

Инструкция по установке Middle архитектуры в ручном режиме

Содержание

Инструкция по установке Middle архитектуры в ручном режиме	1
Содержание	2
0. Описание архитектуры	5
0.1 Front-end / back-end приложения	5
0.2 Система Управления Базами Данных	6
0.3 Документ сервер	6
1. Установка CS	7
1.1 Скачиваем скрипт	7
1.2 Выполняем команды	7
1.2.1 Обновление пакетов	7
1.2.2 Удаление пакетов, которые мешают инсталляции	7
1.2.3 Проверяем и, при наличии, удаляем конфигурационный файл	7
1.2.4 Запускаем установку	7
1.3 Когда установка завершится	8
2. Установка DS	9
2.0 Обновляем пакеты	9
2.1 Создаём репозиторий nginx и устанавливаем	9
2.1.1 Открываем редактором	9
2.1.2 Вставляем данные ниже	9
2.1.3 Устанавливаем nginx командой	9
2.1.4 Редактируем основной конфигурационный файл	10
2.2 Установка и настройка PostgreSQL	10
2.2.1 Установите пакет	10
2.2.2 Инициализируйте базу данных PostgreSQL и добавьте в автозагрузку	10
2.2.3 Включите метод аутентификации 'trust'	11
2.2.3.1 Откройте файл в текстовом редакторе	11
2.2.3.2 Найдите строку host all all 127.0.0.1/32 ident и замените ее на следующую	11
2.2.3.3 Затем найдите строку host all all ::1/128 ident и замените ее на следующую	11
2.2.4 Перезапустите сервис PostgreSQL	11
2.2.5 Создайте базу данных и пользователя PostgreSQL	11
2.2.6 Установка Redis	11
2.2.7 Установка RabbitMQ	11
2.3 Установка и настройка Сервера Документов	12
2.3.1 Добавьте репозиторий	12
2.3.1.1 Откройте редактором файл	12
2.3.1.2 Добавьте данные ниже	12
2.3.1.3 Устанавливаем пакет DS командой	12

2.3.2	Запустите зависимые сервисы и добавьте их в автозагрузку	12
2.3.3	Запустите скрипт настройки	12
2.3.3.1	Параметры подключения к PostgreSQL, Redis и RabbitMQ	12
2.3.3.1.1	Для PostgreSQL	12
2.3.3.1.2	Для Redis	12
2.3.3.1.3	Для AMQP	13
2.5	Если установлен и работает firewalld , добавьте исключения	13
2.6	Отключите SELINUX	13
2.7	Перезагрузите сервер	13
2.8	Добавьте лицензию	13
2.9	Проверка работы:	13
3.	Установка и настройка MySQL	14
3.1	Обновляем пакеты	14
3.2	Удаляем пакеты	14
3.3	Добавляем репозиторий	14
3.4	Устанавливаем MySQL	14
3.5	Запускаем MySQL	14
3.6	Узнаём пароль от root MySQL командой	14
3.7	Предварительная настройка для каждой ноды MySQL	14
3.7.1	Подключаемся к СУБД	14
3.7.2	Обновляем пароль	15
3.8	Настройка Master	15
3.8.1	Обновляем конфигурационный файл my.cnf	15
3.8.2	Обновляем конфигурационный файл mysqld.service	16
3.8.3	Перезапускаем сервис	16
3.9	Настраиваем Slave	17
3.9.1	Обновляем конфигурационный файл my.cnf	17
3.9.2	Обновляем конфигурационный файл mysqld.service	18
3.9.3	Перезапускаем сервис	18
3.10	Подключение SLAVE к MASTER	19
3.10.1	Создаём учётку для подключения на мастере	19
3.10.2	Получаем значения master_log и master_position	19
3.10.2.1	Выполняем команду на мастере:	19
3.10.2.2	Сохраняем значения из столбцов File и Position	19
3.10.3	Подключаемся на slave и выполняем команды	20
3.10.3.1	Останавливаем slave	20
3.10.3.2	Настраиваем slave одной командой с полученными данными ранее	20
3.10.3.3	Запускаем slave	20
3.10.3.4	Проверяем статус	20

