**Руководство по установке Primo RPA Orchestrator на Ubuntu Server 22.04**

**(при наличии интернета на сервере)**

**Обновляем список пакетов и систему**

# sudo apt update

# sudo apt upgrade

# sudo reboot

Переходим в режим root:

# sudo -i (вводим пароль) далее все действия будут выполняться под пользователем root

**Установка PostgreSQL 13**

# sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb\_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'

# wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -

# apt update

# apt install postgresql-13 -y

# systemctl status postgresql

Разрешаем подключение к PostgreSQL по сети

# vim /etc/postgresql/13/main/postgresql.conf

Находим строку:

#listen\_addresses = 'localhost'

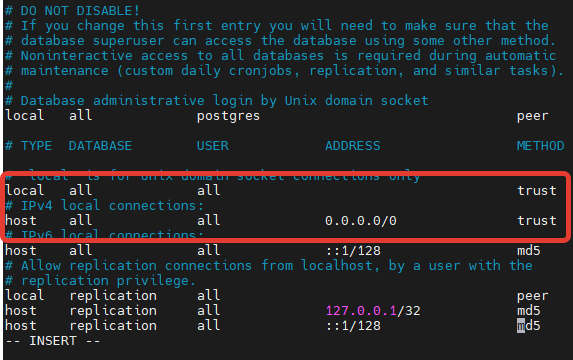
и вносим следующие изменения:

listen\_addresses = '\*'

**Далее открываем файл pg\_hba.conf**

# vim /etc/postgresql/13/main/pg\_hba.conf

Ищем строки и вносим туда следующие изменения:



# systemctl restart postgresql

Далее

# sudo -i -u postgres psql -U postgres -c 'create database ltoolslicense;'

# sudo -i -u postgres psql -c 'grant all privileges on database ltoolslicense to postgres;'

# sudo -i -u postgres psql -c "alter user postgres with password 'postgres'"

# systemctl restart postgresql

# systemctl status postgresql

**Установка RabbitMQ Server**

# sudo apt install curl gnupg apt-transport-https -y

# curl -1sLf "https://keys.openpgp.org/vks/v1/by-fingerprint/0A9AF2115F4687BD29803A206B73A36E6026DFCA" | sudo gpg --dearmor | sudo tee /usr/share/keyrings/com.rabbitmq.team.gpg > /dev/null

# curl -1sLf https://ppa1.novemberain.com/gpg.E495BB49CC4BBE5B.key | sudo gpg --dearmor | sudo tee /usr/share/keyrings/rabbitmq.E495BB49CC4BBE5B.gpg > /dev/null

# curl -1sLf https://ppa1.novemberain.com/gpg.9F4587F226208342.key | sudo gpg --dearmor | sudo tee /usr/share/keyrings/rabbitmq.9F4587F226208342.gpg > /dev/null

Команду ниже выполняем полностью скопировав текст подсвеченный зеленым полем:

sudo tee /etc/apt/sources.list.d/rabbitmq.list <<EOF

## Provides modern Erlang/OTP releases

##

deb [signed-by=/usr/share/keyrings/rabbitmq.E495BB49CC4BBE5B.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-erlang/deb/ubuntu jammy main

deb-src [signed-by=/usr/share/keyrings/rabbitmq.E495BB49CC4BBE5B.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-erlang/deb/ubuntu jammy main

## Provides RabbitMQ

##

deb [signed-by=/usr/share/keyrings/rabbitmq.9F4587F226208342.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-server/deb/ubuntu jammy main

deb-src [signed-by=/usr/share/keyrings/rabbitmq.9F4587F226208342.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-server/deb/ubuntu jammy main

EOF

# sudo apt update -y

# sudo apt install -y erlang-base \

erlang-asn1 erlang-crypto erlang-eldap erlang-ftp erlang-inets \

erlang-mnesia erlang-os-mon erlang-parsetools erlang-public-key \

erlang-runtime-tools erlang-snmp erlang-ssl \

erlang-syntax-tools erlang-tftp erlang-tools erlang-xmerl

# sudo apt install rabbitmq-server -y --fix-missing

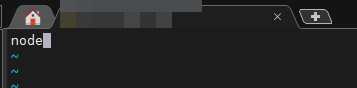
Далее смотрим имя нашего сервера:

# hostname

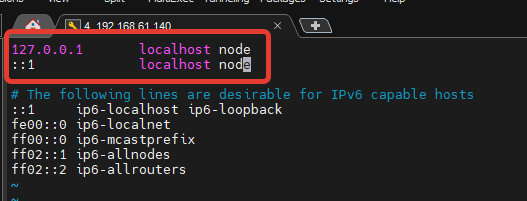


# vim /etc/hostname

Прописываем его в файл:



