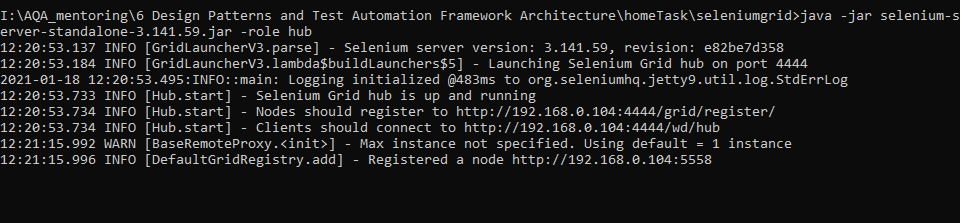
запустить центарльный сервер (hub).

**Стартуем Hub.**

Открываем окно команд и выполняем:

-jar selenium-server-standalone-3.141.59.jar -role hub

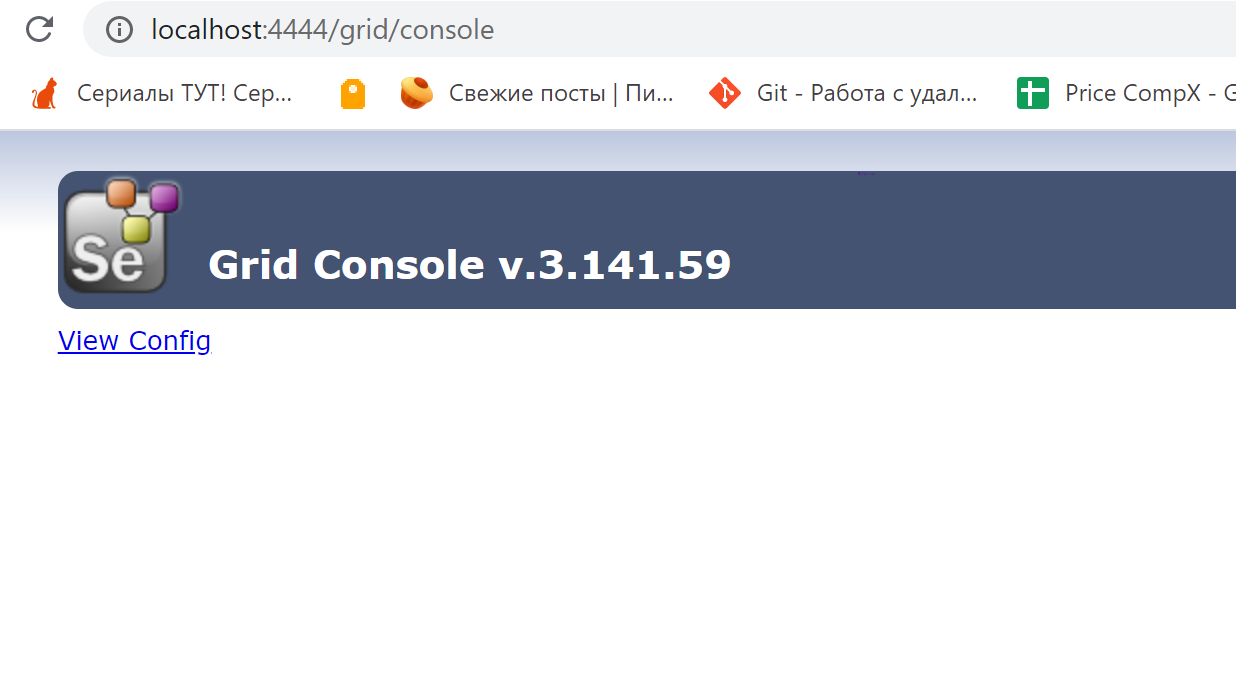


Nodes should register to http://192.168.0.104:4444/grid/register/  
 Selenium Grid hub is up and running

Первая строчка означает, что теперь мы можем регистрировать узлы на урл, где запущен hub. Вторая строчка означает,  
что hub успешно запущен.

Для того, чтобы проверить, что все стартовало правильно, нужно набрать в браузере

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



**Запускаем узлы кластера**: **Стартуем Node.**

Node — это по сути экземпляр selenium server, на котором можно стартануть n-ое количество браузеров.

В моем случае и **Hub** и **Node** расположены на одном компьютере, но это не сильно меняет ситуацию.

