



ELTEX-SC

Установка платформы

Версия платформы – 1.18

Содержание

Установка и настройка сервера Eltex SC	3
Ubuntu Server 16	3
Ubuntu Server 18	6
Проверка работоспособности платформы	16
Резервное копирование данных	16
Чеклист	17
Настройка почтовых e-mail и sms оповещений eltex-notification-gw(eltex-ngw) при работе с ELTEX-SC.....	20
Описание	20
Способы запуска/остановки	20
Конфигурация.....	21
/etc/eltex-ngw/notification.properties.....	21
/etc/default/eltex-ngw.....	23
Переход с http на https.....	24
Шаги, необходимые для перехода на https	24
Справка по сертификатам	24
Способ определения типа сертификата	24
Справка.....	25

Установка и настройка сервера Eltex SC

Система «Eltex SC» строится по клиент-серверной архитектуре. Серверную часть рекомендуется устанавливать на многопроцессорный компьютер под управлением OS Ubuntu 16.

Производительность сервера зависит от числа пользователей, которые будут зарегистрированы на платформе. Рекомендуемые характеристики сервера для определенного числа абонентов приведены в таблице ниже.

Число абонентов	до 5 000	до 10 000	до 10 000	до 20 000
Число аппаратных серверов	1	1	2	2
CPU, ГГц	i5 3,0	i7 4,0	i5+i5 3,0	i7+i7 4,0
RAM, Гб	8	16	8+8	16+16
HDD, Гб	1000	2000	1000+2000	1000+4000

Для развертывания сервера Eltex SC необходимо скачать и установить пакеты платформы в указанном порядке:

Ubuntu Server 16

1. Установить программное обеспечение **software-properties-common**

Установка

```
sudo apt-get install -y software-properties-common
```

2. Установить программное обеспечение **openjdk-8-jre**

Установка

```
sudo apt-get install -y openjdk-8-jre
```

3. Установить программное обеспечение **gnupg**

Установка

```
sudo apt-get install -y gnupg
```

4. Скачать ключ "server-4.0.asc" утилитой **wget****Установка**

```
wget https://www.mongodb.org/static/pgp/server-4.0.asc
```

5. Применить ключ "server-4.0.asc"

Установка

```
sudo apt-key add server-4.0.asc
```

6. Создать в текстовом редакторе файл

Установка

```
/etc/apt/sources.list.d/mongodb-org-4.0.list
```

7. Записать строчку в файл, сохранить файл

Установка

```
deb [ arch=amd64 ] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/4.0  
multiverse
```

8. Обновить пакеты в системе

Установка

```
sudo apt-get update
```

9. Установить пакет **mongodb-org****Установка**

```
sudo apt-get install -y mongodb-org
```

10. Запустить сервис **mongod****Установка**

```
sudo service mongod start
```

11. Установить программное обеспечение **nginx****Установка**

```
sudo apt-get install -y nginx
```

Ubuntu Server 18

1. Установить программное обеспечение **software-properties-common**

Установка

```
sudo apt-get install -y software-properties-common
```

2. Установить программное обеспечение **openjdk-8-jre**

Установка

```
sudo apt-get install -y openjdk-8-jre
```

3. Установить программное обеспечение **gnupg**

Установка

```
sudo apt-get install -y gnupg
```

4. Скачать ключ "server-4.0.asc" утилитой **wget**

Установка

```
wget https://www.mongodb.org/static/pgp/server-4.0.asc
```

5. Применить ключ "server-4.0.asc"

Установка

```
sudo apt-key add server-4.0.asc
```

6. Создать в текстовом редакторе файл

Установка

```
/etc/apt/sources.list.d/mongodb-org-4.0.list
```

7. Записать строчку в файл, сохранить файл

Установка

```
deb [ arch=amd64 ] https://repo.mongodb.org/apt/ubuntu bionic/mongodb-org/4.0  
multiverse
```

8. Обновить пакеты в системе

Установка

```
sudo apt-get update
```

9. Установить пакет **mongodb-org****Установка**

```
sudo apt-get install -y mongodb-org
```

10. Запустить сервис **mongod****Установка**

```
sudo service mongod start
```

11. Установить программное обеспечение **nginx**

Установка

```
sudo apt-get install -y nginx
```


Скачать и установить пакеты платформы в указанном порядке:

 Получение пакетов осуществляется через коммерческий отдел.

