

STANDARD OPERATING PROCEDURE [SOP]

Setup Guide for Selenium Grid

Purpose:

To set up Selenium Grid, which enables the distributed execution of tests across different browsers and machines.

Selenium Grid allows parallel test execution, reducing the overall test execution time.

Scope

This procedure applies to all users involved in setting up and managing the Selenium Grid infrastructure for automated testing in a [Development/Testing/QA] environment.

Document detail	
Reference Number	SOP – 001
Version	1.0
Effective date	29/01/2025
Prepared by	Shivom Kumari
Reviewed by	Ankur Kumar
Approved by	

Table of Content		
1	Prerequisites	3
2	Tools & Software Required	3
3	Procedure	3
4	Maintenance and Monitoring	6
5	Troubleshooting	6
6	Conclusion	6

Prerequisites

Before setting up Selenium Grid, ensure the following:

- Java 8 or higher installed on the machine (for both Hub and Node)
- Selenium WebDriver libraries
- Maven or Gradle (if dependencies are managed using these tools)
- A stable internet connection

Tools & Software Required

- **Selenium Standalone Server** (JAR file)
- **Java Development Kit (JDK)** (Version 8 or later)
- **Browser drivers** (e.g., ChromeDriver, GeckoDriver, etc.)
- **Operating System:** Compatible with the system (Windows/Linux/Mac)

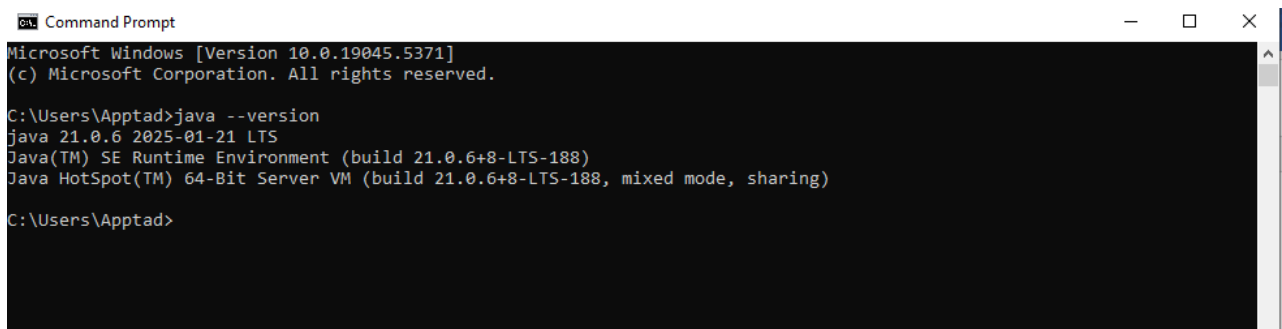
Procedure

Step 1: Install Java Development Kit (JDK)

1. Download and install the Java Development Kit (JDK) from [Oracle's website](https://www.oracle.com/in/java/technologies/javase-downloads.html) or use your preferred version.

Linux	macOS	Windows
Product/file description	File size	Download
x64 Compressed Archive	228.70 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.zip (sha256)
x64 Installer	205.21 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.exe (sha256)
x64 MSI Installer	203.96 MB	https://download.oracle.com/java/23/latest/jdk-23_windows-x64_bin.msi (sha256)

2. Verify installation by running the following command in the terminal/command prompt:



```
Command Prompt
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Apptad>java --version
java 21.0.6 2025-01-21 LTS
Java(TM) SE Runtime Environment (build 21.0.6+8-LTS-188)
Java HotSpot(TM) 64-Bit Server VM (build 21.0.6+8-LTS-188, mixed mode, sharing)

C:\Users\Apptad>
```

Step 2: Download Selenium Standalone Server

1. Go to the Selenium official website: <https://www.selenium.dev/downloads/>

Selenium Server (Grid)

The Selenium Server is needed in order to run Remote Selenium WebDriver (Grid).

