

Проекты VK WorkSpace

Кластер RabbitMQ

Оглавление

Шаг 1. Подготовка нод (серверов)	3
Шаг 2. Установка RabbitMQ	3
Шаг 3. Подготовка кластера RabbitMQ	4
Шаг 4. Настройка RabbitMQ	5
Шаг 5. Конфигурация HAProxy (блок RabbitMQ)	6
Шаг 6. Подключение к RabbitMQ	6

Для работы приложения необходимо установить RabbitMQ версии 3.12.

В статье описана установка RabbitMQ и Erlang 25.3.2.2-1 для ОС Ubuntu Linux 20.04 и 22.04. Вы можете ознакомиться с руководством в [официальной документации RabbitMQ](#).

Установка состоит из шести этапов:

1. Подготовка нод (серверов)
2. Установка RabbitMQ
3. Подготовка кластера RabbitMQ
4. Настройка RabbitMQ
5. Конфигурация HAProxy (блок rabbitmq).
6. Подключение к RabbitMQ

Шаг 1. Подготовка нод (серверов)

Минимальное количество серверов для организации кластера — три.

1. Создайте три ноды (сервера) с последовательно пронумерованными именами хостов:

- rabbitmq-server1.your_domain
- rabbitmq-server2.your_domain
- rabbitmq-server3.your_domain

2. Создайте необходимые сопоставления имён хостов в DNS. Если такой возможности нет, внесите нужные записи в `/etc/hosts`.

Шаг 2. Установка RabbitMQ

1. Установите необходимые пакеты:

```
sudo apt-get install curl gnupg apt-transport-https -y
```

2. Импортируйте все необходимые ключи:

```
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
```

Актуальные ключи для разных версий RabbitMQ перечислены в [официальной документации RabbitMQ](#).

3. Добавьте репозитории RabbitMQ:

```
sudo tee /etc/apt/sources.list.d/rabbitmq.list <<EOF
deb [signed-by=/usr/share/keyrings/rabbitmq.E495BB49CC4BBE5B.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-erlang/deb/ubuntu $(lsb_release -cs) main
deb-src [signed-by=/usr/share/keyrings/rabbitmq.E495BB49CC4BBE5B.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-erlang/deb/ubuntu $(lsb_release -cs) main
deb [signed-by=/usr/share/keyrings/rabbitmq.9F4587F226208342.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-server/deb/ubuntu $(lsb_release -cs) main
deb-src [signed-by=/usr/share/keyrings/rabbitmq.9F4587F226208342.gpg] https://ppa1.novemberain.com/rabbitmq/rabbitmq-server/deb/ubuntu $(lsb_release -cs) main
EOF
```

4. Обновите кеш пакетов:

```
sudo apt-get update -y
```

5. Установите пакеты Erlang:

```
sudo apt-get install -y erlang-base=1:25.3.2.2-1 erlang-asn1=1:25.3.2.2-1 erlang-crypto=1:25.3.2.2-1 erlang-eldap=1:25.3.2.2-1 erlang-ftp=1:25.3.2.2-1 erlang-inets=1:25.3.2.2-1 erlang-mnesia=1:25.3.2.2-1 erlang-os-mon=1:25.3.2.2-1 erlang-parsetools=1:25.3.2.2-1 erlang-public-key=1:25.3.2.2-1 erlang-runtime-tools=1:25.3.2.2-1 erlang-snmp=1:25.3.2.2-1 erlang-ssl=1:25.3.2.2-1 erlang-syntax-tools=1:25.3.2.2-1 erlang-tftp=1:25.3.2.2-1 erlang-tools=1:25.3.2.2-1 erlang-xmerl=1:25.3.2.2-1
```

6. Установите rabbitmq-server и его зависимости:

```
sudo apt-get install rabbitmq-server=3.12.0-1 -y --fix-missing
```

7. Запустите rabbitmq-server:

```
sudo systemctl enable --now rabbitmq-server
```

Шаг 3. Подготовка кластера RabbitMQ

1. На каждой ноде rabbitmq создайте файл `/etc/rabbitmq/rabbitmq-env.conf` и добавьте переменные окружения `RABBITMQ_NODENAME` и `RABBITMQ_USE_LONGNAME`

Пример заполнения для `rabbitmq-server1.your_domain`:

```
RABBITMQ_NODENAME=rabbit@rabbitmq-server1.your_domain
RABBITMQ_USE_LONGNAME=true
```

