**Облачные вычисления 3**

Редактируем yaml файл, подставляя пароль от VK Cloud (см. рисунок 1).

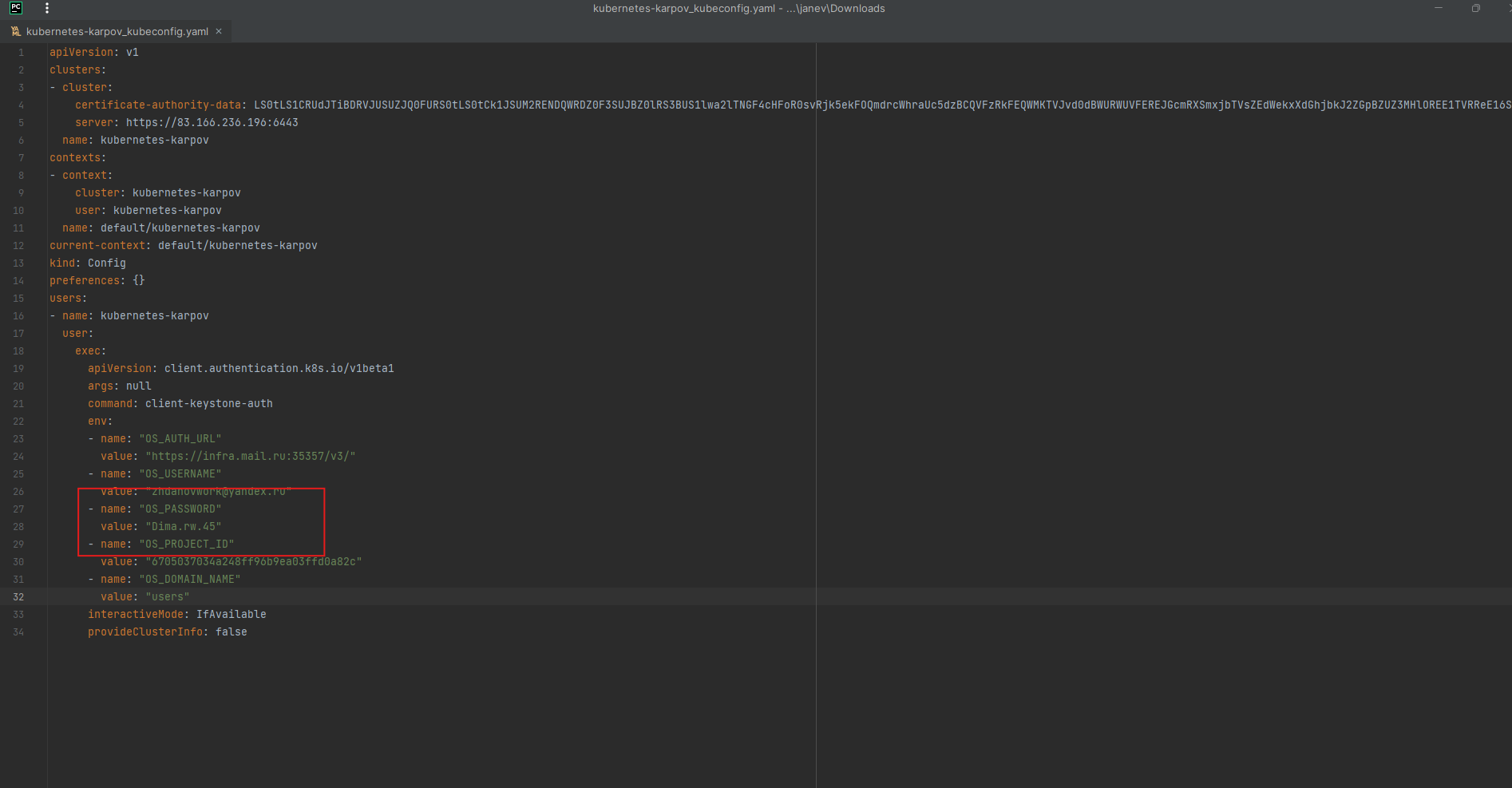


Рисунок 1.

scp -i C:/Users/janev/Downloads/karpov\_vm.pem -r C:/Users/janev/Downloads/kubernetes-karpov\_kubeconfig.yaml [ubuntu@83.166.237.165](mailto:ubuntu@83.166.237.165):. (точка в конце обязательна).

С помощью команды scp передаем файл yaml с параметрами Kubernetes, созданным в VK Cloud на виртуальную машину, созданную в первом уроке (см. рисунок 2).

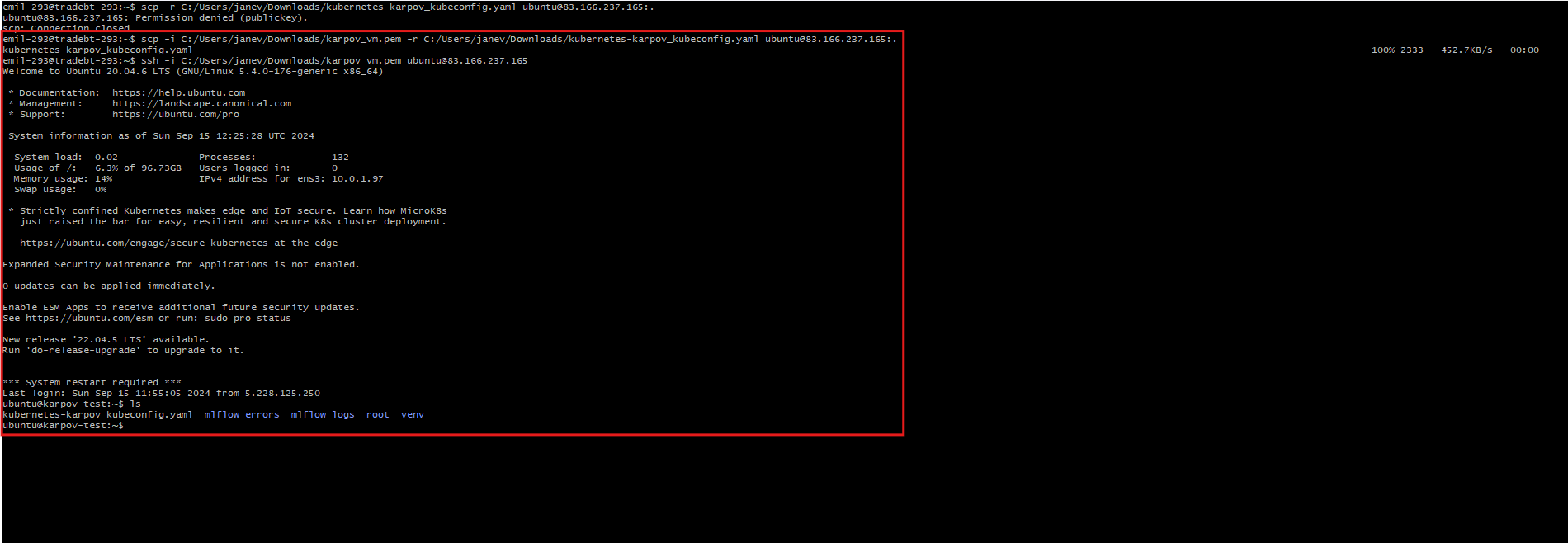


Рисунок 2.

Задаём переменную KUBECONFIG (см. рисунок 3).

