ZABBIX 5.0

ШИФРОВАНИЕ СОЕДИНЕНИЙ БАЗЫ ДАННЫХ, СЕРВЕРА И ФРОНТЕНДА

Александр Петров-Гаврилов
Инженер Технической поддержки

01

ЧТО ЗНАЧИТ “БЕЗОПАСНОСТЬ”?

Состояние, при котором не угрожает опасность, есть защита от опасности.

- Толковый словарь русского языка

ЧТО ТАКОЕ ШИФРОВАНИЕ?

ЧТО ТАКОЕ ШИФРОВАНИЕ?

- Процесс конвертирующий информацию в изначальном виде так же известном как открытый текст, в альтернативную форме известную как шифротекст.
- Авторизованные стороны, могут дешифровать шифротекст обратно в открытый, получив доступ к изначальное информации.
- Шифрование не предотвращает перехват данных, но сохраняет оригинальную информацию от перехватившего её.
- Может быть симметричным и асимметричным.

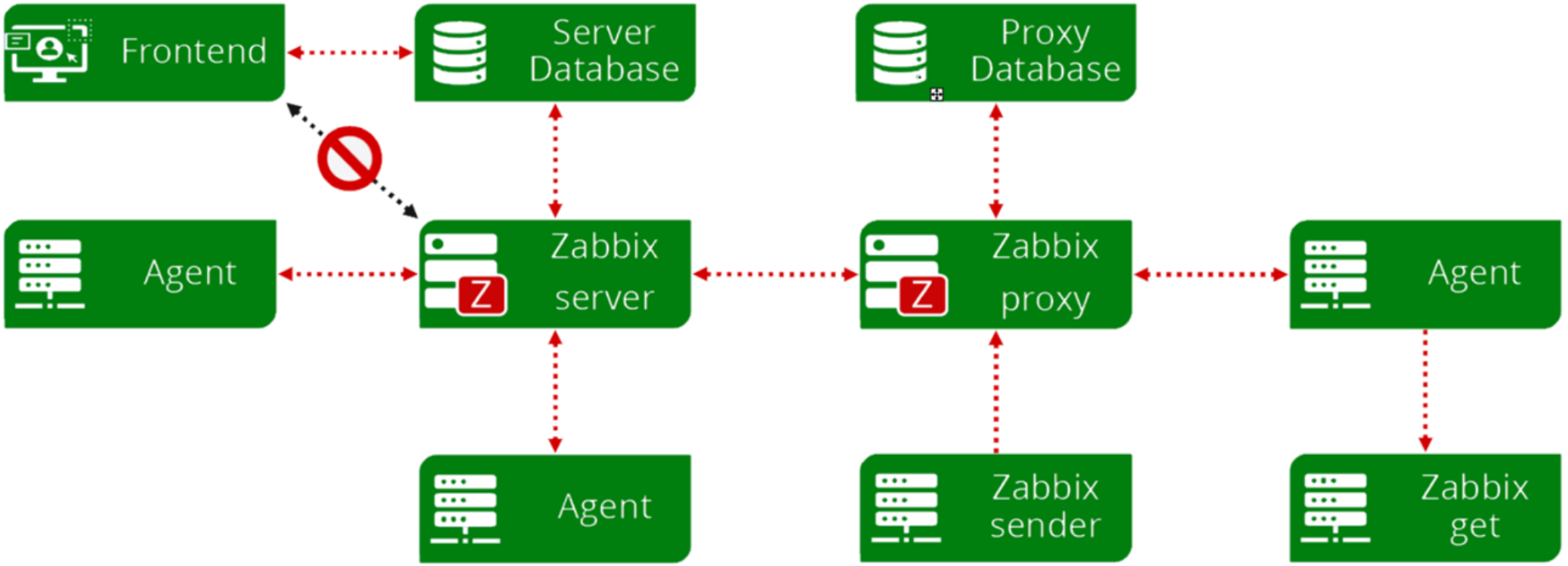
ЗАЧЕМ НЕОБХОДИМО ШИФРОВАНИЕ?

- Чтобы обезапасить данные.
- Чтобы сохранить информацию.
- Для конфеденциальности.
- И т.е.

