

Система резервного копирования и восстановления данных «ROC
Backup»

Порядок развертывания Системы

Листов 10

Москва 2024г.

Оглавление

1 Порядок установки	3
1.1.1 Установка Docker Engine	3
1.1.2 Установка ROC backup.....	6
2 Настройка ROC backup	9
3 Рекомендации по миграции данных	10

1 Порядок установки

Для корректной установки системы ROC backup необходимо предварительно получить у поставщика системы логин и пароль для доступа к репозиторию с образами компонент системы. Установка и настройка системы ROC backup состоит из следующих шагов:

1. Выделение DNS адрес и IP адреса для сервера ROC backup.
2. Выпуск ssl сертификата.
3. Выделение сервера для системы бэкапирования.
4. Установка актуальной версии операционной системы (на примере семейства Linux) на сервер согласно документации на данную операционную систему.
5. Выделение и настройка раздела для бэкапов.
6. Отключение контроля перерасхода памяти ядром и перезагрузка сервера командой:

```
echo "vm.overcommit_memory=1" >> /etc/sysctl.conf  
reboot
```

7. Настройка синхронизации времени с глобальными или локальными ntp серверами.
8. Установка docker engine, docker compose (см. п. 1.1.1)
9. Размещение на сервера файлов настройки (конфиг) (ссылка на файле передается по почте):

```
prod-docker-compose.rockbackup.yml, nginx.conf, .env.example
```

10. Настройка на сервере доступа к интернет.
11. Запуск системы (см. п. 1.1.2).
12. Настройка в ROC backup (см. п. 2).

1.1.1 Установка Docker Engine

Порядок установки компонентов Docker Engine и Docker Compose может незначительно отличаться в зависимости от используемого дистрибутива Linux. Ниже приведен порядок действия для ОС **Альт Сервер 11, Astra Linux Common Edition 2.12.29 (Орел), РЕД ОС 7.3, Ubuntu Server 22.04.**

1.1.1.1 Альт Сервер 11

1. Получение списка обновлений и установка Docker Engine и Docker Compose.
Для этого необходимо выполнить команды:

```
sudo apt-get update
sudo apt-get install docker-ce, docker-compose -y.
```

2. Настройка запуска сервиса docker без использования команды sudo. Для этого в группу docker необходимо добавить пользователя, в сеансе которого запускается сервис:

```
sudo usermod -aG docker $USER.
```

3. Применение настройки группы:

- a. Зайти в операционную систему под текущим пользователем:

```
logout.
```

- b. Запустить сервис docker:

```
sudo systemctl enable --now docker.
```

- c. Проверить, что сервис запущен корректно:

```
docker -v
docker ps -a
```

Если установка выполнена успешно, первая команда выведет версию Docker Engine, вторая – пустой список контейнеров.

1.1.1.2 Astra Linux Common Edition 2.12.29 (Орел)

1. Подключение репозитория и установка Docker Engine:

```
apt update && apt upgrade
sudo apt-get install apt-transport-https ca-certificates curl gnupg2
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo apt-key
add -
echo "deb [arch=amd64] https://download.docker.com/linux/debian
stretch stable" | sudo tee -a /etc/apt/sources.list
sudo apt update
sudo apt install docker-ce
sudo systemctl status docker – для проверки, что сервис запущен
корректно.
```

Если сервис запущен, то команда выведет статус сервиса «Active: active (running)».

Нажмите клавиши CTRL+C, чтобы завершить выполнение команды.

2. Проверка установленной версии Docker Engine с помощью команды:

```
docker --version.
```

3. Настройка запуска сервиса docker без использования команды sudo. Для настройки в группу docker необходимо добавить пользователя, в сеансе которого запускается сервис:

```
sudo usermod -aG docker $USER.
```

4. Применение настройки группы. Для применения настроек группы необходимо перезайти в операционную систему под текущим пользователем:

```
logout.
```

1.1.1.3 РЕД ОС 7.3

1. Установка Docker Engine и Docker Compose. Для установки необходимо выполнить команду:

```
yum install docker-ce docker-compose.x86_64 -y.
```

