



Платформа  
**Eltex SC. АСД**

Установка платформы  
Версия: 1.17.2

## Содержание

1	Установка сервера Eltex SC. АСД.....	3
1.1	Установка зависимостей.....	3
1.1.1	Установка зависимостей на ОС Ubuntu 20.....	3
1.1.2	Установка зависимостей на ОС Ubuntu 18.....	5
1.1.3	Установка зависимостей на ОС Ubuntu 16.....	7
1.2	Установка пакетов Eltex.....	9
2	Конфигурирование сервера Eltex SC. АСД.....	11
3	Проверка работоспособности платформы .....	15
4	Настройка сервиса eltex-notification-gw (eltex-ngw) .....	18
4.1	Остановка и запуск сервиса.....	18
4.2	Конфигурирование сервиса .....	19
4.2.1	/etc/eltex-ngw/notification.properties .....	19
4.2.2	/etc/default/eltex-ngw.....	21
5	Резервное копирование данных.....	22
6	Справка.....	23

## 1 Установка сервера Eltex SC. АСД


**Платформа Eltex SC. АСД** – это программно-аппаратный комплекс по автоматизированному сбору показаний с приборов учета ресурсов тепла, электроэнергии, горячего и холодного водоснабжения. Платформа позволяет обеспечивать контроль и анализ потребляемых ресурсов с индивидуальных приборов учета на объекте. Мониторинг аварийных ситуаций осуществляется с помощью автоматических оповещений по SMS, e-mail. Информация о показаниях приборов учета доступна в приложении «Умные счетчики».

Система Eltex SC. АСД строится по клиент-серверной архитектуре. Серверную часть рекомендуется устанавливать на многопроцессорный компьютер под управлением OS Ubuntu 20.04.

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


Число абонентов	до 5 000	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. АСД необходимо установить зависимости, а затем скачать и установить пакеты платформы в порядке, приведенном в данном руководстве.

 Установка Eltex SC. АСД на сервер, не отвечающий минимальным системным требованиям, ведет к неработоспособности платформы!

### 1.1 Установка зависимостей

В данном разделе описан процесс установки зависимостей на многопроцессорный компьютер под управлением OS Ubuntu 20.04 (рекомендуется).

 Если на вашем компьютере установлена версия OS Ubuntu 18 или Ubuntu 16, пропустите следующие шаги и выполните установку зависимостей согласно разделам [Установка зависимостей на OS Ubuntu 18](#) или [Установка зависимостей на OS Ubuntu 16](#) соответственно.