ZABBIX И ШИФРОВАНИЕ

- Zabbix поддерживает шифрованные соеденинения между компонентами Zabbix используя протокол Transport Layer Security (Протокол защиты транспортного уровня, TLS) версий 1.2 и 1.3 (в зависимости от крипто библиотек).
- Поддерживается так же шифрование на основе сертификатов и “предварительного общего ключа” - PSK

ZABBIX И ШИФРОВАНИЕ



ПРИМЕР ШИФРОВАНИЯ

1. Сгенерируйте PSK ключ и сохраните в файл

```
# openssl rand -hex 32 > /etc/zabbix/keys/agent.psk
# chmod 400 /etc/zabbix/keys/agent.psk
# chown zabbix:zabbix /etc/zabbix/keys
```

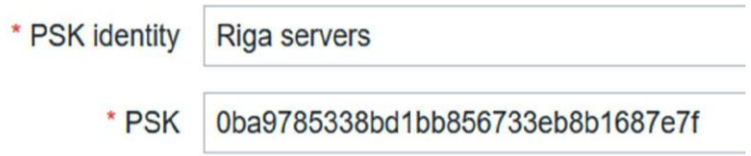
2. Измените конфигурацию Zabbix агента

```
# TLSAccept=psk
# TLSPSKIdentity=Riga servers
# TLSPSKFile=/etc/zabbix/keys/agent.psk
```