Открываем консоль и пишем:

Node для браузера **FireFox**

java -Dwebdriver.gecko.driver=geckodriver.exe -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://localhost:4444/grid/register -port 5559 -browser browserName=firefox

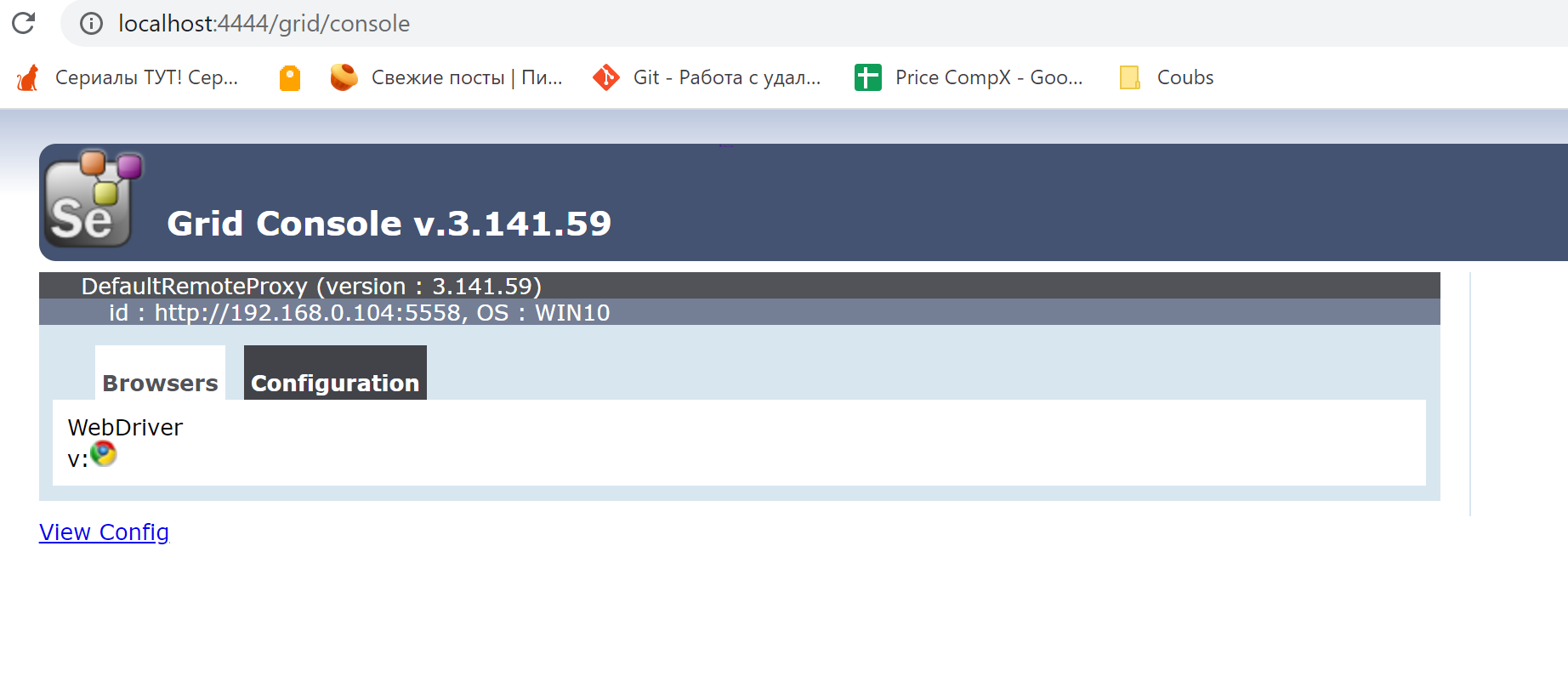
**Chrome**

java -Dwebdriver.chrome.driver=chromedriver.exe -jar selenium-server-standalone-3.141.59.jar -role webdriver -hub http://localhost:4444/grid/register -port 5558 -browser browserName=chrome

Нужно дописать путь к драйверам соответствующих браузеров

-Dwebdriver.chrome.driver=chromedriver.exe. Скачать драйверы можно на сайте selenium.org

Для того, чтобы проверить, что все настроено корректно, нужно открыть все тот же <http://localhost:4444/grid/console> и убедиться, что там отображаются истансы Chrome.



Следует добавить, конфигурировать Hub и Node можно с помощью JSON файлов.

Пример для **Hub**:

java -jar selenium-server-standalone.jar -role hub -hubConfig hubconfig.json

Пример для **Node**:

java -jar selenium-server-standalone.jar -role node -nodeConfig nodeconfig.json