1. Пакет `eltex-notification-gw_1.x.xx_all.deb` (вместо `xx` указывается актуальная версия)

Установка

```
sudo dpkg -i eltex-notification-gw_1.x.xx_all.deb
```

2. Пакет `eltex-sc_1.18-xxx_all.deb` (вместо `xxx` указывается актуальная версия)

Установка

```
sudo dpkg -i eltex-sc_1.18-xxx_all.deb
```

3. Пакет `eltex-sc-web_1.18-xxx_all.deb` (вместо `xxx` указывается актуальная версия)

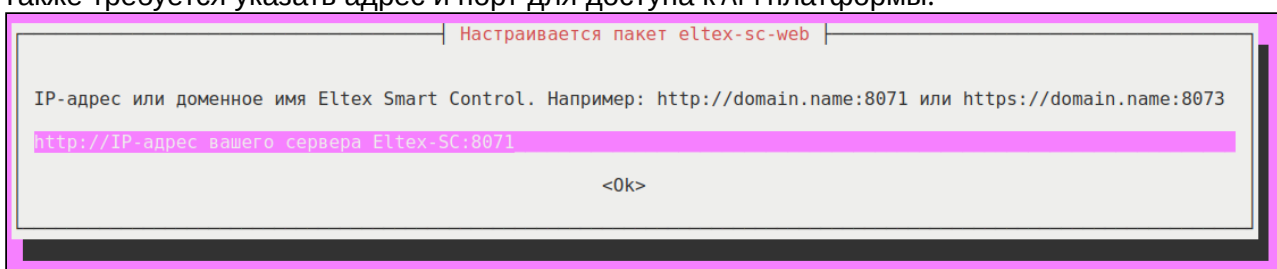
Установка

```
sudo dpkg -i eltex-sc-web_1.18-xxx_all.deb
```

В процессе установки пакета `eltex-sc-web_1.18-xxx_all.deb` потребуется указать порт для доступа по `http` и `https` (по умолчанию 8074 и 8075):



Также требуется указать адрес и порт для доступа к API платформы:



⚠ Здесь и далее подразумевается, что адрес платформы может быть как IP-адресом, так и полным доменным именем.

По окончании установки будет запущен демон `eltex-sc`. Проверить его состояние можно командой:

```
service eltex-sc status
```

Результат должен быть следующим:

```
● eltex-sc.service - IoT Server
  Loaded: loaded (/usr/lib/systemd/system/eltex-sc.service; disabled; vendor preset: enabled)
  Active: active (running) since Cp 2020-05-06 11:11:00 +07; 1s ago
    Main PID: 12914 (java)
      Tasks: 15
     Memory: 64.1M
        CPU: 547ms
    CGroup: /system.slice/eltex-sc.service
            └─12914 /usr/bin/java -Dfile.encoding=UTF-8 -Diot.config=/etc/eltex-sc/server.yml -Diot.
```

Включение сервисов `eltex-sc` и `mongod` в автозагрузку:

```
systemctl enable eltex-sc.service
systemctl enable mongod.service
```

Логи сервера находятся в `/var/log/eltex-sc/server.log`

Конфигурация сервера: `/etc/eltex-sc/server.yml`

Содержимое конфигурационного файла:

```
testData:
  environment: false

fileStorage:
  path: /var/lib/eltex-sc/files

logger:
  mongodb: ERROR
  springdata: INFO

controllerGateTCP:
  port: 8069

controllerGate:
  port: 8070

controllerGateSecurity:
  port: 8072

api:
  port: 8071
  sslPort: 8073

ui:
  port: 80

mqttBroker:
  enabled: false
  host: localhost
  port: 1883

defaultUrl: http://localhost

electricMeterScheduler:
  cron: 0 0/30 * * * ?
```

```
electricMeterArchiveScheduler:
  cron: 0 0 17 * * ?

services:
  ngw:
    host: localhost
    port: 8040

  push:
    serverKey: <уникальное значение>

dictionary:
  path: dict.json

languageInterface:
  language: ru

hz:
  instance:
    name: iot-core-hz-instance
  group:
    name: iot-core
    password: '11111'
  network:
    port: '5705'

mjollnir:
  sync-period: 0 0 23 1/1 * ?
  login: platform
  password: platform
  url: http://smart.eltex-co.ru:8078/api/v1

iot-core:
  host: localhost
  port: 27017
  user: ''
  password: ''
  database: iot-core

file-storage:
  host: localhost
  port: 27017
  user: ''
  password: ''
  database: iot-fs
```

```
license-storage:
  host: localhost
  port: 27017
  user: ''
  password: ''
  database: iot-licenses

eventlog:
  host: localhost
  port: 27017
  user: ''
  password: ''
  database: iot-events

monitoring-system:
  countAttempt: 3
  delayAttempt: 0

kafka:
  bootstrap-servers: localhost:9092
  enabled: false

video:
  flussonic:
    ip: <IP-адрес:порт>
  ivideon:
    client_id: <уникальное значение>
    client_secret: <уникальное значение>
    ip_for_catch_event: <уникальное значение>

ssl:
  key: /etc/ssl/private/eltex-sc-api.key
  crt: /etc/ssl/certs/eltex-sc-api.crt

proxy:
  enabled: false
  host: ''
  port: 8050

billing:
  cron: 0 0 0 * * ?
  ftp:
    host: <IP-адрес/доменное имя>
    port: <порт>
    login: <уникальное значение>
    password: <уникальное значение>
    workdir: <папка выгрузки>

notifications:
  duplicate_delay_sec: 3600
  antispam_time_sec: 5
```

```

delay_push_time_millsec: 250

guard:
  deviceRequestDelay: 7000

swagger:
  enable: false
  key: ''

