Перенос сервера на Fedora OS

Используемая версия Redhat: Fedora Server 23 (также тестировалось на CentOS Mini 6.9)

# Установка Fedora

Проблемы, устранённые после установки:

1. Вместо русских символов отображались квадратики (□)

Решение: в файле /etc/profile (/etc/profile.local) прописать:

setfont Cyr\_a8x16

1. После закрытия крышки ноутбука система уходила в гибернацию

Решение: в файле /etc/systemd/logind.conf меняем параметр:

HandleLidSwitch=ignore

перезагружаем сервис:

service restart systemd-logind

# Подготовка сервера к работе

Создаём нового пользователя:

useradd mitrakov

passwd mitrakov

Прописываем пользователя в sudoers:

visudo

Добавляем строку:

mitrakov ALL=(ALL) ALL

Подключаем сетевой кабель к Ethernet-интерфейсу и конфигурируем статический IP-адрес. Для этого открываем файл /etc/sysconfig/network-scripts/ifcfg-em1

Во-первых, меняем BOOTPROTO с dhcp на static:

BOOTPROTO="static"

Во-вторых, добавляем следующие настройки:

IPADDR=192.168.1.3

NETMASK=255.255.255.0

BROADCAST=192.168.1.255

NETWORK=192.168.1.0

GATEWAY=192.168.1.1

Рестартуем сервис:

systemctl restart network.service

Также конфигурируем DNS:

В файле /etc/resolv.conf прописываем серверы DNS:

nameserver 192.168.1.1

nameserver 8.8.8.8

Открываем порты для веб-сервера, SSH:

iptables -I INPUT 1 -p tcp --dport 80 -j ACCEPT

iptables -I INPUT 1 -p tcp --dport 8080 -j ACCEPT

iptables -I INPUT 1 -p tcp --dport 22 -j ACCEPT

Настраиваем перенаправление портов: трафик, приходящий на порт 80, будет перенаправлен на 8080:

iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-ports 8080

iptables -t nat -A PREROUTING -p udp --dport 80 -j REDIRECT --to-ports 8080

Вместо перенаправления портов можно использовать утилиту authbind (см. далее).

# Подготовка маршрутизатора

На маршрутизаторе в раздел «Виртуальный сервер» следует добавить два правила:

* TCP-трафик с порта 80 перенаправить на 192.168.1.3:80 (для HTTP-сервера)
* TCP-трафик с порта 54122 перенаправить на 192.168.1.3:22 (для работы SSH)

C:\Users\Tommy\YandexDisk\Скриншоты\2016-01-07 21-55-45 2016-01-07 14-56-19 ASUS Wireless Router - Yandex - Windows Photo Viewer.png

Также при использовании хоста для сервера имён следует добавить правило для порта 53 (см. далее).

# Подготовка к работе Tomcat

Tomcat по умолчанию работает на порту 8080. Чтобы его использовать в роли веб-сервера, требуется либо перенаправить трафик с порта 80 на порт 8080 (уже сделано ранее), либо привязать сервер на порт 80 и использовать утилиту authbind, которая разрешит привязку к порту, меньшему 1024[[1]](#footnote-1).

Укажем редирект по умолчанию:

В каталог /var/lib/tomcat/webapps/ROOT добавить файл index.html:

<?xml version="1.0" encoding="ISO-8859-1"?>

<!DOCTYPE html>

<html>

<head>

<title>Apache Tomcat</title>

<meta http-equiv="refresh" content="0; url=http://2xdx.ru/guap" />

</head>

<body>

</body>

</html>

# Проблемы с Tomcat

Недавно (2017-09-13) обнаружил проблему с извлечением данных из заголовков HTTP; если данные не в латинице, то заголовок м.б. проигнорирован. Для решения проблемы следует указать кодировку сервера явно (/etc/tomcat/server.xml):

<Connector port="80" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443"

URIEncoding="UTF-8" />

# Подготовка к работе PostgreSQL

Устанавливаем дополнения:

dnf install postgresql-server postgresql-contrib

Инициализируем сервис Postgresql:

systemctl enable postgresql

postgresql-setup initdb

chkconfig postgresql on

service postgresql start

*Примечание: для CentOS 6 команды будут следующими:*

service postgresql initdb

chkconfig postgresql on

service postgresql start

Заходим в оболочку psql:

sudo su - postgres

psql

Создаём юзера и БД (не забываем ставить точку с запятой в конце каждой команды):

CREATE ROLE mitrakov LOGIN password '555';

CREATE DATABASE varlam ENCODING 'UTF8' OWNER mitrakov;

Если вдруг вылезла ошибка:

new encoding (UTF8) is incompatible with the encoding of the template database (SQL\_ASCII)

то задаём новую БД через шаблон:

CREATE DATABASE varlam WITH TEMPLATE template0 ENCODING 'UTF8' OWNER mitrakov;

Открываем файл конфигурации /var/lib/pgsql/data/postgresql.conf и изменяем параметр:

listen\_addresses = '\*'

Это позволит администрировать БД с любого хоста (по умолчанию разрешён только localhost).