# vim /etc/hosts



# reboot

После перезагрузки от root:

# adminPassword="Qwe123"'!'"@#"

# systemctl enable rabbitmq-server

# rabbitmq-plugins enable rabbitmq\_management

# rabbitmqctl add\_user 'admin' ${adminPassword}

# rabbitmqctl set\_user\_tags admin administrator

# rabbitmqctl set\_permissions -p / admin '.\*' '.\*' '.\*'

# systemctl restart rabbitmq-server

Проверяем статус:

# systemctl status rabbitmq-server

**Установка NGINX**

# apt install nginx -y

Далее нам понадобится архив **nginx-linux.zip** из \Primo Orchestrator 1.23.5.0\Distr\Linux

Распаковываем его в папку:

# unzip nginx-linux.zip -d /tmp/nginx/

И копируем содержимое:

# cp /tmp/nginx/\* /etc/nginx

# systemctl restart nginx

Проверяем статус:

# systemctl status nginx

**Установка .NET Core 7**

# unzip dotnet7-ubuntu.zip -d /tmp/

# dpkg -i /tmp/dotnet7/\*.deb

Смотрим версию dotnet

# dotnet --info

**Установка WebApi**

# mkdir -p /opt/Primo/WebApi

# unzip WebApi-linux.zip -d /opt/Primo/WebApi/

# cp /opt/Primo/WebApi/Primo.Orchestrator.WebApi.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.WebApi.service

# chmod -R +x /opt/Primo/WebApi/Primo.Orchestrator.WebApi

# vim /opt/Primo/WebApi/appsettings.ProdLinux.json

Задаем переменную окружения:

# export ASPNETCORE\_ENVIRONMENT=ProdLinux

# systemctl start Primo.Orchestrator.WebApi

# systemctl status Primo.Orchestrator.WebApi

**Установка States**

# mkdir -p /opt/Primo/States

# unzip States-linux.zip -d /opt/Primo/States/

# cp /opt/Primo/States/Primo.Orchestrator.States.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.States.service

# chmod -R +x /opt/Primo/States/Primo.Orchestrator.States

# vim /opt/Primo/States/appsettings.ProdLinux.json

# systemctl start Primo.Orchestrator.States

# systemctl status Primo.Orchestrator.States

**Установка Notifications**

# mkdir -p /opt/Primo/Notifications

# unzip Notifications-linux.zip -d /opt/Primo/Notifications/

# cp /opt/Primo/Notifications/Primo.Orchestrator.Notifications.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.Notifications.service

# chmod -R +x /opt/Primo/Notifications/Primo.Orchestrator.Notifications

# vim /opt/Primo/Notifications/appsettings.ProdLinux.json

# systemctl start Primo.Orchestrator.Notifications

# systemctl status Primo.Orchestrator.Notifications

**Установка MachineInfo**

# mkdir -p /opt/Primo/MachineInfo

# unzip MachineInfo-linux.zip -d /opt/Primo/MachineInfo/

# cp /opt/Primo/MachineInfo/Primo.Orchestrator.MachineInfo.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.MachineInfo.service

# chmod -R +x /opt/Primo/MachineInfo/Primo.Orchestrator.MachineInfo

# vim /opt/Primo/MachineInfo/appsettings.ProdLinux.json

# systemctl start Primo.Orchestrator.MachineInfo

# systemctl status Primo.Orchestrator.MachineInfo

**Установка RobotLogs**

# mkdir -p /opt/Primo/RobotLogs

# unzip RobotLogs-linux.zip -d /opt/Primo/RobotLogs/

# cp /opt/Primo/RobotLogs/Primo.Orchestrator.RobotLogs.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.RobotLogs.service

# chmod -R +x /opt/Primo/RobotLogs/Primo.Orchestrator.RobotLogs

# vim /opt/Primo/RobotLogs/appsettings.ProdLinux.json

# systemctl start Primo.Orchestrator.RobotLogs

# systemctl status Primo.Orchestrator.RobotLogs

**Установка веб-интерфейса**

# unzip UI.zip -d /opt/Primo/UI

Переходим по адресу:

<https://ipaddr:44392>

**Установка RDP2**

# mkdir -p /opt/Primo/RDP2

# unzip RDP2-Astra.zip -d /opt/Primo/RDP2/

# cp /opt/Primo/RDP2/Primo.Orchestrator.RDP2.service /etc/systemd/system/

# systemctl daemon-reload

# systemctl enable /etc/systemd/system/Primo.Orchestrator.RDP2.service

# chmod -R +x /opt/Primo/RDP2/Primo.Orchestrator.RDP2

# vim /opt/Primo/RDP2/appsettings.ProdLinux.json

# systemctl start Primo.Orchestrator.RDP2

# systemctl status Primo.Orchestrator.RDP2