```

Параметры конфигурационного файла:

- controllerGate port — порт сервера, к которому подключаются контроллеры, по умолчанию 8070;
- controllerGateSecurity port — порт сервера, к которому подключаются контроллеры с использованием сертификата SSL, по умолчанию 8072;
- api port — порт сервера для взаимодействия с api платформы через веб-интерфейс или мобильное приложение;
- api sslPort — порт сервера для взаимодействия с api платформы через веб-интерфейс или мобильное приложение с использованием SSL сертификата;
- ui — порт указываемый при формировании ссылок в письмах оповещений и т.д., например для функционала **Восстановление пароля**;
- defaultUrl — адрес платформы при формировании ссылок в письмах оповещений и т.д., например для функционала **Восстановление пароля**;
- services — настройки внешних сервисов для работы с платформой;
- logger — уровень логирования сервера. Для включения режима debug выставить у mongodb: DEBUG;
- electricMeterScheduler — параметр, отвечающий за автоматический опрос прибора платформой (опрос выполняется каждые 30 минут, начиная от начала часа по умолчанию). Пример: cron: 30/1 * * * ? — опрос каждые 30 минут;
- electricMeterArchiveScheduler — параметр, отвечающий за автоматический опрос архива прибора платформой;
- mqttBroker — настройка сервиса MQTT;
- video — настройка видеосервиса Flussonic;
- proxy — настройка прокси сервера;
- billing — настройка сервиса биллинга;
- guard — настройка задержки перед постановкой на охрану;
- swagger — сервис для взаимодействия и просмотра api платформы ELTEX-SC. Для доступа необходимо обратиться к :8071/api/v1/swagger-ui.html#/admin-logger-api-controller.
- notifications — настройка защиты сервиса push-уведомлений.

Параметры:

- **duplicate_delay_sec** — период времени, в течение которого будет блокироваться дубликат сообщения (полностью совпадающий) — Секунды;
- **antispam_time_sec** — период времени, в течение которого будет блокироваться отправка последующих однотипных сообщений (не полностью совпадающих, контроллер вне доступа 1, 2 и тд.) — Секунды;
- **delay_push_time_millisec** — задержка между отложенными push-уведомлениями — Миллисекунды.

Порты, используемые платформой **по умолчанию**:

- **8069** — порт для подключения контроллеров Ethernet в режиме TCP-клиент к платформе;
- **8070** — WS-порт для подключения контроллеров к платформе;
- **8071** — HTTP-порт API-платформы;
- **8072** — WSS-порт для подключения контроллеров к платформе;
- **8073** — HTTPS-порт API-платформы.

По окончании установки и конфигурирования сервер Eltex SC готов к работе. Взаимодействие с платформой происходит через веб-интерфейсы пользователей и администратора, а также через мобильное приложение.

Во всех случаях, чтобы войти в платформу, нужно указать ее адрес и номер порта. В общем случае в адресную строку браузера вводится следующее: **<Адрес сервера Eltex-SC>:8074**.

Проверка работоспособности платформы


Резервное копирование данных

Резервные копии баз данных (далее БД) расположены в следующих директориях:

- /var/backups/mongobackups/iot-core — основная БД платформы;
- /var/backups/mongobackups/iot-fs — БД файлообменника;
- /var/backups/mongobackups/iot-events — БД журнала событий пользователей и устройств;
- /var/backups/mongobackups/iot-licenses — БД лицензий.