#### 1.1.1 Установка зависимостей на OS Ubuntu 20

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

##### Установка

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

## 2. Установите программное обеспечение **openjdk-8-jre**

### Установка

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

## 3. Установите программное обеспечение **gnupg**

### Установка

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

## 4. Скачайте ключ "server-4.0.asc" утилитой **wget**

### Установка

```
wget https://www.mongodb.org/static/pgp/server-4.4.asc
```

## 5. Примените ключ "server-4.4.asc"

### Установка

```
sudo apt-key add server-4.4.asc
```

## 6. Создайте в текстовом редакторе файл

### Установка

```
/etc/apt/sources.list.d/mongodb-org-4.4.list
```

## 7. Запишите в файл следующую строку, сохраните файл

### Установка

```
deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse
```

## 8. Обновите пакеты в системе

### Установка

```
sudo apt-get update
```

## 9. Установите пакет **mongodb-org**

### Установка

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

## 10. Запустите сервис **mongod**

### Установка

```
sudo service mongod start
```

## 11. Установите программное обеспечение **nginx**

### Установка

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

### 1.1.2 Установка зависимостей на ОС Ubuntu 18

#### 1. Установите программное обеспечение **software-properties-common**

### Установка

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

#### 2. Установите программное обеспечение **openjdk-8-jre**

### Установка

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

### 3. Установите программное обеспечение **gnupg**

**Установка**

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

### 4. Скачайте ключ "server-4.0.asc" утилитой **wget**

**Установка**

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

### 5. Примените ключ "server-4.0.asc"

**Установка**

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

### 6. Создайте в текстовом редакторе файл

**Установка**

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

### 7. Запишите в файл следующую строку, сохраните файл

**Установка**

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

### 8. Обновите пакеты в системе

**Установка**

```
sudo apt-get update
```

## 9. Установите пакет **mongodb-org**

### Установка

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

## 10. Запустите сервис **mongod**

### Установка

```
sudo service mongod start
```

## 11. Установите программное обеспечение **nginx**

### Установка

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

### 1.1.3 Установка зависимостей на ОС Ubuntu 16

#### 1. Установите программное обеспечение **software-properties-common**

### Установка

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

#### 2. Установите программное обеспечение **openjdk-8-jre**

### Установка

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

#### 3. Установите программное обеспечение **gnupg**

### Установка

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

#### 4. Скачайте ключ "server-4.0.asc" утилитой **wget**

##### Установка

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

#### 5. Примените ключ "server-4.0.asc"

##### Установка

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

#### 6. Создайте в текстовом редакторе файл

##### Установка

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

#### 7. Запишите в файл следующую строку, сохраните файл

##### Установка

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

#### 8. Обновите пакеты в системе

##### Установка

```
sudo apt-get update
```

#### 9. Установите пакет **mongodb-org**

##### Установка

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



## 10. Запустите сервис **mongod**

### Установка

```
sudo service mongod start
```


## 11. Установите программное обеспечение **nginx**

### Установка

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

## 1.2 Установка пакетов Eltex

Скачайте и установите пакеты платформы в указанном порядке:

 Получение пакетов осуществляется через Коммерческий отдел.

### 1. Пакет `eltex-notification-gw_X.X.X_all.deb` (вместо X.X.X указывается актуальная версия)

#### Установка

```
sudo dpkg -i eltex-notification-gw_X.X.X_all.deb
```

### 2. Пакет `eltex-sc_1.17-xxx_all.deb` (вместо xxx указывается актуальная версия)

#### Установка

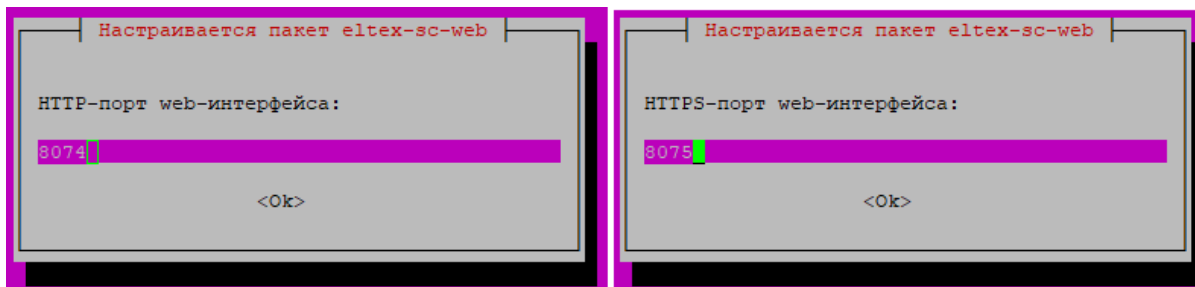
```
sudo dpkg -i eltex-sc_1.17-xxx_all.deb
```

### 3. Пакет `eltex-sc-web_1.17-xxx_all.deb` (вместо xxx указывается актуальная версия)

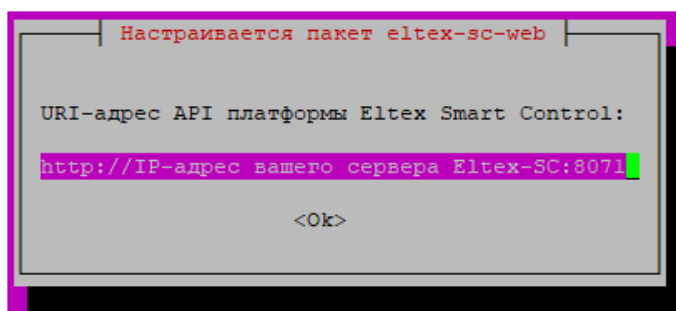
#### Установка

```
sudo dpkg -i eltex-sc-web_1.17-xxx_all.deb
```