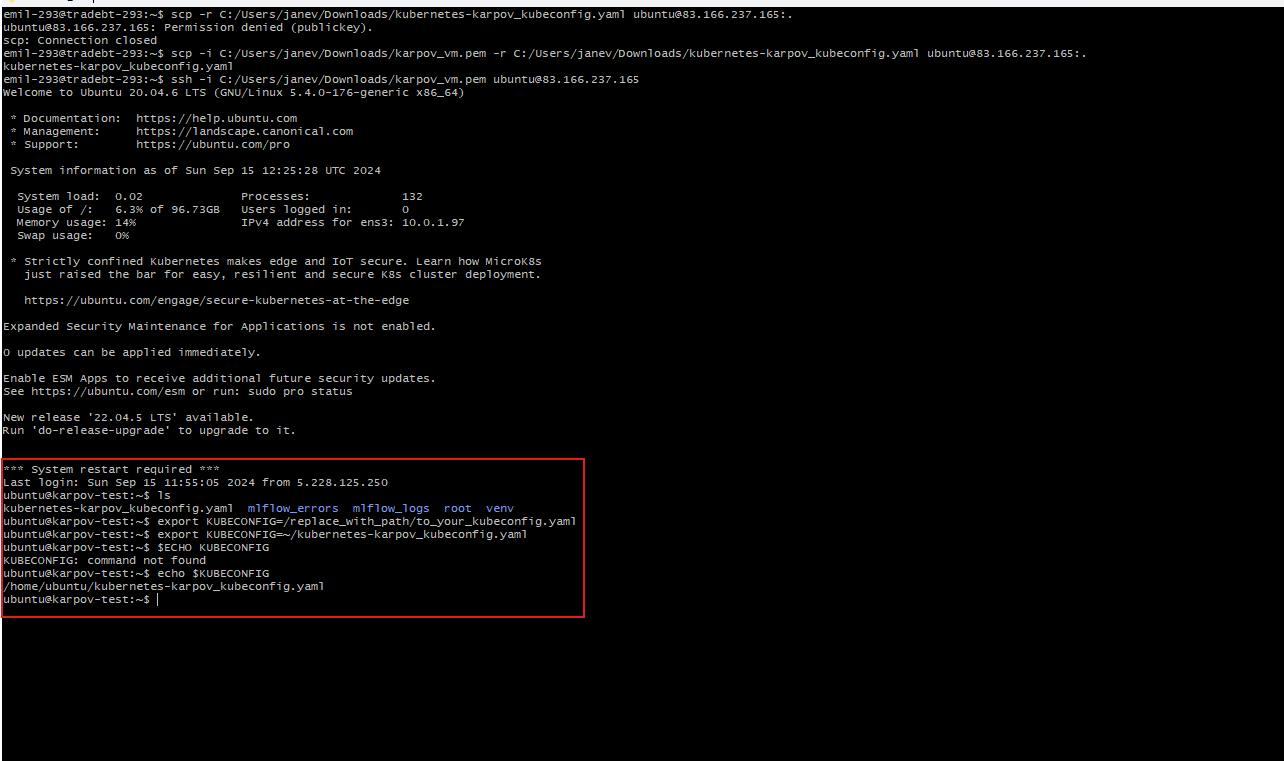


Рисунок 3.

Включаем двухфакторную аутентификацию и доступ по API (см. рисунок 4)

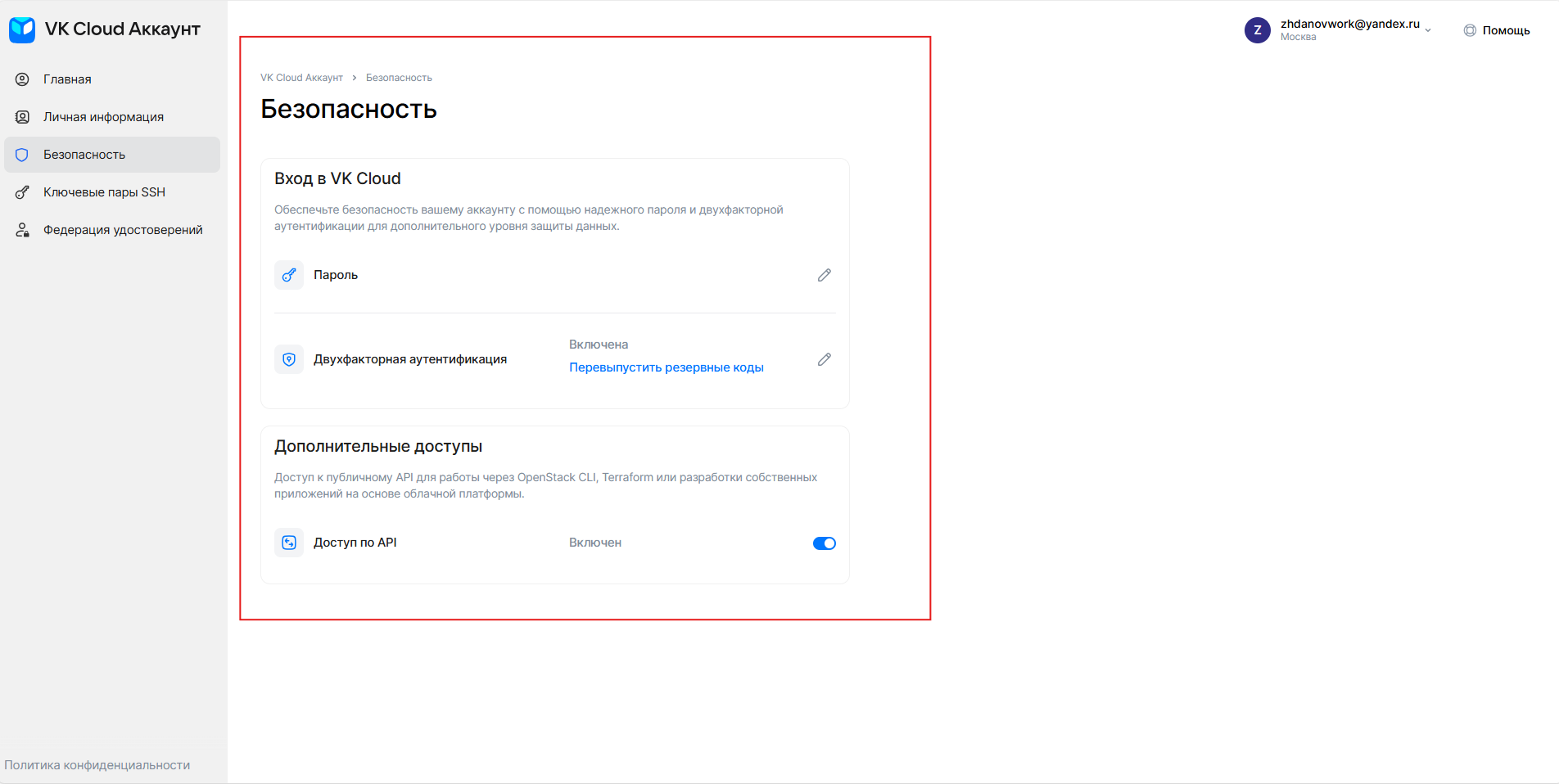


Рисунок 4.

Успешно заходим на кластер Kubernetes (см. рисунок 5)