2. Запуск Docker Engine:

```
systemctl start docker
```

3. Настройка запуска сервиса docker без использования команды sudo. Для настройки необходимо в группу docker добавить пользователя, в сеансе которого запускается сервис:

```
sudo usermod -aG docker $USER.
```

4. Применение настройки группы. Для применения настроек группы необходимо перезайти в операционную систему под текущим пользователем:

```
logout.
```

5. Проверка запуска сервиса

```
docker: docker -v  
docker ps -a
```

Если установка выполнена успешно, то первая команда выведет версию Docker Engine, вторая – пустой список контейнеров.

1.1.1.4 Ubuntu Server 22.04

1. Подключение репозитория и установка Docker Engine:

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --  
dearmor -o /etc/apt/trusted.gpg.d/docker-archive-keyring.gpgsudo apt-get  
install apt-transport-https ca-certificates curl gnupg2
```

```

sudo add-apt-repository "deb [arch=amd64]
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"echo
"deb [arch=amd64] https://download.docker.com/linux/debian stretch
stable" | sudo tee -a /etc/apt/sources.list
sudo apt update
sudo apt install docker-ce
sudo apt install docker-compose
sudo systemctl status docker – для проверки, что сервис запущен
корректно.

```

Если сервис запущен, то команда выведет статус сервиса «Active: active (running)».

Нажмите клавиши CTRL+C, чтобы завершить выполнение команды.

2. Проверка установленной версии Docker Engine с помощью команды:

```
docker --version.
```

3. Настройка запуска сервиса docker без использования команды sudo. Для настройки в группу docker необходимо добавить пользователя, в сеансе которого запускается сервис:

```
sudo usermod -aG docker $USER.
```

4. Применение настройки группы. Для применения настроек группы необходимо перезайти в операционную систему под текущим пользователем:

```
logout.
```

1.1.2 Установка ROC backup

1. Создать на сервере каталоги:

```
mkdir -p /opt/robackup/data /opt/robackup/conf
```

2. Скопировать файлы nginx.conf, prod-docker-compose.robackup.yml, .env.example из дистрибутива в каталог /opt/robackup

3. Перейти в каталог /opt/robackup: cd /opt/robackup

4. Скопировать файл .env.example в .env

cp .env.example в .env

5. Изменить значения переменных в конфигурационном файле .env, где:

№	Ключ	Описание
1.	<i>SECRET_KEY</i>	Секретный ключ, используемый Django для криптографической подписи.

№	Ключ	Описание
2.	<i>ALLOWED_HOSTS</i>	Список хостов/доменных имен, которые Django может обрабатывать. Например: ALLOWED_HOSTS=127.0.0.1,localhost,172.16.1.37
3.	<i>CSRF_TRUSTED_ORIGINS</i>	Список доверенных источников для небезопасных запросов (например, POST).
4.	<i>HOST_URL</i>	URL, на котором будет опубликовано решение.
5.	<i>DJANGO_SUPERUSER_PASSWORD</i>	Пароль супер-пользователя в goshbackup.
6.	<i>UI_PORT</i>	Порт, на котором необходимо опубликовать решение.
7.	<i>COMPANY_NAME</i>	Название компании для использования в качестве дерева орг. структуры.
8.	<i>CERTIFICATE</i>	Путь до сертификата публичного ключа для проверки лицензии.
9.	<i>KEY</i>	Путь до сертификата ключа для формирования лицензии.
10.	<i>KEY_PASSWORD</i>	Пароль к ключу для формирования лицензии.
11.	<i>ADMIN_EMAIL</i>	Почтовый ящик администратора, с которого будут отправляться сообщения.
12.	<i>ORGANIZATION_CLIENT_ID</i>	Id (client_id) приложения с доступом к чтению электронной почты сотрудников организации.
13.	<i>ORGANIZATION_CLIENT_SECRET</i>	Секретный ключ (client_secret) приложения с доступом к чтению электронной почты сотрудников организации.
14.	<i>ORG_ID</i>	Идентификатор организации в Яндекс 360.
15.	<i>POSTGRES_DB</i>	Имя БД, используемой в docker контейнере.
16.	<i>POSTGRES_USER</i>	Пользователь БД используемой в docker контейнере.
17.	<i>POSTGRES_PASSWORD</i>	Пароль к БД, используемой в docker контейнере.
18.	<i>POSTGRES_PORT</i>	Порт БД, используемой в docker контейнере.
19.	<i>POSTGRES_HOST</i>	Имя хоста БД, используемой в docker контейнере.
20.	<i>REDIS_HOST</i>	Имя хоста redis, который используется в качестве: очереди сообщений для celery и сервера кеширования.
21.	<i>REDIS_PORT</i>	Порт redis.
22.	<i>USE_S3</i>	Использовать S3 для хранения media файлов. Имеет преимущество над USE_MINIO.

№	Ключ	Описание
23.	<i>AWS_ACCESS_KEY_ID</i>	Уникальный идентификатор для доступа к API хранилища.
24.	<i>AWS_SECRET_ACCESS_KEY</i>	Секретный ключ для доступа к API хранилища.
25.	<i>AWS_STORAGE_BUCKET_NAME</i>	Имя S3 bucket.
26.	<i>AWS_S3_ENDPOINT_URL</i>	URL для обращения к S3.
27.	<i>USE_MINIO</i>	Использовать MINIO для хранения media файлов.
28.	<i>MINIO_STORAGE_ENDPOINT</i>	URL Minio.
29.	<i>MINIO_STORAGE_ACCESS_KEY=MINIO_ROOT_USER</i>	Уникальный идентификатор для доступа к API хранилища.
30.	<i>MINIO_STORAGE_SECRET_KEY</i>	Секретный ключ для доступа к API хранилища.
31.	<i>STATIC_ROOT</i>	Абсолютный путь к директории со статическими файлами. Если переменная не настроена, то Django использует значение: <абсолютный путь к проекту>/static/
32.	<i>STATIC_URL</i>	URL для обращения к статическим файлам.
33.	<i>MEDIA_ROOT</i>	Абсолютный путь к директории, в которой хранятся файлы. Если переменная не настроена, то Django использует значение: <абсолютный путь к проекту>/media/
34.	<i>MEDIA_URL</i>	URL для обращения к файлам.

6. Авторизоваться в репозитории с образами компонент системы:

```
docker login -u USERNAME repo.croc.ru
```

Здесь USERNAME - логин, переданный поставщиком системы для доступа к репозиторию.

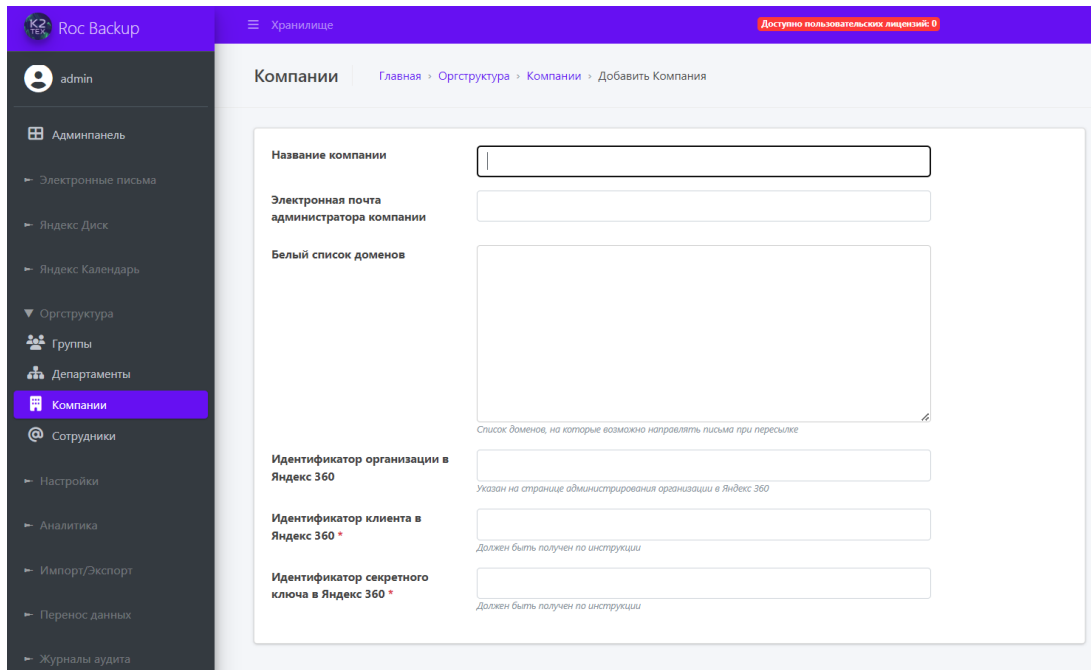
7. Запуск решения:

```
docker compose -f prod-docker-compose.rocbackup.yml up -d
```

8. Проверка запуска 5-ти контейнеров: `docker ps`

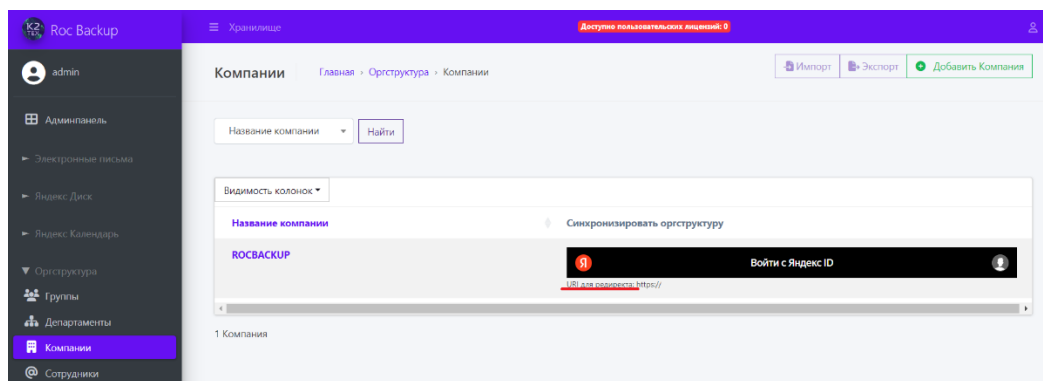
2 Настройка ROC backup

1. Выполнить работы по настройке сервисного приложения в Яндекс: <https://yandex.ru/support/business/security-service-applications.html>
2. Выполнить работы по настройке компании и синхронизация оргструктуры в ROC backup:
 - В меню перейти в раздел: Главная/Оргструктура/Компании
 - Добавить компанию последовательно заполняя каждое поле.



Уточнение:

- Электронная почта администратора компании – почта, через которую вы заходите как администратор в Яндекс.
- Идентификатор секретного ключа в Яндекс 360 – указывается Client secret вида 1f4e12d8fb7842ce5f2bcc33cdaa0182. Сам токен указывать не нужно.
- Сформированный URL внести в Яндекс под учетной записью администратора.



- Запустить синхронизацию оргструктуры, нажав на черную кнопку (скрин выше).
- Далее проверить, что перенесли все сотрудники.

3 Рекомендации по миграции данных

Скорость миграции данных зависит от пропускной способности канала связи, от объема данных, подлежащих миграции, а также от размера мигрируемых файлов.

Рекомендуемая пропускная способность не должна быть ниже 70 Мб/с.

Миграцию данных мы рекомендуем проводить в ночное время из-за снижения нагрузки на выделенный канал связи.

Мы рекомендуем проводить миграцию данных в два этапа:

1. Этап 1:
 - a. Выбирается 50-100 случайных пользователей и для них выполняется миграция данных в ночное время.
 - b. Выбирается 50-100 случайных пользователей и для них выполняется миграция данных в дневное время.
 - c. Снимается статистика скорости миграции.
2. Этап 2:
 - a. На основании статистики из Этапа 1 формируется план-график миграции данных.
 - b. План-график согласовывается с Заказчиком.
9. На основании план-графика запускается миграция данных.