Latest stable version [4.28.1](#)

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

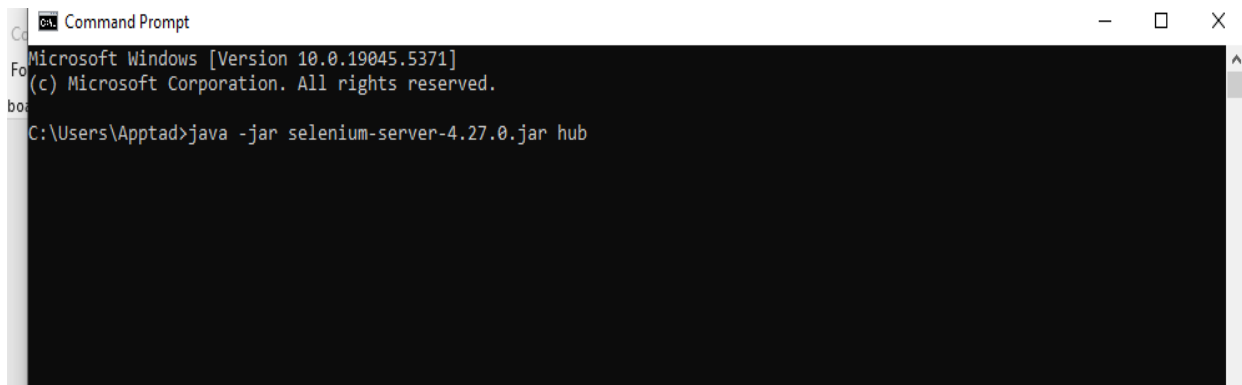
To run the Grid with popular browsers using Docker see the [repository](#).

To deploy the Grid to Kubernetes cluster see the Helm chart [configuration](#).

2. Download the latest version of the Selenium Standalone Server (JAR file).
3. Save the JAR file to a known directory (e.g., C:\selenium or /usr/local/selenium).

Step 3: Start Selenium Hub

1. Navigate to the directory where you saved the Selenium JAR file.
2. Open a terminal/command prompt and run the following command to start the Selenium Hub:

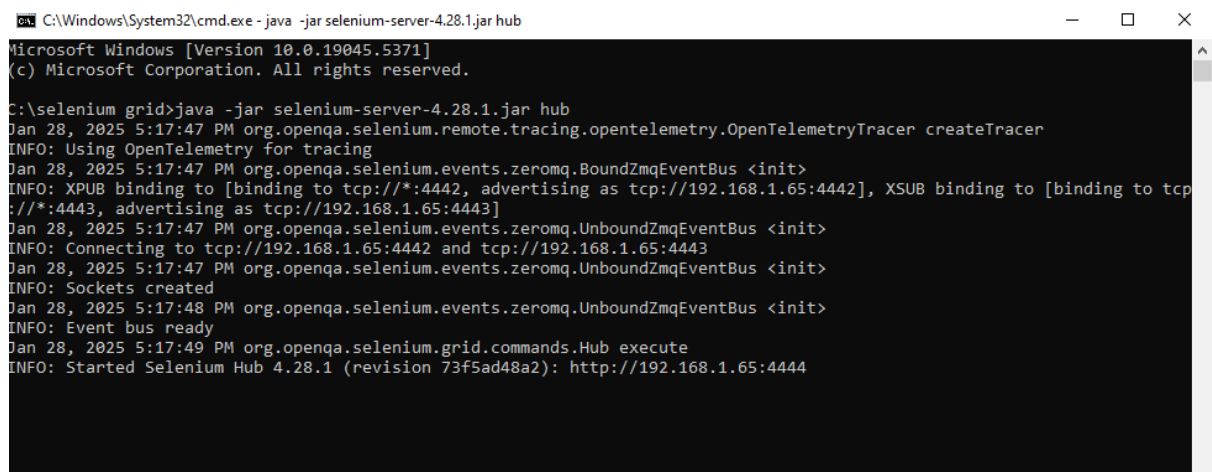


```
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Apptad>java -jar selenium-server-4.27.0.jar hub
```