Резервное копирование данных выполняется примерно в 4–5 утра каждый день и автоматически архивируется в файл формата "<имя БД>_YYYY-MM-DD_НН-ММ.tar.gz", где YYYY-MM-DD — дата, НН-ММ — время, а параметр "имя БД" соответствует имени БД, копирование которой выполняется (iot-core, iot-fs, iot-events, iot-licenses).

Одна самая свежая копия каждой базы также находится в соответствующей папке.

 Платформа автоматически удаляет архивы старше 7 суток.

При необходимости скрипт для создания резервных копий может быть запущен вручную из терминала с помощью командной оболочки:

```
sh /etc/cron.daily/iot-backupdb
```

Для восстановления любой базы данных из резервной копии требуется:

1. Распаковать архив с БД;
2. Остановить сервис eltex-sc, если он запущен:

```
sudo service eltex-sc stop
```

3. Выполнить команду:

```
sudo mongorestore -d <имя БД> <путь до самой последней директории из архива>
```

4. Запустить сервис eltex-sc:

```
sudo service eltex-sc start
```

Директория хранения журналов работы платформы: **/var/log/eltex-sc/**

Файлы разбиваются на части по размеру. Каждая часть сохраняется в файл с именем: server-YYYY-MM-DD.NN.log, где YYYY-MM-DD — дата, а NN — номер части.

Чеклист

Проверить занятость ядер процессора и оперативной памяти

```
htop
```

Следить за *MEM%* и *CPU%* процессов mongo и eltex-scl!

Проверить занятость физического места на диске

```
df -h
```

Обратить внимание на использование диска всеми подразделами /dev и на примонтированное сетевое хранилище для резервных копий data.eltex.loc.

Проверить загруженность коллекций в БД

```
mongotop
```

Время записи и чтения в/из коллекций должно стремиться к 0.

Проверить соединения контроллеров с платформой

```
netstat -na | grep 8070
```

Параметры, на которые следует обратить внимание:

- Состояние соединений:
 - **ESTABLISHED, LISTEN** — норма;
 - **LAST_ACK** — связь с контроллером уже потеряна, рядом должен быть переоткрытый коннект с того же IP;
 - **TIME_WAIT, CLOSE_WAIT** — соединение “повисло”, неудачное закрытие;
- **2-я** колонка — очередь принятых сервером пакетов;
- **3-я** колонка — очередь пакетов, отосланных на контроллеры;
- **5-я** колонка — IP-адреса контроллеров.

Необходимо следить, чтобы пакеты не копились в очереди на контроллер. Если такое происходит, значит веб-сокеты этого контроллера недоступны, и нужно посмотреть в логах платформы, что случилось и как давно. ID контроллера можно найти в БД по его IP-адресу (поле **controllerIpAddress**) либо непосредственно в логах.

Проверить наличие ошибок в логе

Логи платформы находятся в разделе `/var/log/eltex-sc*`. Они разбиваются на части по размеру. Чтобы проверить все журналы на наличие некоторого параметра за определенную дату (например, за 15 января 2020 года) и записать результаты в файл, нужно выполнить команду:

```
grep <значение для поиска> server-2020-01-15* > <имя файла для вывода>
```

Если требуется, можно дописать следующие опции после значения для поиска:

- **-i** — не чувствителен к регистру;
- **-n** — номера строк;
- **-h** — подавляет запись имени файла перед каждой строкой в результирующем файле;
- **-A** — количество строк после совпадения;
- **-B** — количество строк до совпадения.

Общий вид команды:

```
grep -i -n -A 5 -B 2 error server-2020-01-15* > errors.log
```

Ключевые слова для поиска:

- *ERROR*;
- *ID* или *IP* контроллера;
- *PONG* — ключевое слово, которое пишется, если контроллер не отвечает.

Опрос счетчиков

Опрос счетчиков происходит в соответствии с указанными в файле `server.yml` временными параметрами опроса. Примерная продолжительность 35 секунд.

Для определения времени опроса нужно найти в логах начало опроса первого счетчика и конец опроса последнего по сообщениям вида:

- Start executeMessage method for [deviceType] device id = [deviceId];
- End executeMessage method for [deviceType] device id = [deviceId].

