Selenium Grid rozpoczął w tym momencie pracę na domyślnym porcie 4444 i oczekuje na podłączenie serwerów Selenium. Numer portu można zmienić podając kolejny parametr `--port xxxx`

4. W przeglądarce wpisujemy adres <http://localhost:4444/grid/console> i otrzymujemy widok siatki serwerów Selenium podłączonych do naszego Huba. W tym momencie nie mamy podłączonego żadnego serwera więc będziemy widzieć tylko ustawienia samego Huba Selenium.
5. Podłączamy serwer Selenium do Huba Selenium Grid poleceniem  
*java -jar selenium-server.jar -role node -hub http://localhost:4444/grid/register -port 5555*  
Po wydaniu tego polecenia i podłączeniu się serwera do Huba odświeżamy widok w przeglądarce



6. Na koniec w kodzie naszego skryptu testowego zamieniamy dotychczasową definicję uruchamiania przeglądarki

```
@BeforeClass
public void beforeClass() {
    driver = new FirefoxDriver();
}
```



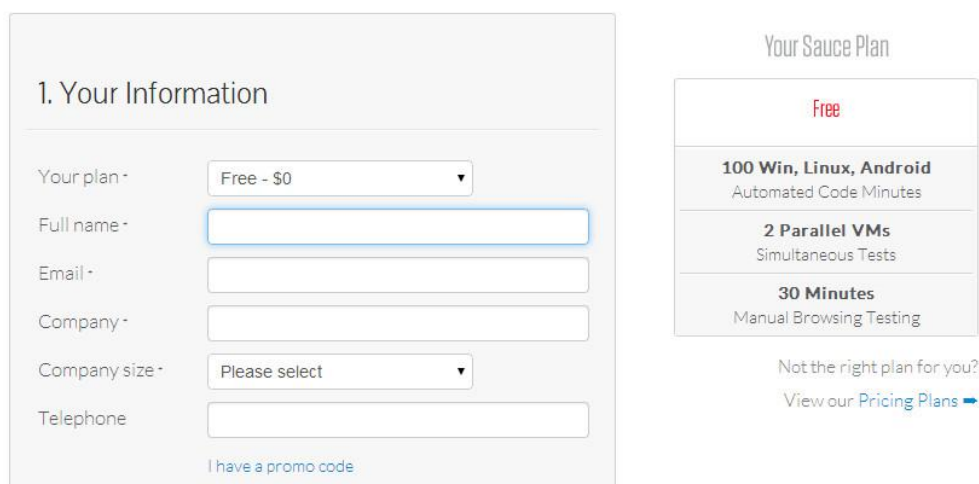
```
@BeforeClass
public void setUp() throws Exception
{
    driver = new RemoteWebDriver(new URL("http://localhost:4444/wd/hub"), DesiredCapabilities.firefox());
}
```

Podczas testowania kompatybilności aplikacji internetowej należy sprawdzić jej działanie na co najmniej kilku najpopularniejszych przeglądarkach. Pociąga to za sobą wymóg posiadania odpowiedniej infrastruktury w postaci kombinacji przeglądarek, ich wersji oraz systemów operacyjnych na których mogą być zainstalowane.

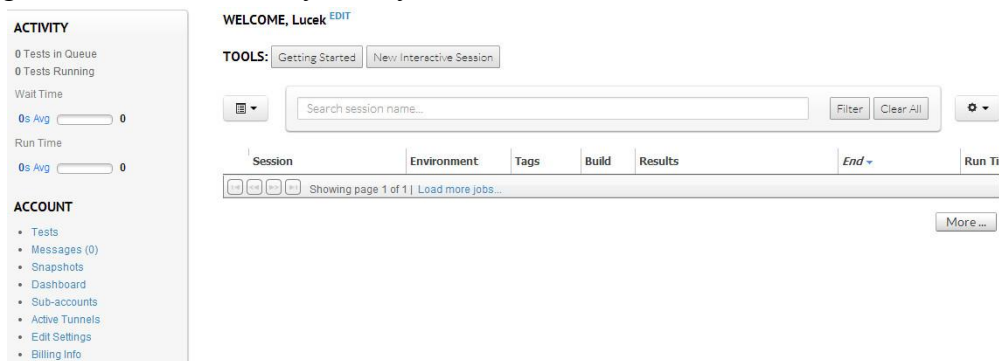
Z pomocą w takich sytuacjach przychodzą nam rozwiązania oferowane w tzw. chmurach obliczeniowych (cloud computing), z których jedną z najlepszych ofert posiada SauceLabs - <https://saucelabs.com/home>

Jak przygotować się do pracy z SauceLabs opisane zostanie w poniższych krokach

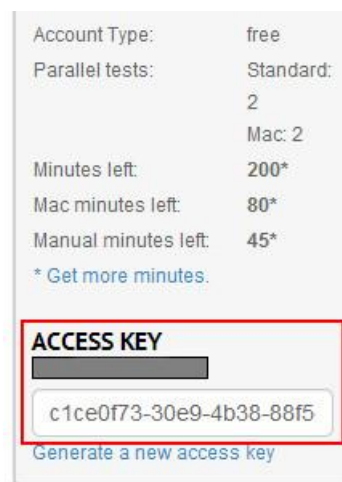
1. Na stronie projektu należy zarejestrować darmowe konto, za pomocą którego będziemy korzystać z zasobów SauceLabs - <https://saucelabs.com/home>



2. Po poprawnym zarejestrowaniu się zostaniemy przeniesieni na stronę administracyjną naszego konta z widokiem wykonanych testów



3. Niezbędnym elementem w procesie łączenia się do chmury SauceLabs jest klucz identyfikujący użytkownika i znaleźć można go u dołu powyższej strony



4. Znając nazwę użytkownika i klucz dostępu można połączyć się z chmurą w celu uruchomienia testu

```
@BeforeClass
public void beforeClass() throws MalformedURLException {
    DesiredCapabilities capabilities = DesiredCapabilities.firefox();
    capabilities.setCapability("version", "23");
    capabilities.setCapability("platform", Platform.WINDOWS);
    driver = new RemoteWebDriver(
        new URL("http://Lukasz:899bd9f5-56b85f25dd8e@ondemand.saucelabs.com:80/wd/hub"),
        capabilities);
    driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
    driver.navigate().to("http://poczta.interia.pl");
}
```

5. Jeśli chodzi o konfigurację przeglądarki to pod adresem <https://saucelabs.com/platforms> znajduje się wykaz wszystkich obsługiwanych przeglądarek w chmurze

Automate your test!

- 1) Define the browser
- 2) Enter a platform and version.

Can't find your browser?  
Change 'webdriver' to 'selenium'

WebDriver java

WebDriver capabilities caps = DesiredCapabilities.android();  
setCapability("platform", "Linux");  
setCapability("version", "4.0");

or

Start manual session