4.1	Создаём пользователя на Мастере MySQL.....	21
4.2	Запускаем на CS скрипт настройки:.....	21
4.2.1	Запускаем первый скрипт.....	21
4.2.2	Запускаем второй скрипт.....	21
4.2.3	Проверяем настройки в конфигурационном файле.....	22
4.2.3.1	Файл web.connections.config.....	22
4.2.3.2	Файл ASC.Xmpp.Server.Launcher.exe.config.....	22
	Примечание: пароль от пользователя root на VM с MySQL для подключения по localhost будет изменён на введённый Вами в скрипте для настройки Community Server.....	22
4.2.4	Подключаемся к MySQL на мастере и слайве.....	23
4.2.4.1	Проверяем, что БД появилась.....	23
4.2.4.2	Проверяем, что таблицы в БД есть.....	23
5.	Подключаем Документ Сервер к CS.....	24
5.1	Настраиваем Документ Сервер.....	25
5.1.1	Редактируем файл local.json.....	25
5.1.2	Перезапускаем сервис.....	25
5.1.3	Проверяем работу сервиса в браузере.....	26
5.2	Настраиваем Community Server.....	27
5.2.1	Правим конфигурационные файлы.....	27
5.2.1.1	web.appsettings.config.....	27
5.2.1.2	TeamLabSvc.exe.config.....	27
5.2.1.3	Перезапускаем сервис.....	27
6.	Подключаем дополнительный диск для CS.....	28
6.1	Добавляем диск к VM.....	28
6.1.1	Проверяем имя диска.....	28
6.1.2	Создаём раздел.....	29
6.1.3	Форматируем диск.....	30
6.2	Монтируем диск и переносим данные.....	30
6.2.1	Выполняем команду.....	30
6.2.2	Переносим данные.....	30
6.2.3	Монтируем диск в нужный нам раздел.....	30
6.2.4	Перезапускаем сервисы r7.....	30
6.2.5	Проверяем в браузере, что страница открывается корректно.....	30
6.2.6	Добавляем в fstab для автоматического монтирования диска.....	31
6.2.7	Перезагружаем сервер для проверки корректности настроек.....	31
7.	Регистрируемся на портале и проверяем работу.....	32

0. Описание архитектуры

В данной инструкции рассмотрим установку программного продукта P7 Сервер Базовый в архитектуре Middle в ручном режиме.

Ниже представлена схема архитектуры Middle

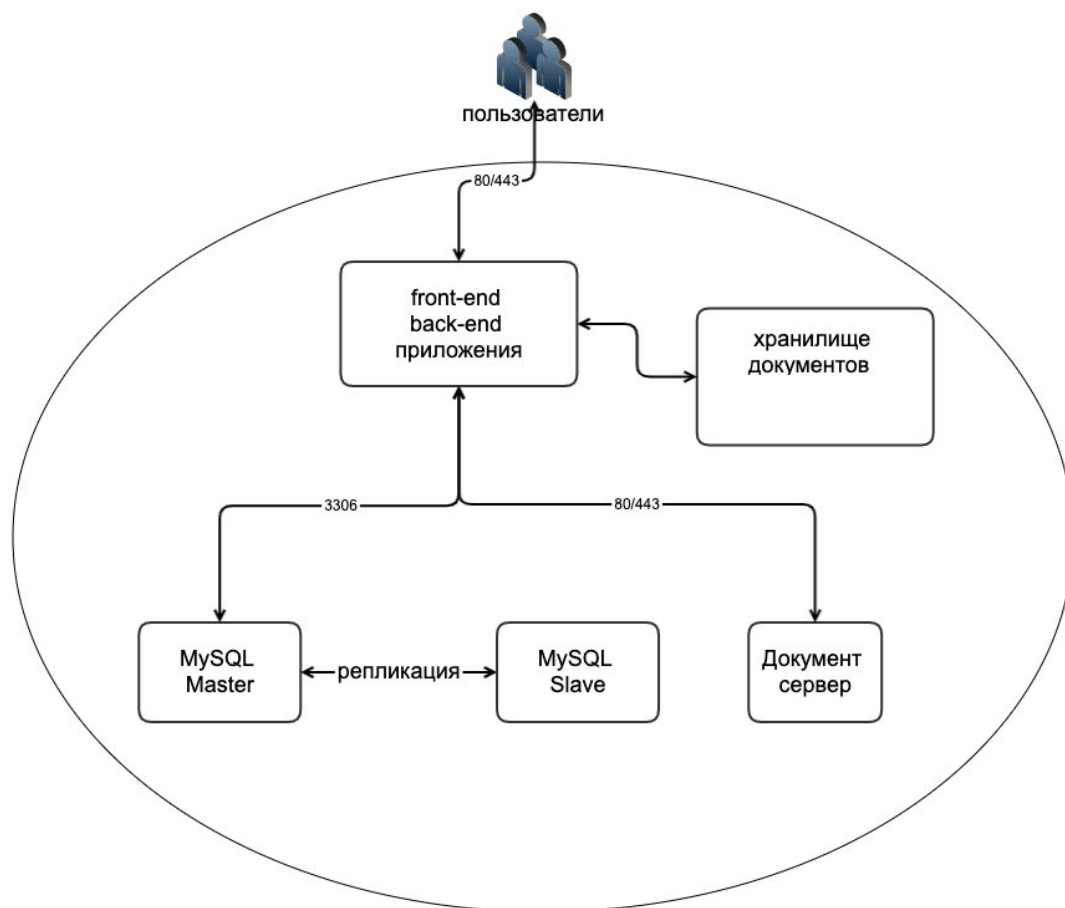


схема middle архитектуры

Преимущества данной архитектуры заключаются в повышении отказоустойчивости системы в целом и снижении нагрузки на сервер приложения путём выноса функциональных модулей.

0.1 Front-end / back-end приложения

Как в случае с Microsoft Windows, так и в случае с Linux версией продукта, front-end и back-end приложения размещается на одном сервере. Нагрузка на сервер приложений снижается за счёт размещения функциональных модулей на отдельных серверах.