Пример заполнения для `rabbitmq-server2.your_domain`:

```
RABBITMQ_NODENAME=rabbit@rabbitmq-server2.your_domain
RABBITMQ_USE_LONGNAME=true
```

Пример заполнения для `rabbitmq-server3.your_domain`:

```
RABBITMQ_NODENAME=rabbit@rabbitmq-server3.your_domain
RABBITMQ_USE_LONGNAME=true
```

2. Скопируйте Cookie `/var/lib/rabbitmq/.erlang.cookie` с первого узла `rabbitmq-server1.your_domain` на все остальные узлы в кластере.

Для работы кластера RabbitMQ у всех узлов, участвующих в кластере, должно совпадать содержимое файла `/var/lib/rabbitmq/.erlang.cookie`.

3. Перезапустите сервис RabbitMQ на каждой ноде:

```
sudo systemctl restart rabbitmq-server
```

4. Остановите приложение на узлах `rabbitmq-server2.your_domain` и `rabbitmq-server3.your_domain` для последующего присоединения к кластеру:

```
sudo rabbitmqctl stop_app
```

5. Сбросьте rabbitmq на узлах `rabbitmq-server2.your_domain` и `rabbitmq-server3.your_domain`:

```
sudo rabbitmqctl reset
```

6. Присоедините узлы `rabbitmq-server2.your_domain` и `rabbitmq-server3.your_domain` к кластеру:

```
sudo rabbitmqctl join_cluster rabbit@rabbitmq-server1.your_domain
```

7. Запустите приложение на узлах `rabbitmq-server2.your_domain` и `rabbitmq-server3.your_domain`:

```
sudo rabbitmqctl start_app
```

8. Проверьте статус кластера:

```
sudo rabbitmqctl cluster_status
```

Шаг 4. Настройка RabbitMQ

Для пароля разрешается применять следующие символы:

- заглавные латинские буквы: от A до Z
- строчные латинские буквы: от a до z
- цифры от 0 до 9
- символы: `-_.`

Зарезервированные (недопустимые) символы: `!*'();:@&=+$,/?%#[]`

1. На каждом узле включите необходимые плагины:

```
sudo rabbitmq-plugins enable \  
rabbitmq_management
```

2. Создайте vhost и дайте пользователю доступ к нему, выполнив команды на первом узле `rabbitmq-server1.your_domain`:

```
sudo rabbitmqctl add_vhost {{ shell_product_name }}vhost  
sudo rabbitmqctl add_user {{ shell_product_name }}user SecretPassword  
sudo rabbitmqctl set_permissions -p {{ shell_product_name }}vhost {{ shell_product_name }}  
user ".*" ".*" ".*"  
sudo rabbitmqctl set_user_tags {{ shell_product_name }}user administrator
```

3. Создайте политику, которая позволяет зеркалировать очереди для всех узлов в кластере, выполнив команду на первом узле `rabbitmq-server1.your_domain`:

```
sudo rabbitmqctl set_policy -p '{{ shell_product_name }}host' MirrorAllQueues ".*" '{"ha-mode":"all"}'
```

Если отсутствует поддержка политики High Availability Mode, используйте политику Quorum:

```
rabbitmqctl set_policy -p '{{ shell_product_name }}vhost' QuorumDefault "^.*" '{"queue-type":"quorum"}' --priority 0 --apply-to queues
```

Чтобы посмотреть настроенные политики, используйте команду:

```
sudo rabbitmqctl list_policies --vhost {{ shell_product_name }}vhost
```

Шаг 5. Конфигурация HAProxy (блок RabbitMQ)

Балансировку нагрузки между нодами кластера RabbitMQ осуществим с помощью [HAProxy](#).

Шаг 6. Подключение к RabbitMQ

Строка для подключения к кластеру RabbitMQ (через HAProxy):

```
amqp://{{ shell_product_name }}user:SecretPassword@haproxy-server.your_domain:5672/{{ shell_product_name }}vhost
```

Строка для подключения к кластеру RabbitMQ с TLS/SSL (через HAProxy с TLS/SSL):

```
amqps://{{ shell_product_name }}user:SecretPassword@haproxy-server.your_domain:5671/{{ shell_product_name }}vhost
```

 Технический писатель: Белова Ирина

 6 мая 2026 г.