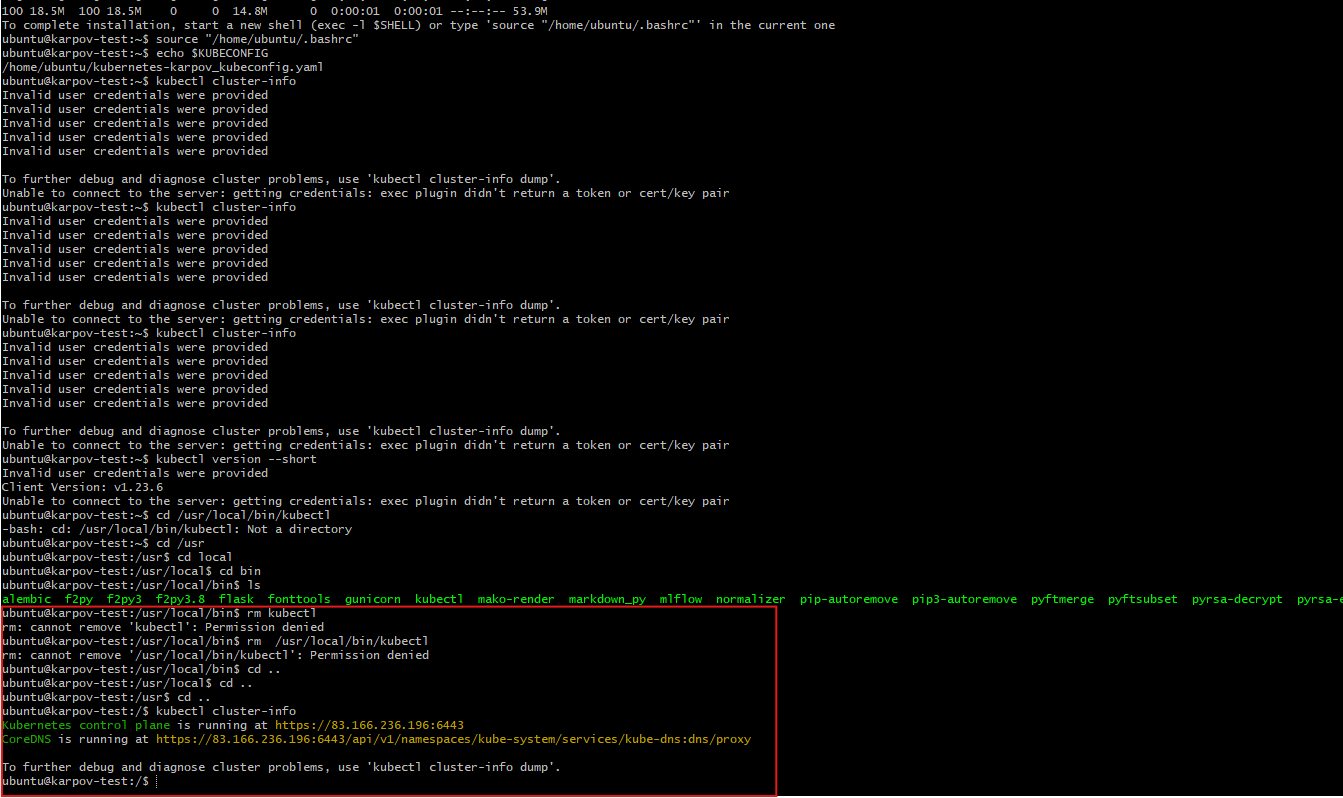


Рисунок 5.

Список запущенных служебных POD’ов (см. рисунок 6).

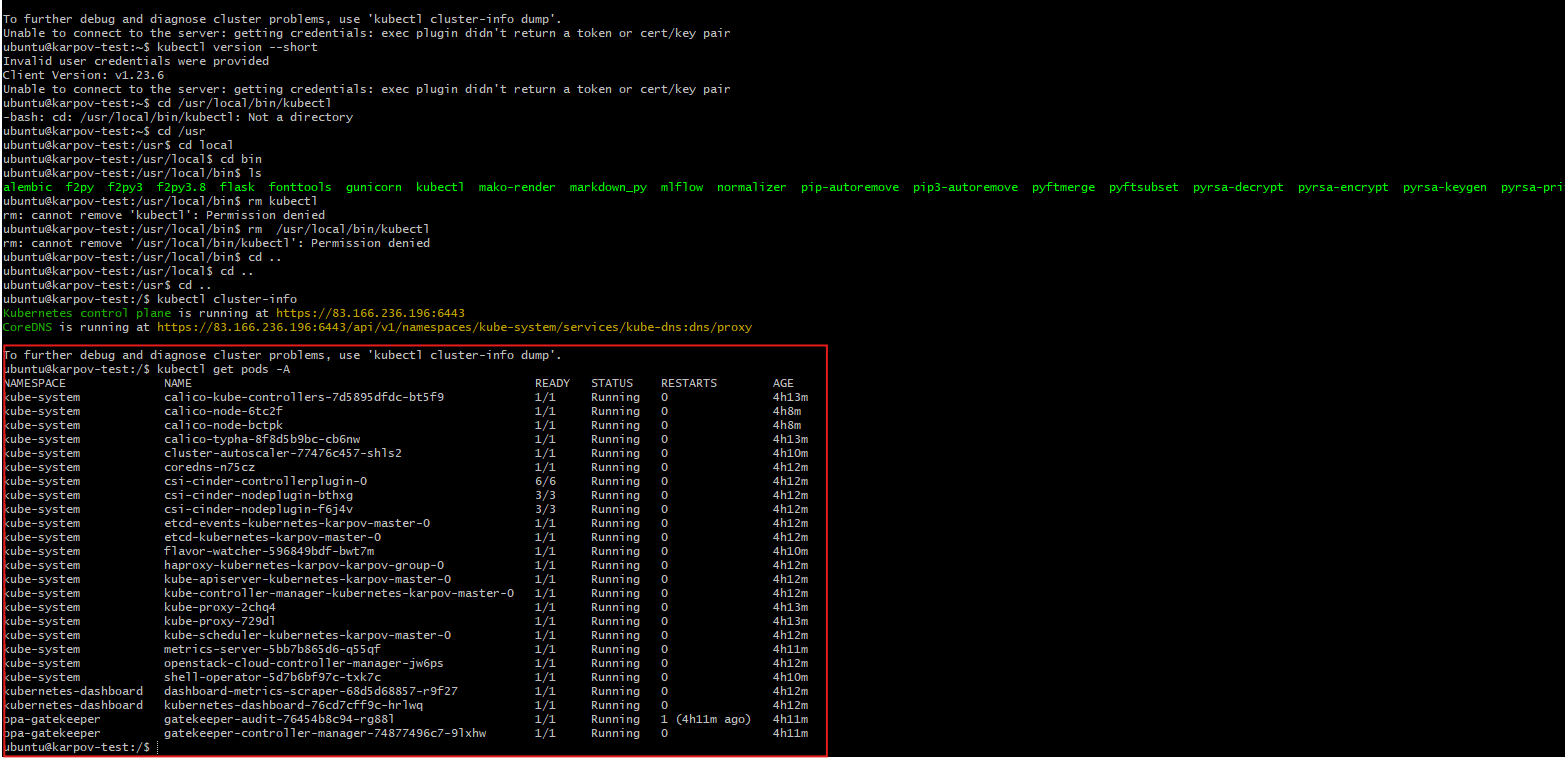


Рисунок 6.

Добавим настройки для удобства (см. рисунок 7).