3. Перезапустите Zabbix агент

```
# systemctl restart zabbix-agent
```

4. Обновите информацию во фронтенде

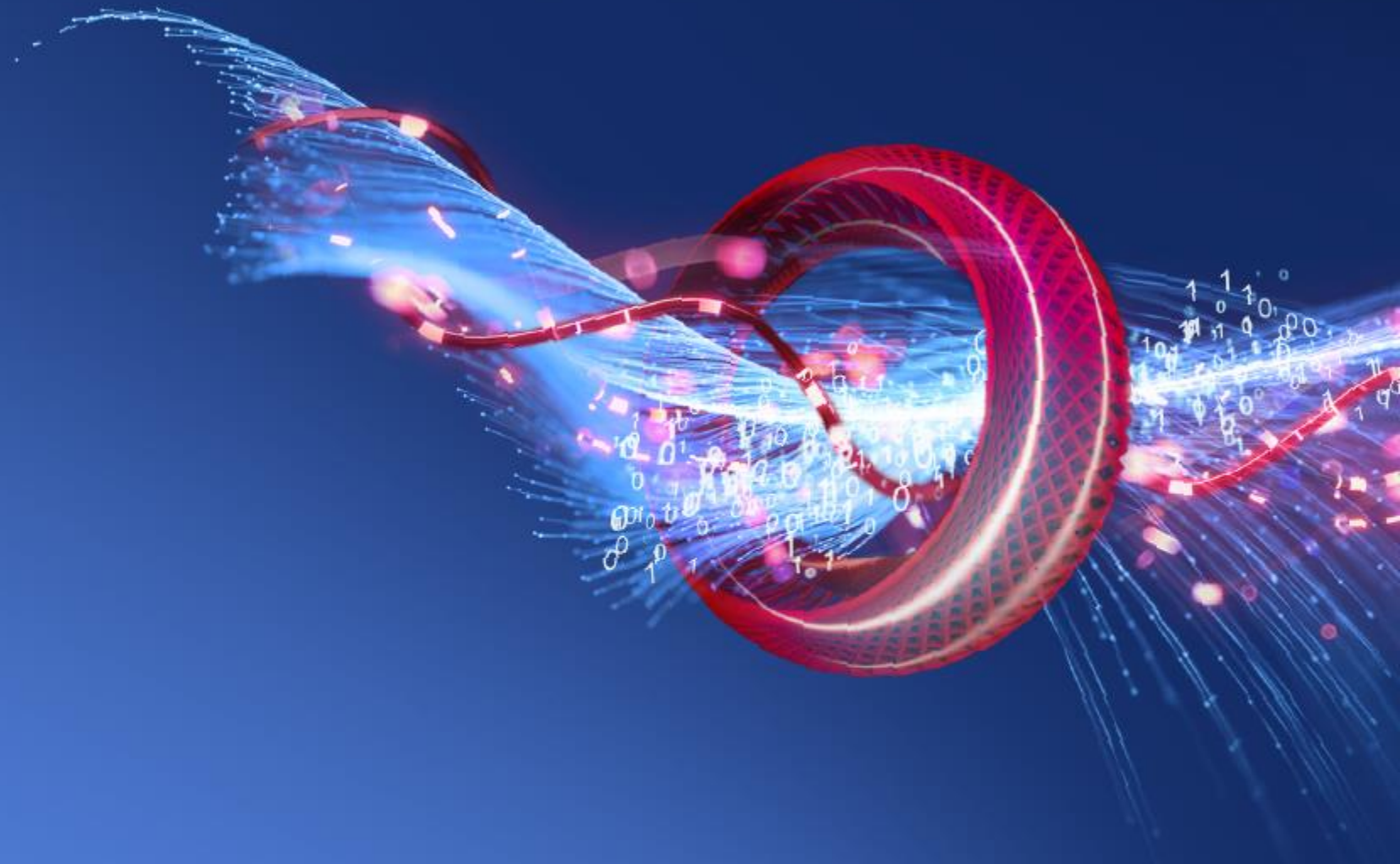


ПОЧЕМУ ИМЕННО БАЗА ДАННЫХ?

Потому что данные

02

ПОДГОТОВКА



СЕТАП?

СЕТАП

- Zabbix 5.0
- Centos 8 (x2 VMs)
- MySQL 8

03

КАК НАСТРОИТЬ ШИФРОВАННОЕ СОЕДЕНИЕ МЕЖДУ БАЗОЙ ДАННЫХ И ZABBIX

КАК?

Рассказать
о шифровании

Показать
как настроить
шифрование

ПОДГОТОВИТЬ СЕРВЕРА

1. Установить CentOS 8, решить где будет база данных и где Zabbix, отредактировать hostnames

```
# vi /etc/hostname
```

2. Назвав, например:

   a. `mydb.localhost.local`

   b. `cazabbix.localhost.local`

3. И отредактировать hosts файл на каждой из машин:

```
# vi /etc/hosts
```

```
192.168.3.92    cazabbix.localhost.local
192.168.3.86    mydb.localhost.local
```

СОЗДАЁМ CA (ЦЕНТР СЕРТИФИКАЦИИ)

1. На будущем Zabbix сервере, проверьте где расположен openssl.cnf

```
# find / -iname openssl.cnf
```

`/etc/pki/tls/openssl.cnf` - By default

2. Найдите расположение CA файлов

```
# cat /etc/pki/tls/openssl.cnf | grep dir
dir               = /etc/pki/CA             # Where everything is kept
```

3. Создайте там же субдиректорию, например:

```
# mkdir /etc/pki/CA/private
```

СОЗДАЁМ CA (ЦЕНТР СЕРТИФИКАЦИИ)

1. Создайте пару ключей и подпишите их(self-sign)

```
# openssl req  -new -x509 -keyout /etc/pki/CA/private/cakey.pem -out /etc/pki/CA/cacert.pem -days 3652 -newkey rsa:4096
```

2. Вы увидите запрос на создание пароля и форму для указания владельца сертификата

```
Generating a RSA private key
................++++
.....................++++
writing new private key to '/etc/pki/CA/private/cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:LV
State or Province Name (full name) []:Riga
Locality Name (eg, city) [Default City]:Riga
Organization Name (eg, company) [Default Company Ltd]:SIA ZABBIX
Organizational Unit Name (eg, section) []:SUPPORT
Common Name (eg, your name or your server's hostname) []:Root CA
Email Address []:
```

ПОДПИСЫВАЕМ ЗАПРОСЫ

1. Чтобы все было чуть легче и менее путанно, можно отредактировать openssl.cnf

```
# vi /etc/pki/tls/openssl.cnf
```

2. Изменив данные об организации, указанные по умолчанию в форме:

```
[ req_distinguished_name ]
countryName                     = Country Name (2 letter code)
countryName_default             = LV
countryName_min                 = 2
countryName_max                 = 2

stateOrProvinceName             = Riga
#stateOrProvinceName_default    = Default Province

localityName                    = Riga
localityName_default            = Riga

0.organizationName              = Organization Name (eg, company)
0.organizationName_default      = SIA ZABBIX

# we can do this but it is not needed normally :-)
#1.organizationName             = Second Organization Name (eg, company)
#1.organizationName_default     = World Wide Web Pty Ltd

organizationalUnitName          = Organizational Unit Name (eg, section)
#organizationalUnitName_default = SIA ZABBIX

commonName                      = Common Name (eg, your name or your server\'s hostname)
commonName_max                  = 64

emailAddress                    = Email Address
emailAddress_max                = 64
```