0.2 Система Управления Базами Данных

База данных является неотъемлемой частью Продукта, обеспечивает хранение и управление следующими данными (укрупнённо):

- информация о пользователях системы;
- содержится мета информация для документов и писем;
- пользовательские данные по рабочим модулям Продукта.

Для устранения факторов, влияющих на деградацию производительности Программного комплекса в целом, в архитектурном решении Middle, система управления базами данных устанавливается на отдельные сервера для исключения воздействия сторонних систем, которые могут привести, в том числе, и к увеличению времени ожидания диска, что является одним из критических показателей для информационных систем.

0.3 Документ сервер

Система документ сервера в данном варианте инсталляции предусматривает размещение на отдельных серверах как в версии Docker контейнера так и с применением обычной установки.

Функционирование документ сервера возможно как на базе операционных систем типа Microsoft Windows так и Linux подобных системах.

Примечание: В командах ниже будет использоваться для редактирования утилита **mc**, можно использовать иной редактор (**vi**, **nano** и т.п.) или установить командой **yum install mc**.

1. Установка CS

Актуальная инструкция доступна на сайте: [Корпоративный сервер P7-Офис](#)

Далее будет описан пример установки для **RedOS 7.3**

1.1 Скачиваем скрипт

```
wget https://download.r7-office.ru/repo/install-RedOS.sh
```

1.2 Выполняем команды:

1.2.1 Обновление пакетов

```
yum update
```

1.2.2 Удаление пакетов, которые мешают инсталляции

```
yum remove mariadb-common mariadb-config mariadb-libs mariadb-connector-c  
mariadb-connector-c-config
```

1.2.3 Проверяем и, при наличии, удаляем конфигурационный файл

```
ls /etc/my.cnf  
rm /etc/my.cnf
```

1.2.4 Запускаем установку

```
bash install-RedOS.sh
```

Примечание:

При получении ошибки установки **GPG** ключей с получаемым информационным сообщением:

```
Ошибка: СБОЙ проверки GPG.
```

Выполните следующие команды:

```
yum clean packages
```

```
bash install-RedOS.sh -u true
```

1.3 Когда установка завершится

Можно убедиться в корректной работе **P7-Офис. Сервер. Профессиональный**. Откройте браузер (рекомендуем в инкогнито) и введите в адресной строке IP-адрес компьютера в **локальной сети** (Например, **http://192.168.0.15**). Серверная версия **P7-Офис. Сервер. Профессиональный** должна работать, что свидетельствует об успешной установке.

2. Установка DS

Актуальная инструкция доступна по ссылке: [Сервер документов P7-Офис](#)

Далее будет описан пример установки для **RedOS 7.3**

2.0 Обновляем пакеты

```
yum update
```

2.1 Создаём репозиторий **nginx** и устанавливаем

2.1.1 Открываем редактором

```
mcedit /etc/yum.repos.d/nginx.repo
```

2.1.2 Вставляем данные ниже

```
[nginx-stable]
name=nginx stable repo
baseurl=http://nginx.org/packages/centos/8/$basearch/
gpgcheck=1
enabled=1
gpgkey=https://nginx.org/keys/nginx_signing.key
[nginx-mainline]
name=nginx mainline repo
baseurl=http://nginx.org/packages/mainline/centos/8/$basearch/
gpgcheck=1
enabled=0
gpgkey=https://nginx.org/keys/nginx_signing.key
```

2.1.3 Устанавливаем **nginx** командой

```
yum install nginx
```

2.1.4 Редактируем основной конфигурационный файл

```
mcedit /etc/nginx/nginx.conf
```

Чтобы он имел вид:

```
user nginx;
worker_processes 1;

error_log /var/log/nginx/error.log warn;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile on;
    #tcp_nopush on;

    keepalive_timeout 65;

    #gzip on;

    include /etc/nginx/conf.d/*.conf;
}
```

2.2 Установка и настройка PostgreSQL

2.2.1 Установите пакет

```
yum install postgresql postgresql-server
```

2.2.2 Инициализируйте базу данных PostgreSQL и добавьте в автозагрузку

```
/usr/bin/postgresql-setup --initdb --unit postgresql
```

```
systemctl enable postgresql.service
```

2.2.3 Включите метод аутентификации 'trust'

2.2.3.1 Откройте файл в текстовом редакторе

```
mcedit /var/lib/pgsql/data/pg_hba.conf
```

2.2.3.2 Найдите строку host all all 127.0.0.1/32 ident и замените ее на следующую

```
host all all 127.0.0.1/32 trust
```

2.2.3.3 Затем найдите строку host all all ::1/128 ident и замените ее на следующую

```
host all all ::1/128 trust
```

2.2.4 Перезапустите сервис PostgreSQL

```
systemctl restart postgresql
```

2.2.5 Создайте базу данных и пользователя PostgreSQL

```
cd /tmp  
sudo -u postgres psql -c "CREATE DATABASE r7office;"  
sudo -u postgres psql -c "CREATE USER r7office WITH password 'r7office';"  
sudo -u postgres psql -c "GRANT ALL privileges ON DATABASE r7office TO r7office;"
```