В процессе установки пакета `eltex-sc-web_1.17-xxx_all.deb` потребуется указать порты для доступа по `http` и `https` (по умолчанию 8074 и 8075):



Также потребуется указать адрес и порт для доступа к API платформы (по умолчанию 8071) в формате : `http://[IP-адрес вашего сервера Eltex-SC]:порт`



- ✓ Адрес сервера может быть как IP-адресом, так и полным доменным именем.

По окончании установки будет запущен сервис (фоновая программа) `eltex-sc`. Проверить его состояние можно командой:

```
service eltex-sc status
```

Результат выполнения команды 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.
```

## 2 Конфигурирование сервера Eltex SC. АСД

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

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

Логи сервера: `/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

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>
```

**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, указываемый 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 необходимо установить mongodb в значение DEBUG.
- electricMeterScheduler – параметр, отвечающий за автоматический опрос прибора платформой (опрос выполняется каждые 30 минут, начиная от начала часа по умолчанию). Пример: cron: 3 0/1 \* \* \* ? – опрос каждые 3 минуты.

**Порты, используемые платформой по умолчанию:**

- **8069** – порт для подключения контроллеров Ethernet к платформе в режиме TCP-клиент;
- **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. АСД 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**.

### 3 Проверка работоспособности платформы

В данном разделе представлен чек-лист для проверки работоспособности платформы.

#### Проверить занятость ядер процессора и оперативной памяти

```
htop
```

При проверке следите за *MEM%* и *CPU%* процессов *mongo* и *eltex-sc*.

#### Проверить занятость физического места на диске

```
df -h
```

Обратите 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 потеряна, рядом должно быть переоткрытое соединение с того же IP;
  - **TIME\_WAIT, CLOSE\_WAIT** — соединение приостановлено, 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 проверить событие и время события в лог-файлах платформы. ID контроллера можно найти в БД по его IP-адресу (поле **controllerIpAddress**) либо непосредственно в лог-файлах.

## Проверить наличие ошибок в лог-файле

Лог-файлы 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 лог-файле время начала опроса первого счетчика и время конца опроса последнего по сообщениям вида:

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

## Проверка web-интерфейса

Самый простой способ проверить доступность web-интерфейса платформы — открыть его в браузере (по IP-адресу 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 web-интерфейса платформы нужно дописать:

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

где порт API — указанный при настройке eltex-sc-web\_1.17-xxx\_all.deb 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 eltex-sc 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.

## 4 Настройка сервиса eltex-notification-gw (eltex-ngw)

Сервис eltex-notification-gw представляет собой интерфейс для отправки e-mail и sms-оповещений пользователям платформы. Eltex-notification-gw обеспечивает взаимодействие компонентов платформы с внешними SMS-шлюзами, call-центрами и серверами электронной почты.

Сервис устанавливается при помощи пакета **eltex-ngw**.

### 4.1 Остановка и запуск сервиса

После установки пакета сервис запускается по умолчанию. 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, используется команда:

```
service eltex-ngw status
```

Если сервис запущен, в ответ последует следующее сообщение:

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

Если сервис не запущен:

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

## 4.2 Конфигурирование сервиса

Сервис настраивается путем редактирования следующих конфигурационных файлов:

### 4.2.1 /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, а также 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
```

#### ❗ mail.smtp.password=password

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

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

- Для включения шифрования соединения с почтовым сервером укажите опцию:

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

#### 4.2.2 /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.

## 5 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\_NN-MM.tar.gz", где YYYY-MM-DD — дата, NN-MM — 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.

## 6 Справка

**Проблема:** после перезагрузки сервера, на котором развернута платформа Eltex SC. АСД, при попытке авторизации сервер возвращает следующее сообщение:



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 его следующей командой:

```
service mongod start
```