ПОДПИСЫВАЕМ ЗАПРОСЫ

1. Создайте директорию для запросов на подписание:

```
# mkdir -p /etc/pki/CA/requests
```

2. Создайте директорию для новых сертификатов:

```
# mkdir -p /etc/pki/CA/newcerts
```

3. Создайте запрос на подписание для Zabbix/CA server, с common name cazabbix.localhost.local :

```
# openssl req -new -keyout /etc/pki/CA/private/zaca_key.pem -out /etc/pki/CA/requests/zaca_req.pem -newkey rsa:2048
```

4. Создайте ещё один запрос на подписание для, но для сервера с БД, common name mydb.localhost.local

```
# openssl req -new -keyout /etc/pki/CA/private/pdb_key.pem -out /etc/pki/CA/requests/pdb_req.pem -newkey rsa:2048
```

ПОДПИСЫВАЕМ ЗАПРОСЫ

1. Создайте index.txt и серийные файлы необходимые openssl для учёта подписанных сертификатов:

```
# touch /etc/pki/CA/index.txt
# echo 01 > /etc/pki/CA/serial
```

2. Создайте подписанный сертификат для Zabbix/CA(подтвердите при запросе):

```
# openssl ca -policy policy_anything -days 365   -out /etc/pki/CA/certs/zaca_crt.pem  -infiles /etc/pki/CA/requests/zaca_req.pem
Certificate is to be certified until Aug 10 08:43:26 2021 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

3. Создайте подписанный сертификат для DB (подтвердите при запросе):

```
# openssl ca -policy policy_anything -days 365   -out /etc/pki/CA/certs/pdb_crt.pem  -infiles /etc/pki/CA/requests/pdb_req.pem
Certificate is to be certified until Aug 10 08:43:45 2021 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

НАСТРАИВАЕМ MYSQL СЕРВЕР

1. Установите MySQL 8

```
# sudo dnf install mysql-server
```

2. Запустите MySQL

```
# systemctl start mysqld
```

3. Запустите скрипт безопасности(на все отвечайте да)

```
# mysql_secure_installation
```

4. Создайте директорию для сертификатов:

```
# mkdir -p /var/lib/mysql/pki
```

5. Проверьте firewall, чтобы убедиться, что порты для MySQL открыты, например:

```
# firewall-cmd --permanent --zone=trusted --add-source=192.0.2.10/32
# firewall-cmd --permanent --zone=trusted --add-port=3306/tcp
# firewall-cmd  --reload
```

НАСТРАИВАЕМ MYSQL СЕРВЕР

1. Скопируйте сертификаты с сервера Zabbix/CA в созданную для них директорию:

```
# scp root@192.168.3.92:/etc/pki/CA/private/pdb_nopass_key.pem /var/lib/mysql/pki/server.key
# scp root@192.168.3.92:/etc/pki/CA/certs/pdb_crt.pem /var/lib/mysql/pki/server.crt
# scp root@192.168.3.92:/etc/pki/CA/cecert.pem /var/lib/mysql/pki/ca.crt
```

2. Обновите права доступа:

```
# chown -R mysql. /var/lib/mysql/pki
```

3. Обновите конфигурационный файл MySQL:

```
# vi /etc/my.cnf.d/mysql-server.cnf
```

4. Добавив строки:

```
[mysqld]
ssl-ca=/var/lib/mysql/pki/ca.crt
ssl-cert=/var/lib/mysql/pki/server.crt
ssl-key=/var/lib/mysql/pki/server.key
```