*Примечание: вообще, такой подход не рекомендуется: лучше указать список доверенных IP-адресов через запятую.*

Изменяем права доступа к БД varlam. Для этого открываем файл /var/lib/pgsql/data/pg\_hba.conf (/etc/postgresql/9.1/main/pg\_hba.conf) и добавляем следующие записи:

# TYPE DATABASE USER ADDRESS METHOD

host varlam mitrakov 127.0.0.1/32 md5

host varlam all 192.168.1.2/32 md5

host varlam all 212.33.245.191/32 md5

local varlam mitrakov md5

Поясним указанное:

1. даёт право доступа к БД с локального хоста
2. даёт право доступа к БД с хоста 192.168.1.2 (для удобства администратора)
3. даёт право доступа к БД с внешнего IP-адреса (требуется для работы веб-сервера)
4. даёт право доступа к БД через Unix-сокет (требуется для создания Backup)

Обязательно закомментировать все записи с методом ident (иначе будет выбираться именно этот метод).

Убеждаемся, что добавлено правило в IP-Tables:

iptables -I INPUT 1 -p tcp --dport 5432 -j ACCEPT

Перезагружаемся:

service postgresql restart

# Восстановление данных в PostgreSQL

Создаём дамп БД на старом сервере:

sudo PGPASSWORD="1234" pg\_dump -U mitrakov varlam > backup.sql

Копируем его (обратите внимание: для SSH используется порт 54122 вместо 22):

scp -P 54122 backup.sql 212.33.245.191:~/

Восстанавливаем из дампа на новый сервер:

psql varlam < backup.sql

# Подготовка к работе Redis

Устанавливаем redis (redis-server в некоторых дистрибутивах)

dnf install redis

# Восстановление данных в Redis

Узнаём, был ли включён AOF. Для этого следует прочесть файл /etc/redis/redis.conf (/etc/redis.conf):

cat /etc/redis/redis.conf | grep 'appendonly'

Если на обоих серверах он отключён (ответ ‘no’), то просто переносим дамп БД.

Останавливаем Redis (на обоих серверах)

service redis stop

Копируем дамп:

scp -P 54122 /var/lib/redis/redis.rdb mitrakov@212.33.245.191:~/

Меняем права дампа:

chown redis: dump.rdb

Копируем его в каталог Redis, заменяя существующий файл БД, и запускаем сервис (на обоих серверах):

chkconfig redis on

service redis start

# Перенос веб-приложения

На старом сервере пакуем архив:

tar -cf a.tar manager/ guap/ varlam.war

Копируем архив на новый сервер:

scp -P 54122 a.tar mitrakov@212.33.245.191:~/

Теперь переносим всё в каталог /var/lib/tomcat/webapps.

Для того чтобы веб-приложение могло создавать файлы картинок, мы обязательно меняем права на папку guap:

chown -R tomcat: guap/

Стартуем сервис:

service tomcat start

# Настройка почтового сервера

Почтовый сервер требуется для отправки Backup’ов базы данных. Если для этих целей существует некий внешний диск, то данный шаг можно пропустить (а соответствующий скрипт изменить).

Устанавливаем ssmtp:

dnf install ssmtp

Открываем файл /etc/ssmtp/ssmtp.conf и убеждаемся в наличии следующих настроек:

root=postmaster

mailhub=mail

mailhub=smtp.yandex.ru:465

FromLineOverride=YES

UseTLS=YES

AuthUser=tom-trix@yandex.ru

AuthPass=12345

Убедитесь, что в конце файла имеется хотя бы одна пустая строка (иначе возможны ошибки).

Теперь открываем файл /etc/ssmtp/revaliases и прописываем:

root:tom-trix@yandex.ru:smtp.yandex.ru:465

mitrakov:tom-trix@yandex.ru:smtp.yandex.ru:465

При правке обоих файлов имейте в виду, что номер порта исходящей почты может быть различным в разных сервисах (для Яндекса это 465).