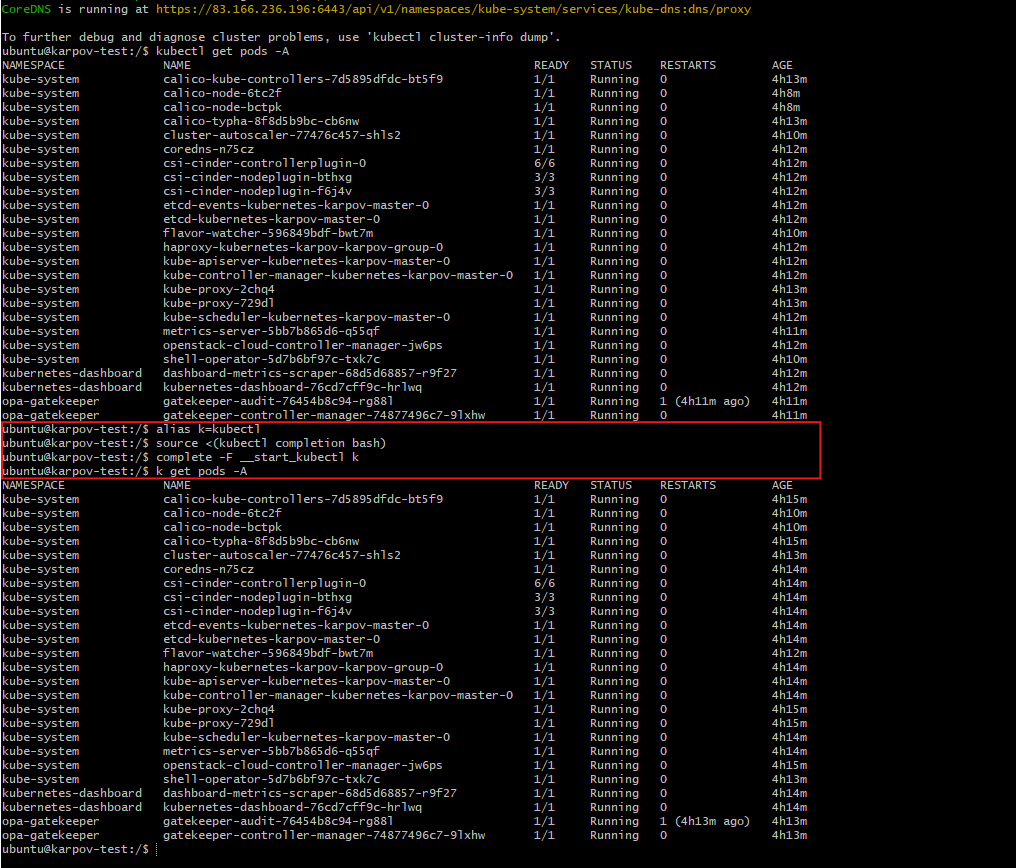


Рисунок 7.

Удаляем Gatekeeper (см. рисунок 8).

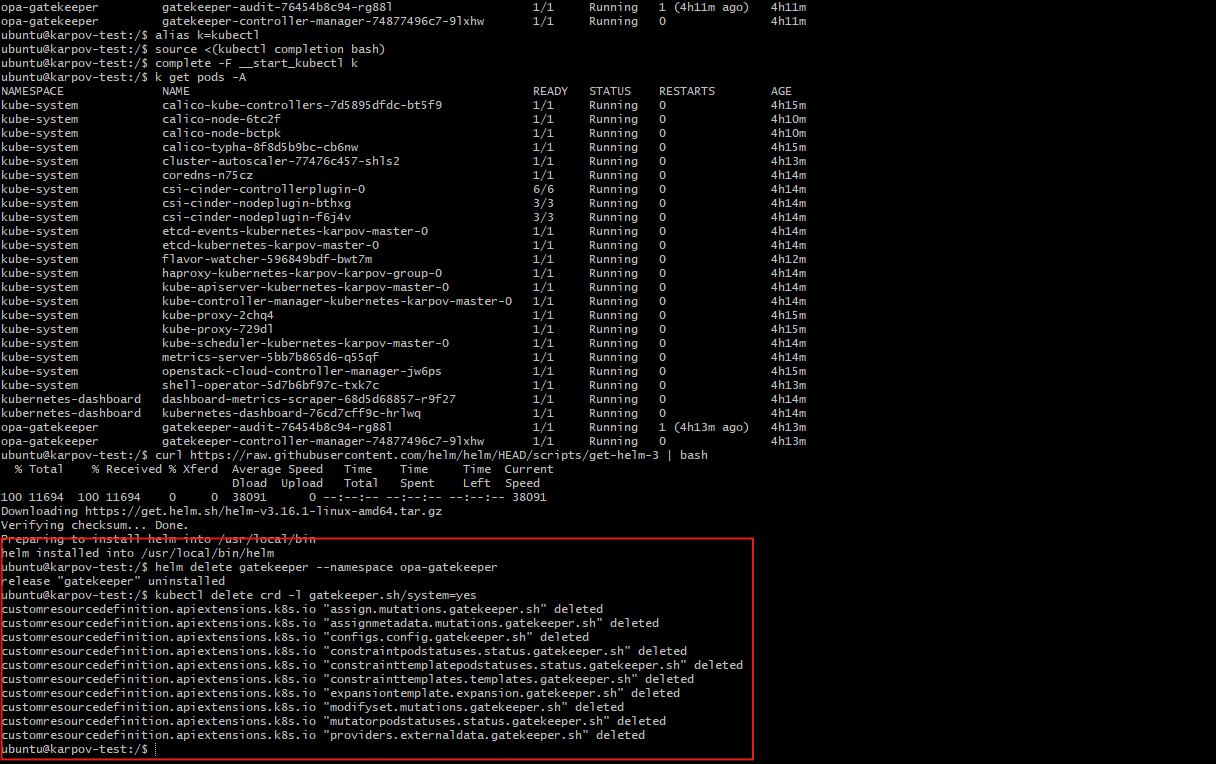


Рисунок 8.

Пропатчим storageclass (см. рисунок 9)

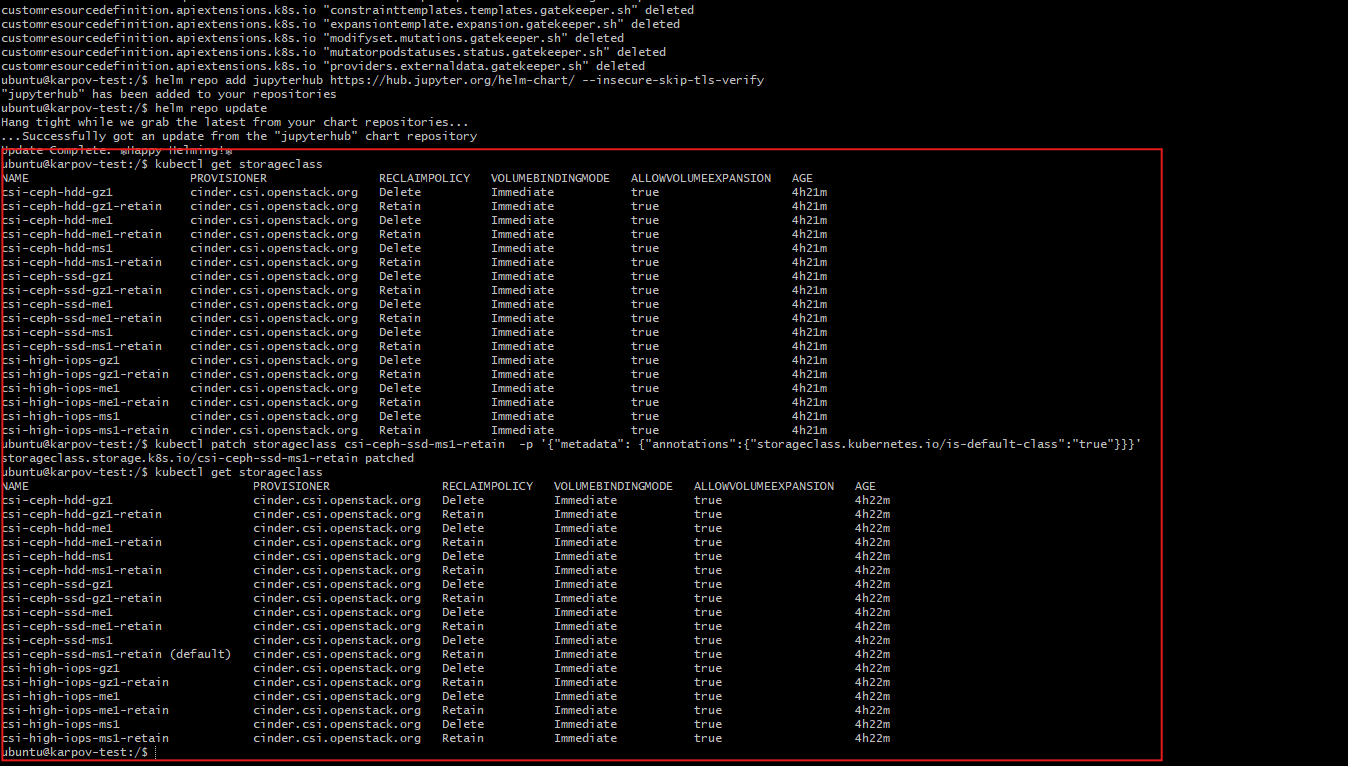


Рисунок 9.

От имени суперпользователя вносим дополнительные параметры (см. рисунок 10).