3. The Hub will start and listen for incoming connections on port 4444 by default. You can confirm it by navigating to the following URL in your browser:

<http://localhost:4444/grid/console>



```
C:\Windows\System32\cmd.exe - java -jar selenium-server-4.28.1.jar hub
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\selenium grid>java -jar selenium-server-4.28.1.jar hub
Dan 28, 2025 5:17:47 PM org.openqa.selenium.remote.tracing.opentelemetry.OpenTelemetryTracer createTracer
INFO: Using OpenTelemetry for tracing
Dan 28, 2025 5:17:47 PM org.openqa.selenium.events.zeromq.BoundZmqEventBus <init>
INFO: XPUB binding to [binding to tcp://*:4442, advertising as tcp://192.168.1.65:4442], XSUB binding to [binding to tcp://*:4443, advertising as tcp://192.168.1.65:4443]
Dan 28, 2025 5:17:47 PM org.openqa.selenium.events.zeromq.UnboundZmqEventBus <init>
INFO: Connecting to tcp://192.168.1.65:4442 and tcp://192.168.1.65:4443
Dan 28, 2025 5:17:47 PM org.openqa.selenium.events.zeromq.UnboundZmqEventBus <init>
INFO: Sockets created
Dan 28, 2025 5:17:48 PM org.openqa.selenium.events.zeromq.UnboundZmqEventBus <init>
INFO: Event bus ready
Dan 28, 2025 5:17:49 PM org.openqa.selenium.grid.commands.Hub execute
INFO: Started Selenium Hub 4.28.1 (revision 73f5ad48a2): http://192.168.1.65:4444
```

Step 4: Download and Configure Node(s)

1. **On each machine** that will act as a Node (remote machine):

- Install the required browser and browser driver (e.g., ChromeDriver, GeckoDriver) on the Node machine.

2. To register a node with the Hub, run the following command on each Node machine:

```
C:\Windows\System32\cmd.exe

C:\selenium grid>java -jar selenium-server-4.28.1.jar node --hub http://192.168.1.10:4444
```

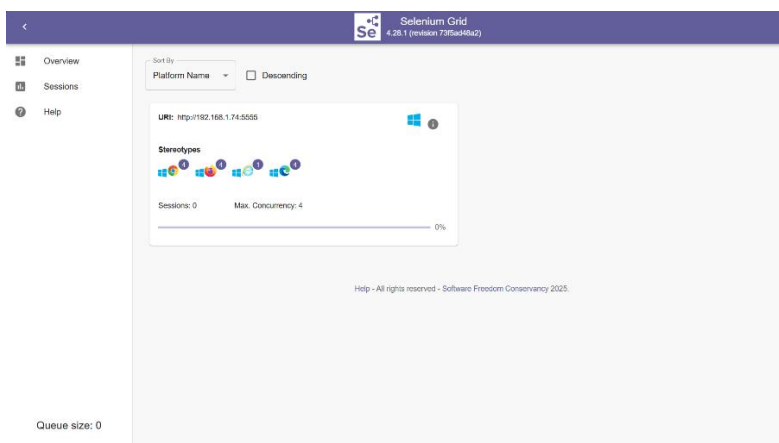
3. Verify Node registration by visiting the Hub console at <http://localhost:4444/grid/console>. You should see the Node listed under "Registered Nodes."

```
C:\Windows\System32\cmd.exe - java -jar selenium-server-4.28.1.jar node --hub http://192.168.1.17

It is advised to delete the driver in PATH and retry
Jan 29, 2025 12:05:06 PM org.openqa.selenium.manager.SeleniumManager lambda$runCommand$1
WARNING: Exception managing MicrosoftEdge: Unable to discover proper msedge driver version in offline mode
Jan 29, 2025 12:05:06 PM org.openqa.selenium.manager.SeleniumManager lambda$runCommand$1
WARNING: The msedge driver version (131.0.2903.86) detected in PATH at C:\Browser Driver\msedge driver.exe might not be compatible with the detected MicrosoftEdge version (132.0.2957.127); currently, msedge driver is recommended for MicrosoftEdge 132.0, so it is advised to delete the driver in PATH and retry
Jan 29, 2025 12:05:07 PM org.openqa.selenium.manager.SeleniumManager lambda$runCommand$1
WARNING: Exception managing firefox: Unable to discover proper geckodriver version in offline mode
Jan 29, 2025 12:05:07 PM org.openqa.selenium.manager.SeleniumManager lambda$runCommand$1
WARNING: Unable to discover proper IEDriverServer version in offline mode
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.config.NodeOptions report
INFO: Adding Internet Explorer for {"browserName": "internet explorer", "platformName": "Windows 10"} 1 times
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.config.NodeOptions report
INFO: Adding Edge for {"browserName": "MicrosoftEdge", "platformName": "Windows 10"} 4 times
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.config.NodeOptions report
INFO: Adding Firefox for {"browserName": "firefox", "platformName": "Windows 10"} 4 times
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.config.NodeOptions report
INFO: Adding Chrome for {"browserName": "chrome", "platformName": "Windows 10"} 4 times
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.Node <init>
INFO: Binding additional locator mechanisms: relative
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.httpd.NodeServer$2 start
INFO: Starting registration process for Node http://192.168.1.74:5555
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.httpd.NodeServer execute
INFO: Started Selenium node 4.28.1 (revision 73f5ad48a2): http://192.168.1.74:5555
Jan 29, 2025 12:05:07 PM org.openqa.selenium.grid.node.httpd.NodeServer$2 lambda$start$1
INFO: Sending registration event...
Jan 29, 2025 12:05:08 PM org.openqa.selenium.grid.node.httpd.NodeServer lambda$createHandlers$2
INFO: Node has been added
```