5. Перезапустите MySQL:

```
# systemctl restart mysqld
```

НАСТРАИВАЕМ MYSQL СЕРВЕР

1. Авторизуйтесь в MySQL:

```
# mysql -u root -p
```

2. Проверьте настройки:

```
mysql> show variables like '%ssl%';
```

```
+-------------------+-----------------------------+
| Variable_name     | Value                       |
+-------------------+-----------------------------+
| admin_ssl_ca      |                             |
| admin_ssl_capath  |                             |
| admin_ssl_cert    |                             |
| admin_ssl_cipher  |                             |
| admin_ssl_crl     |                             |
| admin_ssl_crlpath |                             |
| admin_ssl_key     |                             |
| have_openssl      | YES                         |
| have_ssl          | YES                         |
| mysqlx_ssl_ca     |                             |
| mysqlx_ssl_capath |                             |
| mysqlx_ssl_cert   |                             |
| mysqlx_ssl_cipher |                             |
| mysqlx_ssl_crl    |                             |
| mysqlx_ssl_crlpath|                             |
| mysqlx_ssl_key    |                             |
| ssl_ca            | /var/lib/mysql/pki/ca.crt   |
| ssl_capath        |                             |
| ssl_cert          | /var/lib/mysql/pki/server.crt |
| ssl_cipher        |                             |
| ssl_crl           |                             |
| ssl_crlpath       |                             |
| ssl_fips_mode     | OFF                         |
| ssl_key           | /var/lib/mysql/pki/server.key |
+-------------------+-----------------------------+
```

НАСТРАИВАЕМ MYSQL СЕРВЕР

1. Создайте пользователя Zabbix, для базы данных Zabbix:

```
mysql> create user zabbix@IP identified with mysql_native_password BY 'password’;
mysql> create database zabbix character set utf8 collate utf8_bin;
```

2. Ограничьте доступ к базе Zabbix до пользователя Zabbix:

```
mysql> grant all privileges on zabbix.* to zabbix@IP;
```

3. Ограничьте до обязательного использования SSL авторизации:

```
mysql> alter user ‘zabb'@'%' require ssl;
```

4. Проверьте применились ли настройки:

```
mysql> select user,host,ssl_type from mysql.user;
```

```
+------------------+-----------+----------+
| user             | host      | ssl_type |
+------------------+-----------+----------+
| redhat           | %         | ANY      |
| zabbix           | %         | ANY      |
| mysql.infoschema | localhost |          |
| mysql.session    | localhost |          |
| mysql.sys        | localhost |          |
| root             | localhost |          |
+------------------+-----------+----------+
6 rows in set (0.00 sec)
```

НАСТРАИВАЕМ ZABBIX СЕРВЕР

1. Установите репозиторий Zabbix на сервер Zabbix/CA:

```
# rpm -Uvh https://repo.zabbix.com/zabbix/5.0/rhel/8/x86_64/zabbix-release-5.0-1.el8.noarch.rpm
# dnf clean all
```

2. Установите Zabbix сервер, фронтенд и агента:

```
# dnf install zabbix-server-mysql zabbix-web-mysql zabbix-apache-conf zabbix-agent
```

3. На сервере базы данных выполните команду, чтобы скопировать файлы схемы:

```
# scp root@IP:/usr/share/doc/zabbix-server-mysql*/create.sql.gz /home
```

4. На сервере базы данных импортируйте схему с данным по умолчанию

```
# zcat /home/create.sql.gz | mysql -uroot -p zabbix
```

5. Убедитесь, что все таблицы на месте:

```
mysql> use zabbix;
mysql> show tables;
166 rows in set (0.00 sec)
```

НАСТРАИВАЕМ ZABBIX СЕРВЕР

1. Создайте для Zabbix сервера SSL директорию:

```
# mkdir –p /var/lib/zabbix/ssl/
```

2. Скопируйте файлы сертификатов:

```
# cp /etc/pki/CA/cacert.pem /var/lib/zabbix/ssl/
# cp /etc/pki/CA/certs/zaca_crt.pem /var/lib/zabbix/ssl/server.zabbix.crt
# cp /etc/pki/CA/private/zaca_nopass_key.pem /var/lib/zabbix/ssl/server.zabbix.key
```

3. Обновите права доступа:

```
# chown -R zabbix /var/lib/zabbix/ssl/
# chmod 400 /var/lib/zabbix/ssl/cacert.pem
# chmod 400 /var/lib/zabbix/ssl/server.zabbix.crt
# chmod 400 /var/lib/zabbix/ssl/server.zabbix.key
```