Рисунок 10.

Как пользоваться nano в Linux: <https://losst.pro/kak-sohranit-fajl-v-nano-linux?ysclid=m13pang26m125350756>.

Устанавливаем JupiterHub 3.0.3 версии (см. рисунок 11).

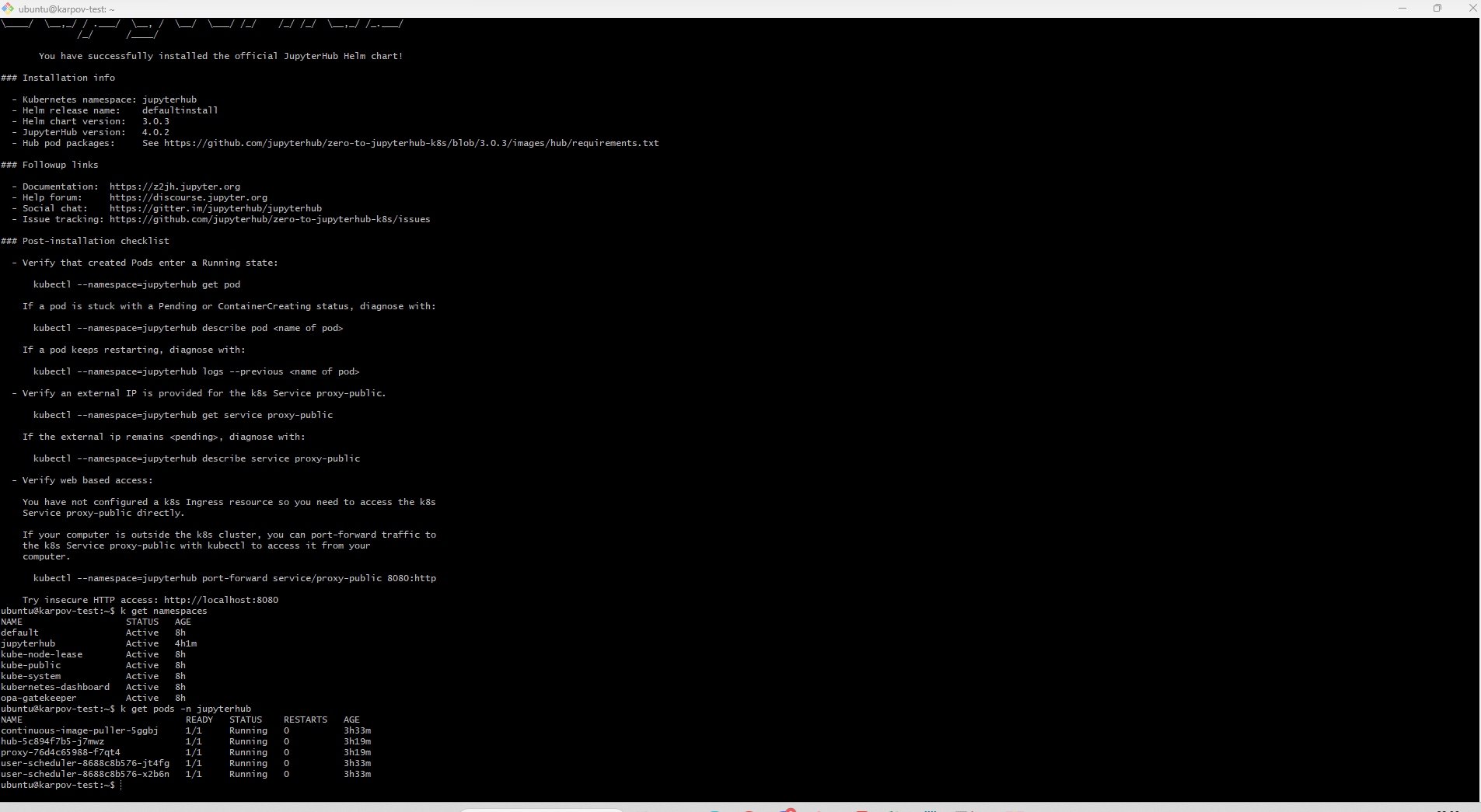


Рисунок 11.

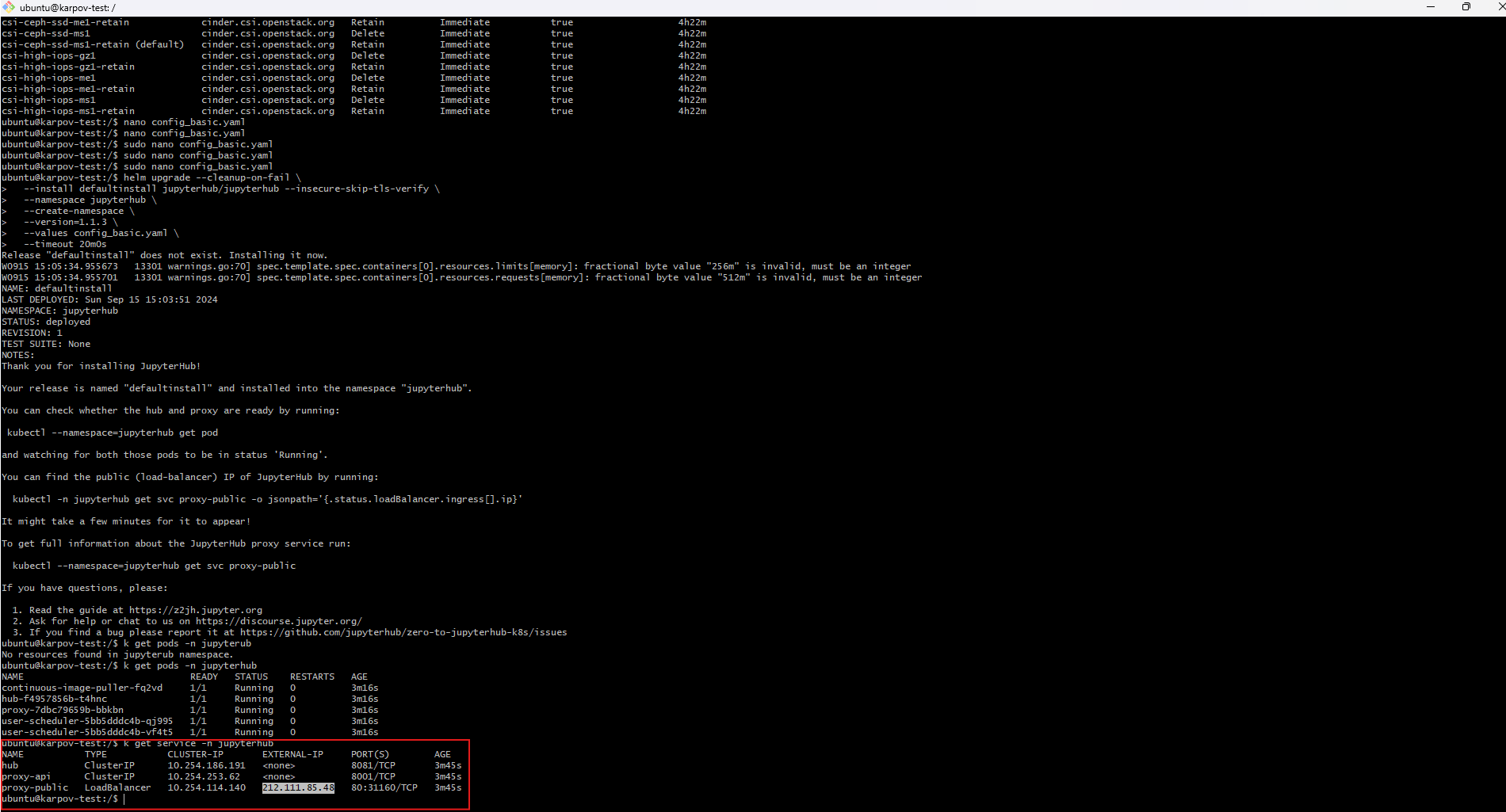


Рисунок 12.