Проверка веб-интерфейса

Самый простой способ проверить доступность веб-интерфейса платформы — открыть его в браузере (по IP или доменному имени — в зависимости от ваших настроек сети и сервера nginx).

Можно проверить доступность API платформы. Для этого в адресной строке браузера к URL веб-интерфейса платформы нужно дописать:

```
<Адрес сервера Eltex-SC>:<порт API>/api/v1/version
```

где порт API — вышеуказанный порт доступа к API по HTTP (по умолчанию **8071**). Платформа должна моментально вернуть данные об установленной версии платформы, версии API и текущем времени на сервере в формате JSON.

Проверка запущенных процессов

Для проверки работоспособности платформы из терминала можно воспользоваться командой:

```
htop -u eltexsc или top -u eltexsc
```

В результате будет отображена информация о java-процессе платформы.

Для отображения информации о сервисе платформы нужно ввести команду:

```
service eltex-sc status
```

В результате будет отображена информация о сервисе платформы: статус — запущен или нет, какие файлы конфигурации используются, с какими ограничениями запущена JVM.

Для старта платформы требуется уже запущенная база данных MongoDB. Проверить, что она запущена, можно через терминал:

```
htop -u mongodb или $ top -u mongodb
```

В результате будет отображена информация о процессе MongoDB.

Для отображения информации о сервисе базы данных нужно ввести команду:

```
service mongod status
```

В результате будет отображена информация о сервисе базы данных: статус — запущен или нет, и какие файлы конфигурации используются.

Настройка почтовых e-mail и sms оповещений eltex-notification-gw(eltex-ngw) при работе с ELTEX-SC

Описание

Сервис представляет собой интерфейс для взаимодействия компонентов платформы и внешними SMS-шлюзами, Call-центрами и серверами электронной почты. Служит для отправки уведомлений пользователям платформы, а также Wi-Fi пользователям.

Устанавливается при помощи пакета **eltex-ngw**.

Способы запуска/остановки

Для остановки сервиса используется команда:

```
service eltex-ngw stop
```

Для запуска сервиса после остановки используется команда:

```
service eltex-ngw start
```

Для проверки того, запущен ли сервис в данный момент или нет, используется команда:

```
service eltex-ngw status
```

В ответ, если сервис запущен, последует сообщение:

```
eltex-ngw start/running, process <pid>
```

В случае, если сервис не запущен:

```
eltex-ngw stop/waiting
```

Конфигурация

/etc/eltex-ngw/notification.properties

Основной конфигурационный файл содержит параметры:

- Ссылка на конфигурационный файл для подключения к SMS-шлюзу (для отправки SMS пользователям):

```
sms.gate.outgoing.sms.config=smc_gate.conf
```

- Ссылка на конфигурационный файл для идентификации пользователя по отправленным им SMS:

```
sms.gate.incoming.sms.config=smpp_gate.conf
```

- Ссылка на конфигурационный файл для идентификации пользователя по исходящим от него звонкам:

```
sms.gate.incoming.call.config=rtk_sbc.conf
```

- Настройка размера очереди SMS на отправку и время ожидания отправки:

```
#Gate pool settings  
sms.gate.pool.size=50  
sms.gate.pool.wait.millis=5000
```

- Настройка размера очереди и времени ожидания на отправку и ссылка на конфигурационный файл для авторизации по входящему звонку:

```
#Call gate configuration  
call.gate.outgoing.call.config=smg_gate.conf  
call.gate.pool.size=50  
pool.wait.millis=5000
```

- Порт, прослушиваемый сервисом для приема запросов на отправку сообщений:

```
#Port to listen for requests
server.port=8040
```

- Настройка подключения к MongoDB для хранения журнала отправки сообщений:

```
#=====
#=====database settings=====
#=====
#mongodb.uri=mongodb://192.168.1.1,192.168.1.2:27017/notification-gw
mongodb.host=localhost
mongodb.port=27017
#mongodb.user=user
#mongodb.password=password
mongodb.name=notification-gw
mongodb.fsfiles.store.period=7
```

- Подключение к серверу электронной почты:

```
#=====
#=====email settings=====
#=====
mail.smtp.submitter=example@eltex.loc
mail.smtp.password=password
mail.smtp.auth=true
mail.smtp.host=eltex.loc
mail.smtp.port=587
mail.smtp.sendpartial=true
mail.smtp.connectiontimeout=5000
mail.gate.pool.size=20
mail.pool.wait.millis=10000
```

Примечание

mail.smtp.password=password

Если у вас в пароле используются спецсимволы типа: "[\ ^ \$. | ? * + ()", тогда необходимо их экранировать, иначе пароль будет неверным.