2.2.6 Установка Redis

```
yum install redis  
systemctl start redis  
systemctl enable redis
```

2.2.7 Установка RabbitMQ

```
yum install rabbitmq-server  
systemctl start rabbitmq-server  
systemctl enable rabbitmq-server
```

2.3 Установка и настройка Сервера Документов

2.3.1 Добавьте репозиторий

2.3.1.1 Откройте редактором файл

```
mcedit /etc/yum.repos.d/r7-office.repo
```

2.3.1.2 Добавьте данные ниже

```
[r7-office]
name=r7-office repo
baseurl=http://download.r7-office.ru/repo/centos/main/noarch/
gpgcheck=0
enabled=1
```

2.3.1.3 Устанавливаем пакет DS командой

```
yum install r7-office-documentserver-ee
```

2.3.2 Запустите зависимые сервисы и добавьте их в автозагрузку

```
systemctl start nginx
systemctl enable nginx
```

2.3.3 Запустите скрипт настройки

```
bash documentserver-configure.sh
```

2.3.3.1 Параметры подключения к PostgreSQL, Redis и RabbitMQ

Используйте следующие данные:

2.3.3.1.1 Для PostgreSQL

```
Host: localhost
Database: r7office
User: r7office
Password: r7office
```

2.3.3.1.2 Для Redis

```
Host: localhost
```

2.3.3.1.3 Для AMQP

```
Host: localhost  
User: guest  
Password: guest
```

2.5 Если установлен и работает **firewalld**, добавьте исключения

```
firewall-cmd --zone=public --add-port=80/tcp --permanent  
firewall-cmd --reload
```

2.6 Отключите **SELINUX**

Отредактируйте конфигурационный файл

```
mcedit /etc/selinux/config
```

Чтобы в нём была строка

```
SELINUX=disabled
```

2.7 Перезагрузите сервер

2.8 Добавьте лицензию

Название файла и путь должны быть точно такими, как указано ниже

```
/var/www/r7-office/Data/license.lic
```

2.9 Проверка работы:

Откройте по **ip** сервера в браузере страницу. Должна быть картинка на сайте, что представлена далее:



3. Установка и настройка MySQL

Актуальную инструкцию рекомендуем смотреть у вендора ПО по ссылке:
[Инструкция](#)

3.1 Обновляем пакеты

```
yum update
```

3.2 Удаляем пакеты

```
yum remove mariadb-common mariadb-config mariadb-libs mariadb-connector-c  
mariadb-connector-c-config
```

3.3 Добавляем репозиторий

Для дистрибутива RedOS 7.3 и Centos 8

```
yum localinstall -y https://dev.mysql.com/get/mysql80-community-release-el8-  
9.noarch.rpm || true
```

Для дистрибутива RedOS 7.2 и Centos 7

```
yum localinstall -y https://repo.mysql.com//mysql80-community-release-el7-  
11.noarch.rpm || true
```

3.4 Устанавливаем MySQL

```
yum install mysql-community-server mysql-community-client
```

3.5 Запускаем MySQL

```
systemctl start mysqld  
systemctl enable mysqld
```

3.6 Узнаём пароль от root MySQL командой

```
cat /var/log/mysqld.log | grep "temporary password" | rev | cut -d " " -f 1 | rev | tail -1
```

3.7 Предварительная настройка для каждой ноды MySQL

3.7.1 Подключаемся к СУБД

```
mysql --connect-expired-password -uroot -p'ПАРОЛЬПОЛУЧЕННЫЙВЫШЕ' -D mysql
```

3.7.2 Обновляем пароль

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'НОВЫЙПАРОЛЬ';  
FLUSH PRIVILEGES;
```

3.8 Настройка Master

3.8.1 Обновляем конфигурационный файл my.cnf

```
mcedit /etc/my.cnf
```

К виду

```
[mysqld]  
server_id = 1  
log_bin = mysql-bin  
sql_mode = 'NO_ENGINE_SUBSTITUTION'  
max_connections = 1000  
group_concat_max_len = 2048  
max_allowed_packet = 1048576000  
character_set_server = utf8  
collation_server = utf8_general_ci  
default-authentication-plugin = mysql_native_password  
innodb_strict_mode = 0  
datadir=/var/lib/mysql  
socket=/var/lib/mysql/mysql.sock  
log-error=/var/log/mysql.log  
pid-file=/var/run/mysqld/mysqld.pid
```

3.8.2 Обновляем конфигурационный файл `mysqld.service`

```
mcedit /lib/systemd/system/mysqld.service
```

Приводим к виду

```
[Unit]
Description=MySQL Server
Documentation=man:mysqld(8)
Documentation=http://dev.mysql.com/doc/refman/en/using-systemd.html
After=network.target
After=syslog.target

[Install]
WantedBy=multi-user.target

[Service]
LimitMEMLOCK = infinity
User=mysql
Group=mysql
Type=notify
TimeoutSec=0
ExecStartPre=+/usr/bin/mysqld_pre_systemd
ExecStart=/usr/sbin/mysqld $MYSQLD_OPTS
EnvironmentFile=-/etc/sysconfig/mysql
LimitNOFILE = infinity
Restart=on-failure
RestartPreventExitStatus=1
Environment=MYSQLD_PARENT_PID=1
PrivateTmp=false
```

3.8.3 Перезапускаем сервис