Запускаем (см. рисунок 13)

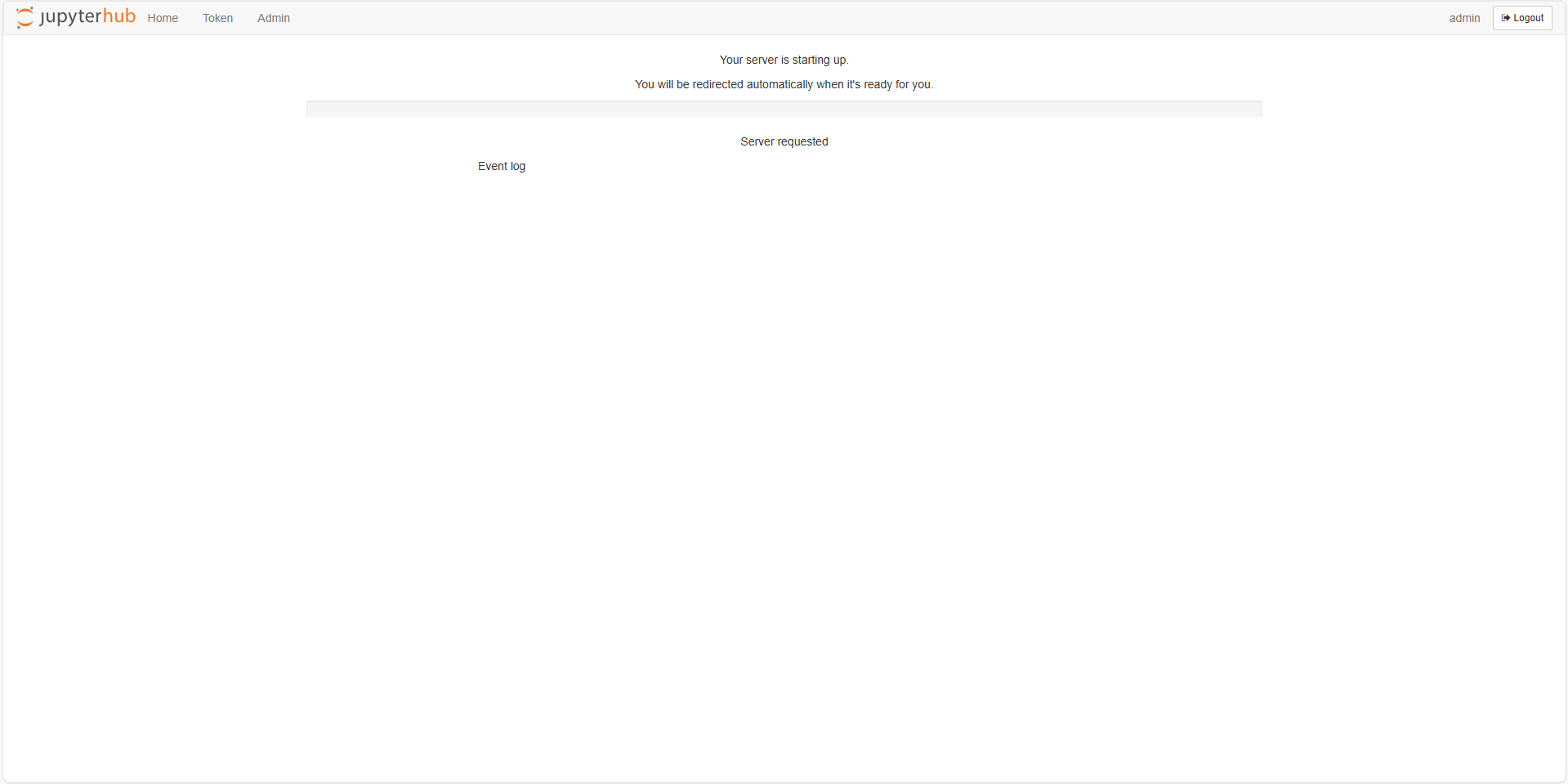


Рисунок 13.

Ждем загрузку (см. рисунок 14)

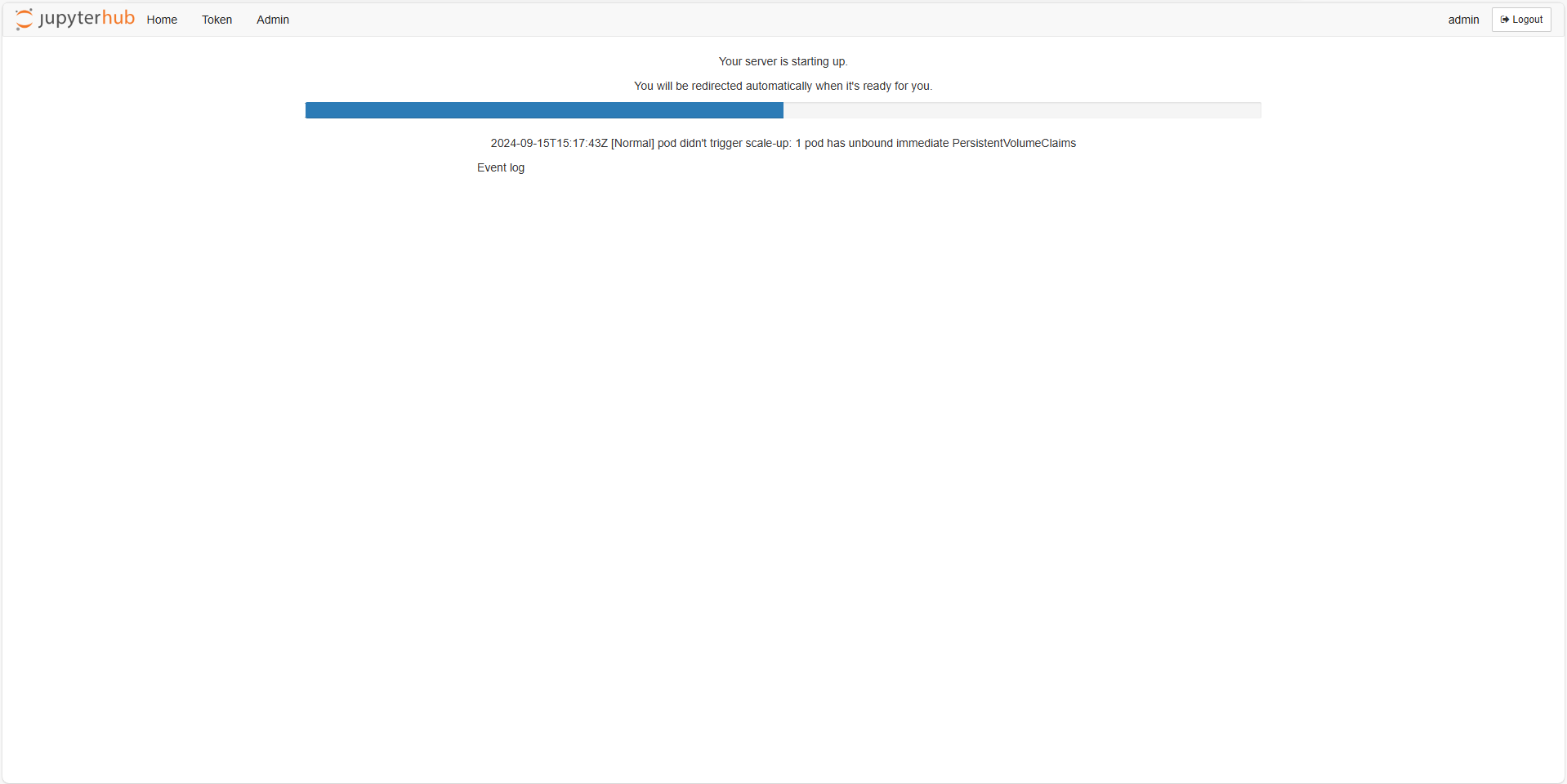


Рисунок 14.