НАСТРАИВАЕМ ZABBIX СЕРВЕР

1. Обновите содержимое конфигурационного файлы Zabbix сервера:

```
# vi /etc/zabbix/zabbix_server.conf
```

2. Обновите настройки доступа к базе данных:

```
DBHost=mydb.localhost.local
DBName=zabbix
DBUser=zabbix
DBPassword=zabbix
```

3. Обновите параметры которые относятся к SSL:

```
DBTLSConnect=verify_full
DBTLSCAFile=/var/lib/zabbix/ssl/cacert.pem
DBTLSCertFile=/var/lib/zabbix/ssl/server.zabbix.crt
DBTLSKeyFile=/var/lib/zabbix/ssl/server.zabbix.key
```

НАСТРАИВАЕМ ФРОНТЕНД ZABBIX

1. Создайте директорию для Apache SSL на Zabbix/CA server:

```
# mkdir -p /etc/httpd/ssl/
```

2. Скопируйте файлы сертификатов:

```
# cp /etc/pki/CA/cacert.pem /etc/httpd/ssl/
# cp /etc/pki/CA/certs/zfca_crt.pem /etc/httpd/ssl/server.crt
# cp /etc/pki/CA/private/zfca_nopass_key.pem /etc/httpd/ssl/server.key
```

3. Обновите права доступа:

```
# chown -R apache /etc/httpd/ssl/
# chmod 400 /etc/httpd/ssl/cacert.pem
# chmod 400 /etc/httpd/ssl/server.crt
# chmod 400 /etc/httpd/ssl/server.key
```

4. Настройте временную зону:

```
vi /etc/php-fpm.d/zabbix.conf
php_value[date.timezone] = Europe/Riga
```

5. Запустите процессы Zabbix:

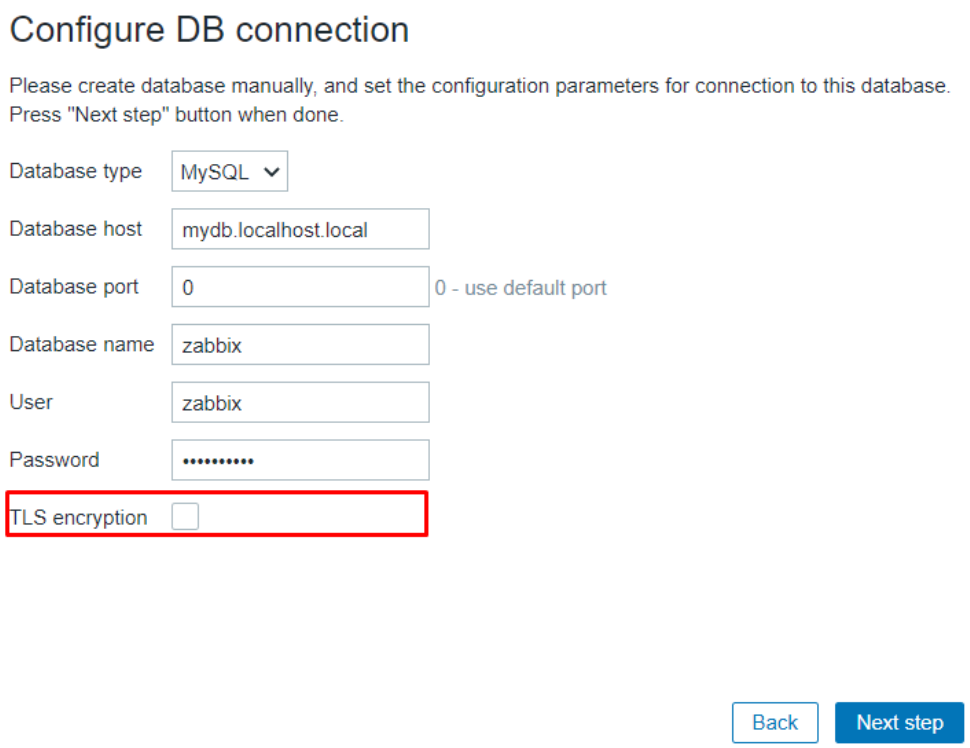
```
# systemctl restart zabbix-server zabbix-agent httpd php-fpm
# systemctl enable zabbix-server zabbix-agent httpd php-fpm
```