Теперь для отправки писем можно использовать команду:

echo -e "to: bobby\nsubject: hello\n" | (cat - && cat /home/tommy/1.txt) | /usr/sbin/ssmtp [bobby@yandex.ru](mailto:bobby@yandex.ru)

## Troubleshooting

Trouble (2018.03.14):

Cannot open smtp.yandex.ru:465

Solution: добавить в тот же конфиг:

TLS\_CA\_File=/etc/pki/tls/certs/ca-bundle.crt

Reference: <https://prosto-tak.ru/ssmtp-i-yandex-pochta/>

# Скрипты, запускаемые в Cron

1. Скрипт отправки Backup базы данных (/var/lib/tomcat/scripts/backup.sh):

#!/bin/bash

#backup for varlam DB (nightly at 1:00 AM)

#ensure there is a line 'local varlam mitrakov md5' in pg\_hba.conf

#cron cmd: 0 1 \* \* \* /var/lib/tomcat/scripts/backup.sh

NOW=$(date +%Y-%m-%d\_%H-%M-%S)

TMP=/home/mitrakov/backup\_$NOW.sql

ADDR=mitrakov-artem@yandex.ru

#sudo PGPASSWORD="541888" pg\_dump -U mitrakov varlam > $TMP[[2]](#footnote-2)

su - mitrakov -c "pg\_dump -U mitrakov varlam > $TMP"

echo -e "to: $ADDR\nsubject: Backup\n" | (cat - && cat $TMP) | /usr/sbin/ssmtp $ADDR

rm $TMP

1. Скрипт удаления картинок с сервера (/var/lib/tomcat/scripts/remove\_charts.sh):

#!/bin/bash

#script to remove all chart pictures

#cron cmd: 0 1 \* \* \* /var/lib/tomcat/scripts/remove\_charts.sh

CHARTSDIR=/var/lib/tomcat/webapps/guap/charts

rm -rf $CHARTSDIR

mkdir $CHARTSDIR

chown tomcat:tomcat $CHARTSDIR

Не забываем файлы скриптов сделать исполняемыми:

chmod +x backup.sh

chmod +x remove\_charts.sh

Теперь прописываем скрипты в Cron:

crontab -e

В редакторе вводим строки:

0 1 \* \* \* /var/lib/tomcat/scripts/backup.sh

0 1 \* \* \* /var/lib/tomcat/scripts/remove\_charts.sh

Скрипты будут запускаться ежедневно в 1:00.

## Перенос доменного имени (deprecated since 2017)

Данная настройка требуется только в том случае, если хост используется в качестве сервера имён.

При занесении DNS-записи регистратор требует наличия, как минимум, двух серверов имён. Если поставщик доменного имени готов предоставить два сервера имён, то данный раздел можно пропустить, а дальнейшие инструкции см. у поставщика.

Устанавливаем bind-сервер:

dnf install bind

В файле /named/named.conf (/etc/bind/named.conf) добавляем новую зону (например, после зоны «.»):

zone "2xdx.ru" {

type master;

file "2xdx.ru";

};

По умолчанию сервер имён прослушивает localhost. Поэтому в том же файле меняем настройки:

listen-on port 53 { any; };

Также следует закомментировать/удалить строку:

allow-query { localhost; };

Теперь создаём файл описания зоны (/var/named/2xdx.ru):

$TTL 3600

2xdx.ru. IN SOA 2xdx.ru. root.2xdx.ru. (2016010701 10800 3600 604800 86400)

2xdx.ru. IN NS ns1.firstvds.ru.

2xdx.ru. IN NS ns2.firstvds.ru.

2xdx.ru. IN A 212.33.245.191

www IN A 212.33.245.191

Обязательно меняем права на этот файл:

chown named: /var/named/2xdx.ru

Перезагружаем настройки:

rndc reload

Должен появиться ответ:

server reload successful

Также не забываем на маршрутизаторе открыть UDP-порт 53 для сервиса DNS.

Теперь указываем IP-адрес сервера у поставщика домена.

1. Напомним, что в Linux приложение не может привязаться к порту, меньшему 1024, без соотв. прав [↑](#footnote-ref-1)
2. Данный вариант выбрать в том случае, если пароль задан к БД, а не к пользователю [↑](#footnote-ref-2)