```
systemctl daemon-reload
systemctl stop mysqld
systemctl restart mysqld
```

3.9 Настраиваем Slave

3.9.1 Обновляем конфигурационный файл my.cnf

```
mcedit /etc/my.cnf
```

К виду

```
[mysqld]
server_id = 2
log_bin = mysql-bin
relay-log = /var/lib/mysql/mysql-relay-bin
relay-log-index = /var/lib/mysql/mysql-relay-bin.index
read_only = 1

sql_mode = 'NO_ENGINE_SUBSTITUTION'
max_connections = 1000
group_concat_max_len = 2048
max_allowed_packet = 1048576000
character_set_server = utf8
collation_server = utf8_general_ci
default-authentication-plugin = mysql_native_password
innodb_strict_mode = 0

datadir=/var/lib/mysql
socket=/var/lib/mysql/mysql.sock
log-error=/var/log/mysql.log
pid-file=/var/run/mysqld/mysqld.pid
```

3.9.2 Обновляем конфигурационный файл `mysqld.service`

```
mcedit /lib/systemd/system/mysqld.service
```

Приводим к виду

```
[Unit]
Description=MySQL Server
Documentation=man:mysqld(8)
Documentation=http://dev.mysql.com/doc/refman/en/using-systemd.html
After=network.target
After=syslog.target

[Install]
WantedBy=multi-user.target

[Service]
LimitMEMLOCK = infinity
User=mysql
Group=mysql
Type=notify
TimeoutSec=0
ExecStartPre=+/usr/bin/mysqld_pre_systemd
ExecStart=/usr/sbin/mysqld $MYSQLD_OPTS
EnvironmentFile=-/etc/sysconfig/mysql
LimitNOFILE = infinity
Restart=on-failure
RestartPreventExitStatus=1
Environment=MYSQLD_PARENT_PID=1
PrivateTmp=false
```

3.9.3 Перезапускаем сервис

```
systemctl daemon-reload
systemctl stop mysql
systemctl restart mysql
```

3.10 Подключение SLAVE к MASTER

Примечание: Так как мы настраиваем на «пустых» и не введенных в работу СУБД репликацию, то можно опустить момент, что необходимо с мастера сделать дампы всех таблиц и прогрузить их на слейв, и перейти к п.3.10.1. НО, если Вы настраиваете на боевом MySQL репликацию, то необходимо сделать дампы всех баз на мастере командой:

```
mysqldump --all-databases --source-data -uroot -p > dump.sql
```

И прогрузить их на слейве командой (файл перенести на слейв предварительно):

```
mysql -uroot -p < dump.sql
```

3.10.1 Создаём учётку для подключения на мастере

Выполняем команды:

```
CREATE USER 'USER_SLAVE'@'IP_SLAVE' IDENTIFIED BY 'PASSWORD_SLAVE';  
GRANT replication slave ON *.* TO 'USER_SLAVE'@'IP_SLAVE';  
ALTER USER 'USER_SLAVE'@'IP_SLAVE' IDENTIFIED WITH mysql_native_password BY  
'PASSWORD_SLAVE';  
FLUSH PRIVILEGES;
```

Где,

USER_SLAVE – логин пользователя, что будет иметь доступ к репликации;
IP_SLAVE – ip адрес, с которого этот пользователь будет иметь возможность подключиться, т.е. ip адрес ВМ, где установлен slave MySQL;
PASSWORD_SLAVE – пароль от пользователя USER_SLAVE.

3.10.2 Получаем значения master_log и master_position

3.10.2.1 Выполняем команду на мастере:

```
show master status;
```

3.10.2.2 Сохраняем значения из столбцов File и Position

```
mysql> show master status;  
+-----+-----+-----+-----+-----+  
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB | Executed_Gtid_Set |  
+-----+-----+-----+-----+-----+  
| mysql-bin.000001 |      1180 |              |                  |                   |  
+-----+-----+-----+-----+-----+
```

3.10.3 Подключаемся на slave и выполняем команды

3.10.3.1 Останавливаем slave

```
stop replica;
```

3.10.3.2 Настраиваем slave одной командой с полученными данными ранее

```
CHANGE REPLICATION SOURCE TO SOURCE_HOST = 'IP_MASTER',  
SOURCE_USER = 'USER_SLAVE',  
SOURCE_PASSWORD = 'PASSWORD_SLAVE',  
SOURCE_LOG_FILE = 'mysql-bin.000001',  
SOURCE_LOG_POS = 1180;
```

Где,

IP_MASTER – ip адрес ВМ с мастером MySQL;
USER_SLAVE - логин пользователя, что будет иметь доступ к репликации;
PASSWORD_SLAVE - пароль от пользователя **USER_SLAVE**;
mysql-bin.000001 – имя файла, полученное в п.3.10.2.2 в графе **File**;
1180 – номер позиции, полученный в п.3.10.2.2 в графе **Position**.

3.10.3.3 Запускаем slave