- Пример 1: Пароль — "326P%1E\" при экранировании он должен выглядеть так — "326P%1E\\";
- Пример 2: Пароль — "326P%1E^" при экранировании он должен выглядеть так — "326P%1E\^".

❗ Для включения шифрования соединения с почтовым сервером нужно указать опцию:

```
mail.smtp.starttls.enable=true
```

/etc/default/eltex-ngw

Содержит параметры запуска приложения:

```
JAVA_INIT_HEAP=16m
# Maximum size of Java heap
JAVA_MAX_HEAP=128m

LOG_LEVEL=info

# Additional arguments to pass to java
JAVA_OPTS="-XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/var/log/eltex-
ngw"
```

- JAVA_INIT_HEAP — количество памяти, выделяемое приложению при старте сервиса;
- JAVA_MAX_HEAP — максимальное количество памяти, которое может арендовать приложение;
- LOG_LEVEL — уровень логирования: error, warning, info, debug;
- JAVA_OPTS — дополнительные опции запуска java.

Переход с http на https

По умолчанию платформа для https использует следующие порты:

- 8073 — для доступа к API;
- 8075 — для доступа к WEB.

Шаги, необходимые для перехода на https

Переместите свои файлы сертификатов в папку `private` и `certs` (или измените путь в конфигурационных файлах).

Проверить наличие прав к файлам и папкам. Необходимые права назначаются с помощью команды `chmod 640`.

Конфигурационный файл `eltex-sc`: `/etc/eltex-sc/server.yml`

```
ssl:
  key: /etc/ssl/private/eltex-sc-api.key;
  crt: /etc/ssl/certs/eltex-sc-api.crt.
```

Конфигурационный файл `eltex-sc-web`: `/etc/nginx/sites-available/eltex-sc-web`

```
ssl_certificate /etc/ssl/certs/eltex-sc-web.crt;
ssl_certificate_key /etc/ssl/private/eltex-sc-web.key.
```

Справка по сертификатам

Ваш сертификат должен быть формате PKCS#8

Пример конвертирования сертификата из PKCS#1 в PKCS#8, `letsencrypt`:

```
openssl pkcs8 -topk8 -inform PEM -outform PEM -nocrypt -in путь_до_ключа -out
имя_для_нового_ключа.
```

Способ определения типа сертификата

```
# head -1 /tmp/eltex-sc-api.key
-----BEGIN PRIVATE KEY-----

# head -1 /etc/ssl/private/eltex-sc-api.key
-----BEGIN RSA PRIVATE KEY-----
```

BEGIN PRIVATE KEY — это кодировка PKCS#8.

BEGIN RSA PRIVATE KEY — это кодировка PKCS#1.

Справка

Проблема: После перезагрузки сервера, на котором развернута платформа ELTEX-SC, при попытке авторизации сервер возвращает следующие сообщения:



Извините, сервис временно недоступен!

Пытаемся возобновить соединение

Лог файл: /var/log/eltex-sc/server.log

```
2020-07-20 05:11:58,110 WARN ClassPathXmlApplicationContext - Exception encountered
during context initialization - cancelling refresh attempt:
org.springframework.beans.factory.BeanCreationException: Error creating bean with
name 'checkMigrationsService' defined in class path resource [global-spring-
context.xml]: Bean instantiation via constructor failed; nested exception is
org.springframework.beans.BeanInstantiationException: Failed to instantiate
[org.eltex.iotcore.server.services.CheckMigrationsService]: Constructor threw
exception; nested exception is com.mongodb.MongoTimeoutException: Timed out after
30000 ms while waiting to connect. Client view of cluster state is {type=UNKNOWN,
servers=[{address=localhost:27017, type=UNKNOWN, state=CONNECTING,
exception={com.mongodb.MongoSocketOpenException: Exception opening socket}, caused by
{java.net.ConnectException: Connection refused (Connection refused)}}]
2020-07-20 05:12:04,472 DEBUG logging - Logging Provider:
org.jboss.logging.Slf4jLoggerProvider
2020-07-20 05:12:04,490 INFO Version - HV000001: Hibernate Validator 5.0.1.Final
```

Решение:

Не включен сервис mongod, необходимо запустить сервис следующей командой:

```
service mongod start
```