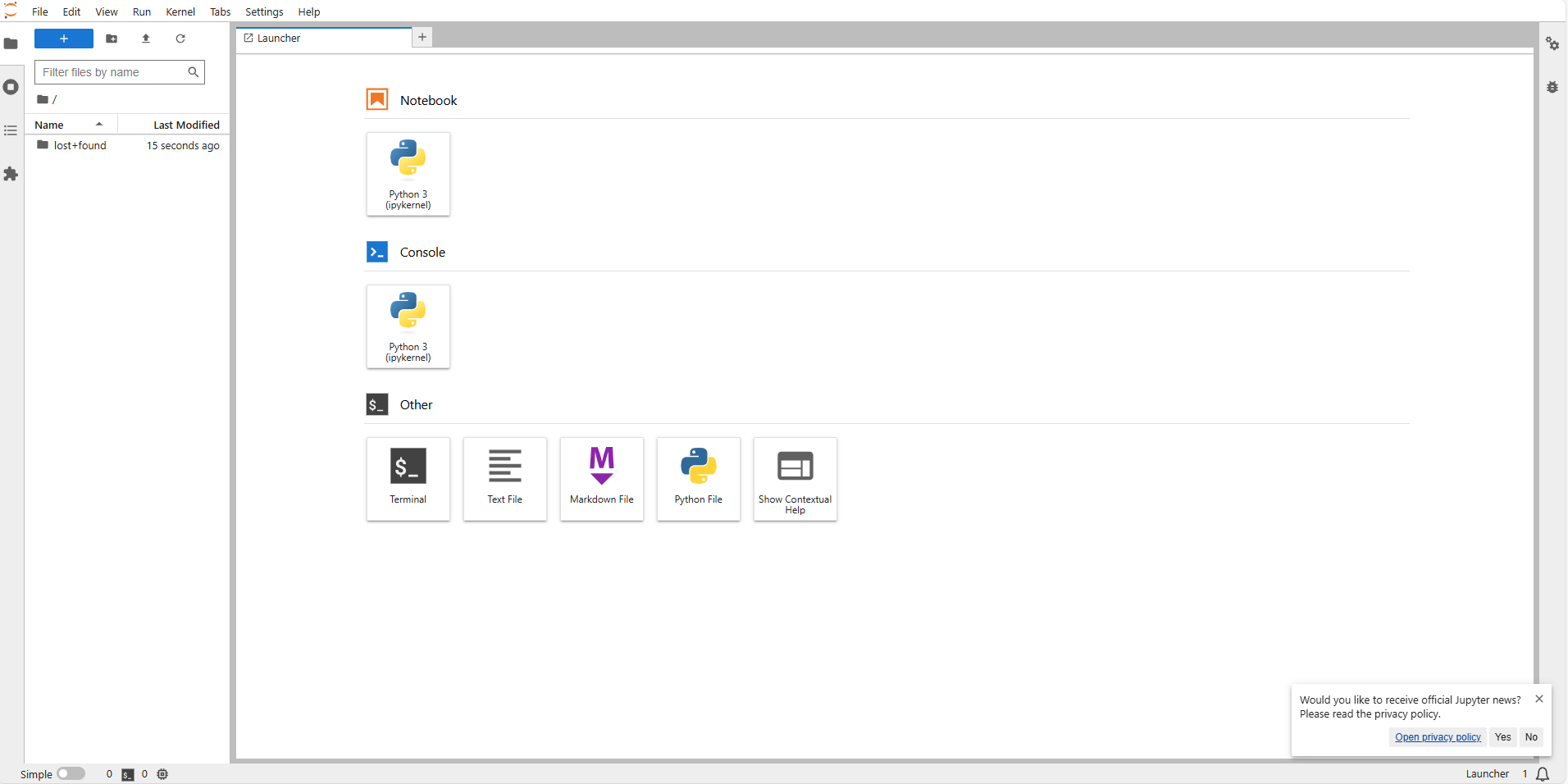


Рисунок 15. Успешный запуск.

Видим, что появился новый POD (см. рисунок 16).

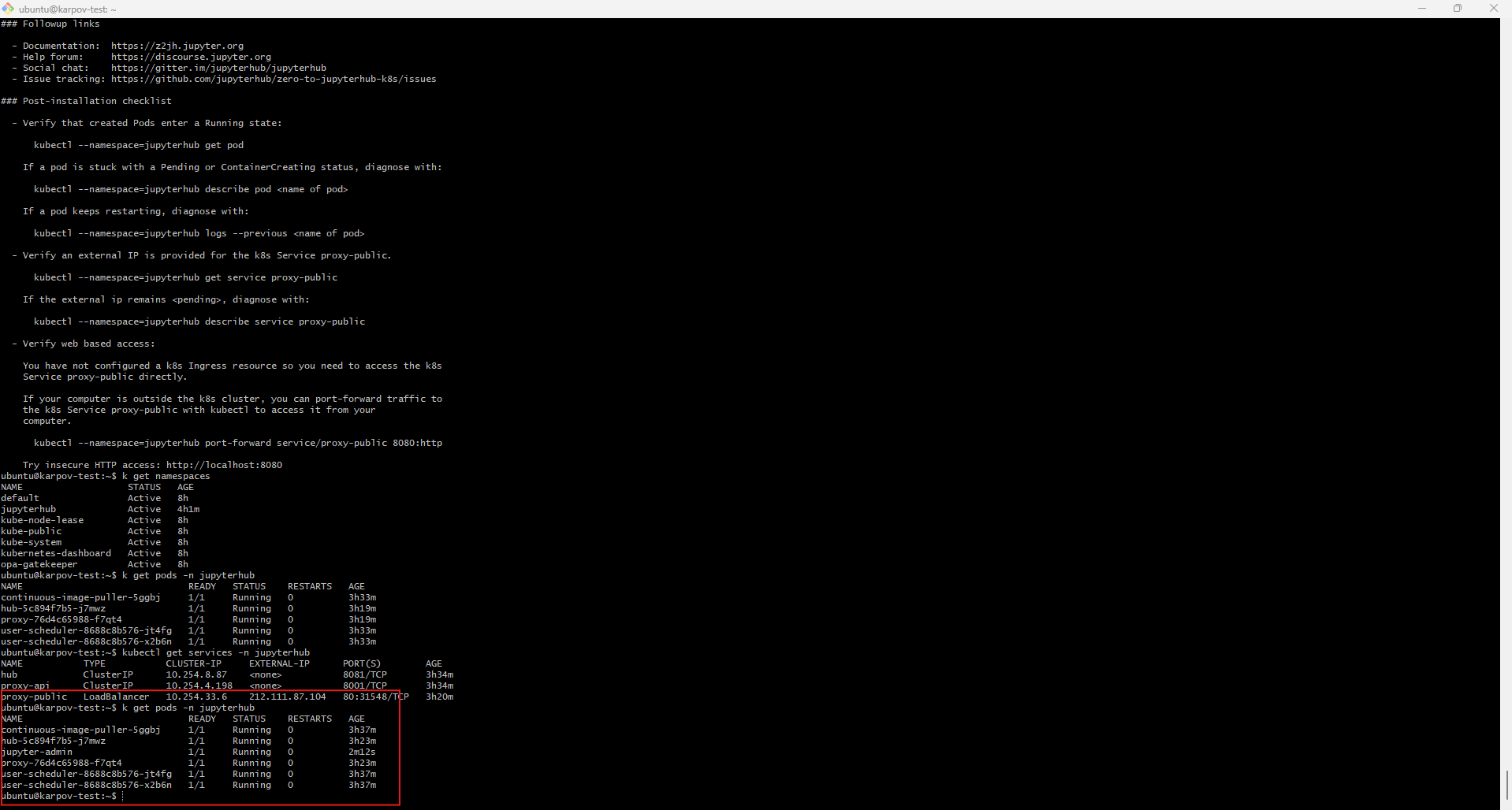


Рисунок 16.