```
start replica;
```

3.10.3.4 Проверяем статус

```
show replica status \G;
```

Здесь нас будут интересовать значения (скриншот ниже есть):

```
Replica_IO_State: Waiting for source to send event  
Replica_SQL_Running_State: Replica has read all relay log; waiting for more updates  
Last_IO_Errno  
Last_SQL_Error
```

Не должно быть ошибок и slave должен ожидать данных от мастера

```
mysql> show replica status \G;  
***** 1. row *****  
  Replica_IO_State: Waiting for source to send event  
    Source_Host: 172.16.2.33  
    Source_User: slave_repl  
    Source_Port: 3306  
    Connect_Retry: 60  
    Source_Log_File: mysql-bin.000001  
  Read_Source_Log_Pos: 1180  
    Relay_Log_File: mysql-relay-bin.000002  
    Relay_Log_Pos: 326  
  Relay_Source_Log_File: mysql-bin.000001  
  Replica_IO_Running: Yes  
  Replica_SQL_Running: Yes  
  Replicate_Do_DB:
```

4. Подключение CS к MySQL

4.1 Создаём пользователя на Мастере MySQL

```
CREATE USER 'root'@'IP_CS' IDENTIFIED BY 'PASSWORD_for_CS';  
GRANT ALL PRIVILEGES ON *.* TO 'root'@'IP_CS';  
FLUSH PRIVILEGES;
```

Где,

root – пользователь для подключения к БД с полными правами для БД r7-office, либо с правами создания БД
PASSWORD_for_CS – пароль, который будет использовать сервер CS и прописан в конфигурационных файлах
IP_CS – ip адрес сервер CS

4.2 Запускаем на CS скрипт настройки:

4.2.1 Запускаем первый скрипт

```
bash /usr/bin/communityserver-configure.sh
```

И указываем данные для подключения и имя БД. **Пароль** используем из п.4.1

```
[root@kh-csmiddle ~]# bash /usr/bin/communityserver-configure.sh  
Configuring MySQL access...  
Host: 172.16.2.33  
Database name: r7-office  
User: root  
Password:  
Trying to establish MySQL connection... OK  
█
```

Произойдут операции создания БД, прогон SQL скриптов, настройка конфигурационных файлов и перезапустятся сервисы.

4.2.2 Запускаем второй скрипт

```
bash /usr/bin/xmppserver-configure.sh
```

```
[root@kh-csmiddle ~]# bash /usr/bin/xmppserver-configure.sh  
Configuring MySQL access...  
Host: 172.16.2.33  
Database name: r7-office  
User: root  
Password:  
Trying to establish MySQL connection... OK  
Restarting services...  
r7-officeJabber.service  
OK
```

Произойдут операции настройки конфигурационных файлов и перезапустятся сервисы.

4.2.3 Проверяем настройки в конфигурационном файле

4.2.3.1 Файл web.connections.config

```
cat /var/www/r7-office/WebStudio/web.connections.config
```

```
[root@kh-csmiddle ~]# more /var/www/r7-office/WebStudio/web.connections.config
<?xml version="1.0" encoding="utf-8"?>
<connectionStrings>
  <clear />
  <add name="default" connectionString="Server=172.16.2.33;Database=r7-office;User ID=root;Password=bbThb75
r Set=utf8;AutoEnlist=false;SSL Mode=none;AllowPublicKeyRetrieval=true;Connection Timeout=30;Maximum Pool Size=3
MySQLClient"/>
</connectionStrings>
```

4.2.3.2 Файл ASC.Xmpp.Server.Launcher.exe.config

```
cat /var/www/r7-office/Services/Jabber/ASC.Xmpp.Server.Launcher.exe.config
```

```
[root@kh-csmiddle ~]# cat /var/www/r7-office/Services/Jabber/ASC.Xmpp.Server.Launcher.exe.config
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <section name="jabberServer" type="ASC.Xmpp.Server.Configuration.JabberConfigurationSection, ASC.Xmpp.Serv
    <section name="log4net" type="log4net.Config.Log4NetConfigurationSectionHandler, log4net" />
    <section name="nlog" type="NLog.Config.ConfigSectionHandler, NLog" />
    <section name="autofac" type="ASC.Common.DependencyInjection.AutofacConfigurationSection, ASC.Common" />
    <section name="consumers" type="ASC.Core.Common.Configuration.ConsumerConfigurationSection, ASC.Core.Common
    <section name="storage" type="ASC.Data.Storage.Configuration.StorageConfigurationSection, ASC.Data.Storage
    <section name="redisCacheClient" type="StackExchange.Redis.Extensions.LegacyConfiguration.RedisCachingSec
    Extensions.LegacyConfiguration" />
  </configSections>
  <connectionStrings>
    <clear />
    <add name="default" connectionString="Server=172.16.2.33;Database=r7-office;User ID=root;Password=bbThb75
    ter Set=utf8;AutoEnlist=false" providerName="MySql.Data.MySqlClient"/>
  </connectionStrings>
```

Примечание: пароль от пользователя **root** на ВМ с **MySQL** для подключения по **localhost** будет изменён на введённый Вами в скрипте для настройки **Community Server**.

4.2.4 Подключаемся к MySQL на мастере и слейве

4.2.4.1 Проверяем, что БД появилась