НАСТРАИВАЕМ ФРОНТЕНД ZABBIX

1. Перейти на адрес вашего фронтеда

```
# http://IP/zabbix
```

2. Добавьте имя базы, пользователя и пароль,подключите TLS шифрование:

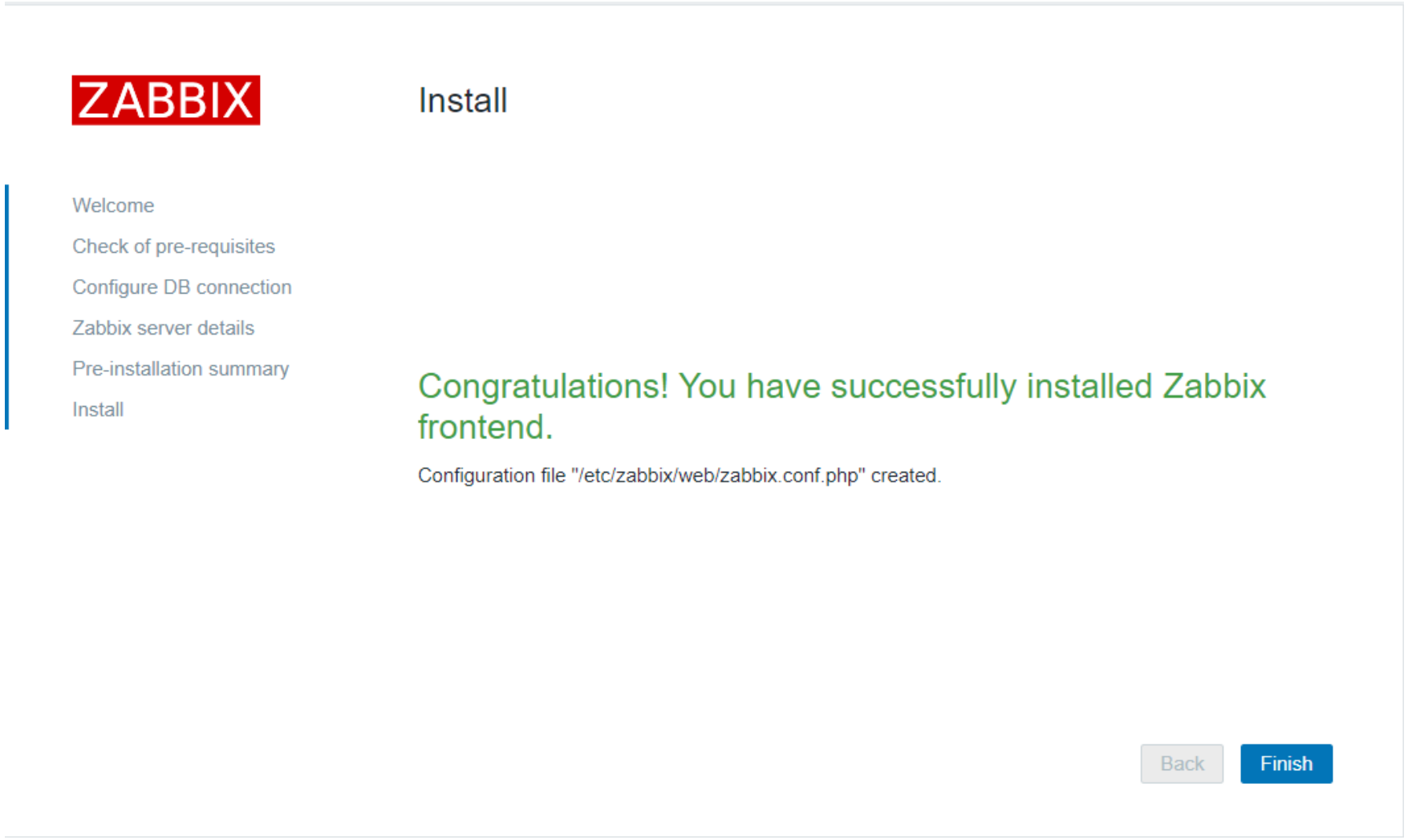