Step 5: Verifying the Setup

1. Once the Hub and Node(s) are configured and running, verify that the Node is correctly registered and can execute tests.
2. To do this, run a sample test using Selenium WebDriver on the Hub to interact with one of the Nodes.



Step 6: Configure Desired Capabilities

To ensure your tests run on specific browsers or versions, configure **Desired Capabilities** in your WebDriver script:

```
// Setup DesiredCapabilities based on the browser parameter
DesiredCapabilities capabilities = new DesiredCapabilities();
if (browser.equalsIgnoreCase("chrome")) {
    capabilities.setBrowserName("chrome");
} else if (browser.equalsIgnoreCase("firefox")) {
    capabilities.setBrowserName("firefox");
}

driver = new RemoteWebDriver(new URL("http://localhost:4444/wd/hub"), capabilities);
driver.manage().window().maximize();
driver.get(prop.getProperty("test_URL"));
wait = new WebDriverWait(driver, Duration.ofSeconds(20));
```

Step 7: Run Parallel Tests

- You can now start running tests in parallel across multiple Nodes using Selenium Grid by specifying different Desired Capabilities for each test.

Maintenance and Monitoring

- **Monitor Hub:** Ensure that the Hub is up and running continuously. You can check the console at <http://localhost:4444/grid/console>.
- **Node Management:** Periodically check that each Node is functioning properly. If a Node goes down, it should automatically be removed from the grid and no longer used for testing.
- **Update Browsers and Drivers:** Keep browser drivers (e.g., ChromeDriver, GeckoDriver) up-to-date with the latest versions to ensure compatibility with Selenium.

Troubleshooting

- **Node not connecting to Hub:** Ensure that the Hub's IP address and port are correctly specified when registering the Node. Check firewall settings to allow communication on port 4444.
- **Browser mismatch:** Verify that the correct browser and driver are installed on the Node machine, and that the Node is correctly configured with the right browser capabilities.
- **Hub not accessible:** Check that the Hub machine's firewall or network settings allow incoming traffic on port 4444.

Conclusion

By following this procedure, you will have a working Selenium Grid setup, allowing for parallel test execution across different browsers and platforms. Ensure that the Hub and Node machines are monitored and properly maintained for smooth operation.