```
show databases;
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| r7-office |
| sys |
+-----+
```

4.2.4.2 Проверяем, что таблицы в БД есть

```
use r7-office;
show tables;
```

```
mysql> show tables;
+-----+
| Tables_in_r7-office |
+-----+
| account_links |
| audit_events |
| backup_backup |
| backup_schedule |
| blogs_comments |
| blogs_posts |
| blogs_reviewposts |
| blogs_tags |
| bookmarking_bookmark |
| bookmarking_bookmarktag |
| bookmarking_comment |
| bookmarking_tag |
| bookmarking_userbookmark |
| bookmarking_userbookmarktag |
| calendar_calendar_item |
| calendar_calendar_user |
| calendar_calendars |
| calendar_event_history |
| calendar_event_item |
+-----+
```

5. Подключаем Документ Сервер к CS

Примечание: Если устанавливаете DS версии 7.3.3 и подключаете к CS 12.1 и более ранней версии, то функциональность "Предоставления доступа" и Загрузки файлов из "Хранилища" во время работы с Документом, при использовании HTTP протокола могут не работать. Для решения проблемы потребуется перевести на HTTPS протокол серверы с CS и DS:

CS Linux: https://support.r7-office.ru/community_server/install-community_server/community_server_linux/community_server_https_own_certificate

DS Linux: https://support.r7-office.ru/document_server/install-document_server/document_server_linux/https_ds

Сервисы Корпоративный сервер и Сервер документов должны быть на одном домене и на одном его уровне

Например,

CS на домене второго уровня stgr7.ru и имеет FQDN имя = cs.stgr7.ru и

DS на домене второго уровня stgr7.ru и имеет FQDN имя = ds.stgr7.ru

5.1 Настраиваем Документ Сервер

5.1.1 Редактируем файл local.json

```
mcedit /etc/r7-office/documentserver/local.json
```

И приводим к виду

```
},
  "token": {
    "enable": {
      "request": {
        "inbox": true,
        "outbox": true
      },
      "browser": true
    },
    "inbox": {
      "header": "AuthorizationJwt"
    },
    "outbox": {
      "header": "AuthorizationJwt"
    }
  },
  "secret": {
    "inbox": {
      "string": "RrTCPPwQ1wq2"
    },
    "outbox": {
      "string": "RrTCPPwQ1wq2"
    },
    "session": {
      "string": "RrTCPPwQ1wq2"
    }
  }
}
```

Где,

RrTCPPwQ1wq2 – Это своего рода пароль;

AuthorizationJwt – хедер, который указывает авторизацию по Jwt.

5.1.2 Перезапускаем сервис

```
supervisorctl restart all
```

Если DS версии 7.3.3 или выше, то используйте следующие команды для перезапуска:

```
systemctl restart ds-converter.service
systemctl restart ds-docservice.service
systemctl restart ds-metrics.service
```

```
[root@kh-dsmiddle ~]# supervisorctl restart all
ds:docservice: stopped
ds:converter: stopped
ds:metrics: stopped
ds:docservice: started
ds:converter: started
ds:metrics: started
[root@kh-dsmiddle ~]# supervisorctl status all
ds:converter          RUNNING   pid 154703, uptime 0:00:06
ds:docservice         RUNNING   pid 154702, uptime 0:00:06
ds:metrics            RUNNING   pid 154704, uptime 0:00:06
```

5.1.3 Проверяем работу сервиса в браузере

Переходим по ip DS, должен быть результат, что ниже



Сервер документов запущен

5.2 Настраиваем Community Server

5.2.1 Правим конфигурационные файлы

5.2.1.1 web.appsettings.config

```
mcedit /var/www/r7-office/WebStudio/web.appsettings.config
```

Приводим к виду ниже строки, где

В значениях:

```
files.docservice.secret указываем СЕКРЕТ, указанный в local.json на DS,  
files.docservice.secret.header указываем хедер также из конфига local.json,  
files.docservice.url.public указываем протокол://IP_DS/  
files.docservice.url.internal оставляем пустым ""  
files.docservice.url.portal пишем протокол://IP_CS/
```

Пример ниже:

```
<add key="files.docservice.secret" value="RrTCPwQ1wq2" />  
<add key="files.docservice.secret.header" value="AuthorizationJwt" />  
<add key="files.docservice.url.public" value="http://172.16.2.44/" />  
<add key="files.docservice.url.internal" value="" />  
<add key="files.docservice.url.portal" value="http://172.16.2.25/" />
```

5.2.1.2 TeamLabSvc.exe.config

```
mcedit /var/www/r7-office/Services/TeamLabSvc/TeamLabSvc.exe.config
```

Приводим к тем же значениям аналогичные переменные, как в примере выше

5.2.1.3 Перезапускаем сервис