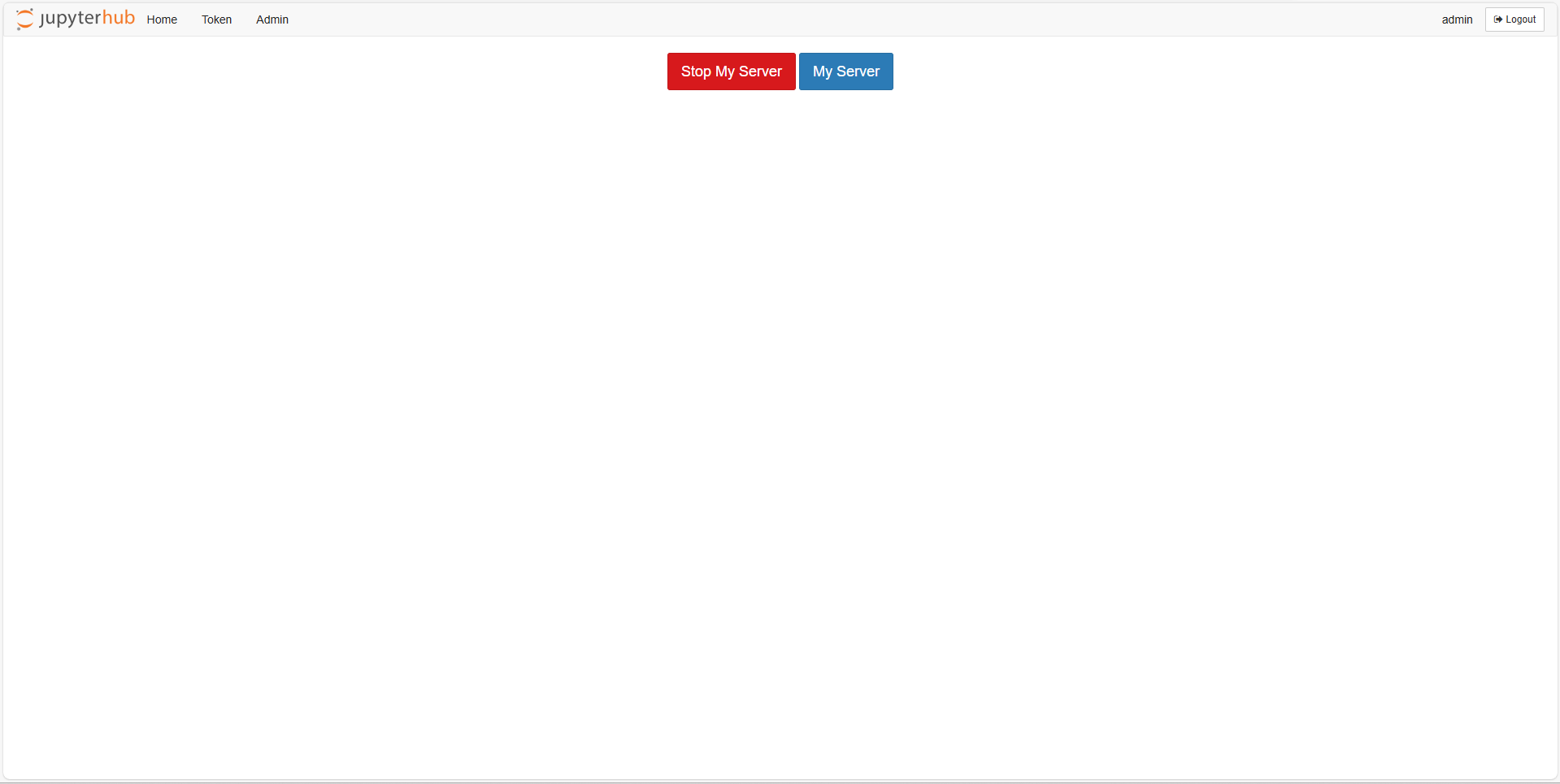


Рисунок 17.

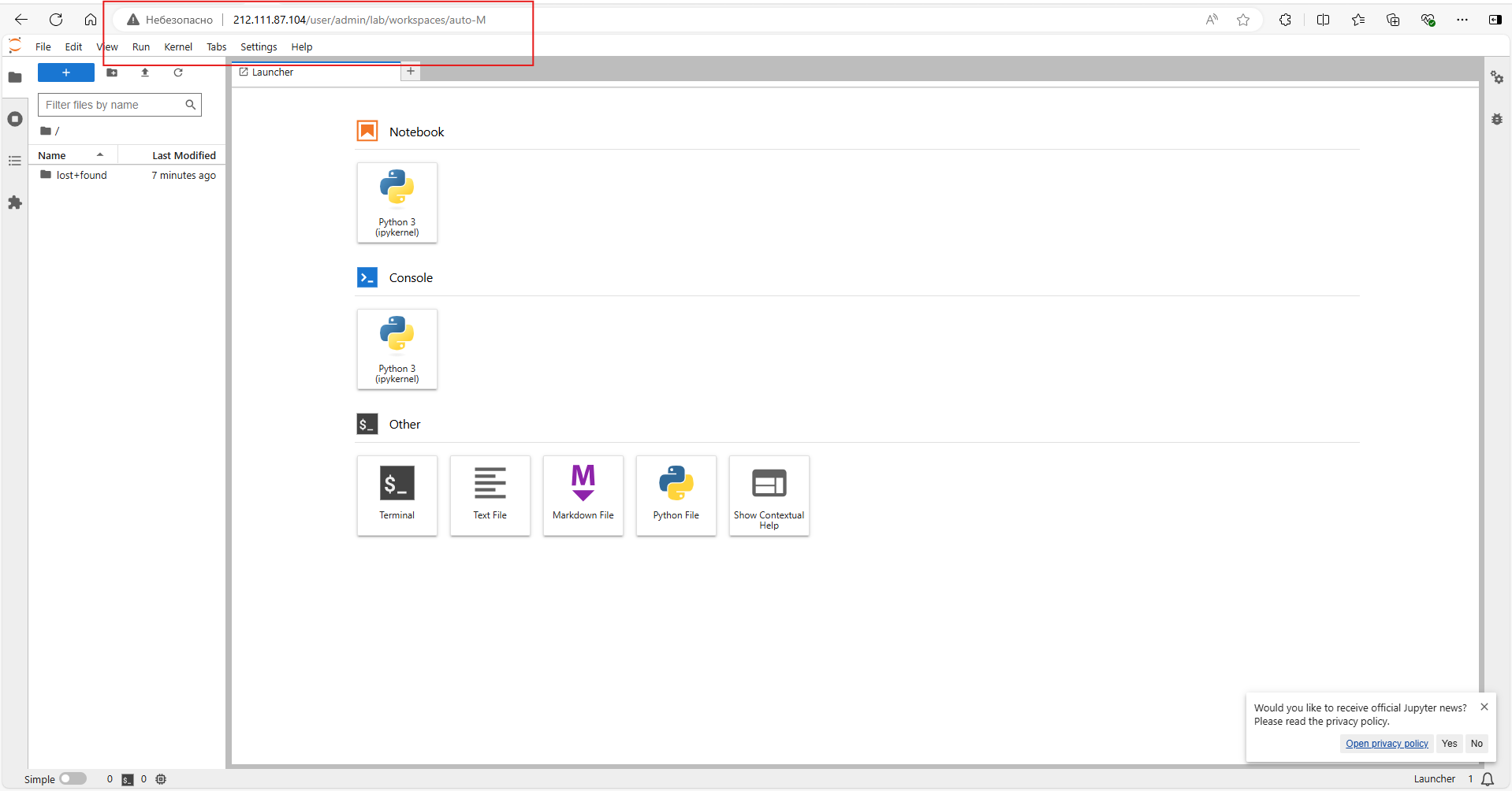


Рисунок 18.