НАСТРАИВАЕМ ФРОНТЕНД ZABBIX

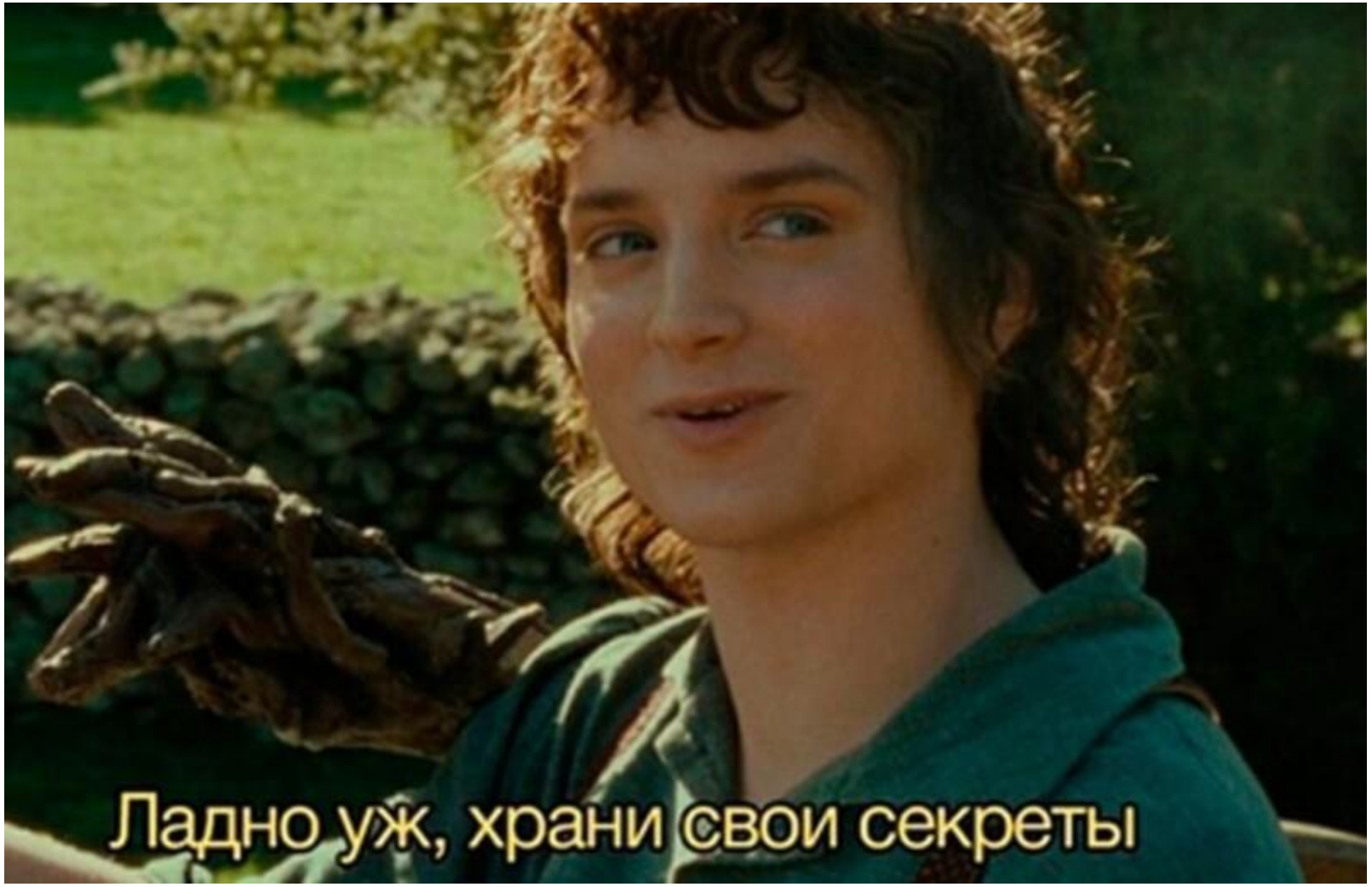
1. Добавьте путь к файлам сертификата:

```
/etc/httpd/ssl/cacert.pem
/etc/httpd/ssl/server.crt
/etc/httpd/ssl/server.key
```

ФИНИШИРУЙТЕ



ПОЗДРАВЛЯЕМ!



Спасибо!