```
systemctl restart monoserve.service
```

6. Подключаем дополнительный диск для CS

Подключаете дополнительный диск удобным для Вас способом.

Пример будет далее, когда добавляем на ВМ виртуальный второй диск на площадке **Selectel**.

6.1 Добавляем диск к ВМ

kh-csmiddle
ACTIVE

⏻
🔄
⋮

ru-3b © • 1ff33752-cf89-491b-93a1-a3dd7f4c0501

Конфигурация
Сетевые диски
Порты
Статистика
Syslog
Консоль

Имя	Тип	Размер, ГБ	Планы бэкапов	
kh-csmiddle Загрузочный	Быстрый	65	Настроить	⋮
kh-cs-data	Базовый	100	Настроить	⋮

6.1.1 Проверяем имя диска

```
fdisk -l
```

```
[root@kh-csmiddle ~]# fdisk -l
Диск /dev/sda: 65 GiB, 69793218560 байт, 136314880 секторов
Disk model: QEMU HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
Тип метки диска: dos
Идентификатор диска: 0xcbe81917

Устр-во      Загрузочный  начало      Конец      Секторы  Размер  Идентификатор  Тип
/dev/sda1    *            2048 136314846 136312799 65G      83 Linux

Диск /dev/loop0: 46,96 MiB, 49242112 байт, 96176 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт

Диск /dev/loop1: 61,96 MiB, 64966656 байт, 126888 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт

Диск /dev/loop2: 43,86 MiB, 45985792 байт, 89816 секторов
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт

Диск /dev/sdb: 100 GiB, 107374182400 байт, 209715200 секторов
Disk model: QEMU HARDDISK
Единицы: секторов по 1 * 512 = 512 байт
Размер сектора (логический/физический): 512 байт / 512 байт
Размер I/O (минимальный/оптимальный): 512 байт / 512 байт
```

6.1.2 Создаём раздел

```
fdisk /dev/sdb
```

И вводим по очереди

```
d
n
p
Далее оставляем по умолчанию и нажимаем Enter
w
```

6.1.3 Форматируем диск

```
mkfs -t ext4 /dev/sdb1
```

Проверяем командой:

```
lsblk -f
```

```
[root@kh-csmiddle ~]# lsblk -f
NAME        FSTYPE     FSVER          LABEL     UUID                                 FSAVAIL  P
loop0      squashfs   4.0
loop1      squashfs   4.0
loop2      squashfs   4.0
sda
├─sda1     ext4       1.0            67d406e8-4dde-492d-81cf-7776c8c4e0bf  42,3G
sdb
├─sdb1     ext4       1.0            e14e6b47-d96b-408e-bce7-fb2d4c7e3ea3
```

6.2 Монтируем диск и переносим данные

6.2.1 Выполняем команду

```
mkdir /mnt/data
mount /dev/sdb1 /mnt/data
```

6.2.2 Переносим данные

```
cd /var/www/r7-office
cp -pr .* /mnt/data/
```

6.2.3 Монтируем диск в нужный нам раздел

```
umount /dev/sdb1
mount /dev/sdb1 /var/www/r7-office/
```

6.2.4 Перезапускаем сервисы r7

```
systemctl restart monoserve r7-office*
```

6.2.5 Проверяем в браузере, что страница открывается корректно

Переходим в браузере по ip сервер CS

Добро пожаловать на портал Р7-Офис



Благодарим Вас за выбор Р7-Офис! В целях безопасности необходимо выполнить процедуру установки пароля.

Пароль

Введите пароль (минимум 8 символов)*

Подтвердить пароль*

Файл активации лицензии

Загрузить лицензию*

Загрузить файл

Настройки регистрации

Ваш портал зарегистрирован со следующего адреса электронной почты:

Ваш портал зарегистрирован за следующим доменом: **kh-csmiddle** ?

Настройки языка и часового пояса

Язык:

русский (Россия) ▼

Часовой пояс:

(UTC+03:00) Европа/Москва ▼

Продолжить

6.2.6 Добавляем в fstab для автоматического монтирования диска

```
mcedit /etc/fstab
```

Добавляем одной строкой в конце файла:

```
UUID=e14e6b47-d96b-408e-bce7-fb2d4c7e3ea3 /var/www/r7-office/ ext4 defaults  
1 1
```

UUID можно получить, выполнив команду

```
blkid
```

```
[root@kh-csmiddle ~]# blkid  
/dev/sda1: UUID="67d406e8-4dde-492d-81cf-7776c8c4e0bf" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="cbe81917-01"  
/dev/loop1: TYPE="squashfs"  
/dev/sr0: BLOCK_SIZE="2048" UUID="2022-08-30-13-28-19-00" LABEL="config-2" TYPE="iso9660"  
/dev/loop2: TYPE="squashfs"  
/dev/loop0: TYPE="squashfs"  
/dev/sdb1: UUID="e14e6b47-d96b-408e-bce7-fb2d4c7e3ea3" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="770d4d00-01"
```

6.2.7 Перезагружаем сервер для проверки корректности настроек

```
reboot
```

7. Регистрируемся на портале и проверяем работу

Для регистрации необходим файл лицензии