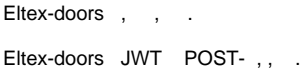


```

•
•
•
  • /
  • API
    •
    •
    •
•
  • /etc/eltex-doors/application.conf
  • /etc/default/eltex-doors
  • /etc/eltex-doors/log4j2.xml

```



```

    eltex-doors, auth_token. , (token.expire.timeout), . Mysql eltex-doors-db.
    " {iat}" " {exp}" payload. .
1.15 doors. , /etc/eltex-doors/keys/.
, , 440 eltex:

```

```
-r--r----- 1 root eltex 1704 Sep 19 10:44 private.pem
-r--r----- 1 root eltex  451 Sep 19 10:44 public.pem
```

private.pem - JWT-  
public.pem - JWT-  
- , , /usr/share/eltex-doors/gen\_keys.sh

 , public.pem

JWT Doors, : access refresh. Access token . Refresh token access token. access token.  
Eltex Doors 15 access 30 refresh.  
access token Eltex Doors access token, refresh token.





```
:
```

```
apt install eltex-doors-db
```

```
:
```

```
apt install eltex-doors
```

/

	service eltex-doors status	<div><div> Active: active (running)</div><div>C</div><div> Active: failed</div></div>
	service eltex-doors start	<div><div> Active: active (running)</div></div>
	service eltex-doors stop	<div><div> Active: failed</div></div>

	service eltex-doors restart	<div>  Active: active (running) </div>
--	-----------------------------	---

## API

Eltex Doors :

- JWT (refresh token access token) .
- access token refresh token.
- JWT.
- JWT. .

access refresh token Eltex Doors POST- JSON URL /api/signin :

```
POST http://localhost:9097/api/signin HTTP
/1.1
Content-Type: application/json
Accept: application/json
{
  "username": "user",
  "password": "password",
  "metadata": {}
}
```

- "username" .
- "password" .
- "metadata" , JWT.

 metadata SoftWLC 1.17.

:

```
curl -XPOST -H "Content-type: application/json" -d '{"username": "user", "password": "password"}'
'http://localhost:9097/api/signin/'
```

- , :

```
{
  "status": "OK",
  "accessToken": "eyJhbGc...",
  "refreshToken": "eyJhb..."
}
```

"status"- ;  
"accessToken"- .

- , :

```
{"status":"FAIL","error":"Invalid username
or password"}
```

"status"- ;  
"error" - .

refresh token . URL /api/signout, .

```
POST http://localhost:9097/api/signout HTTP/1.1
Accept: application/json
Authorization: Bearer eyJhb...
```

:

```
curl -XPOST -H "Authorization: Bearer eyJhb..." 'http://localhost:9097/api/signout/'
```

- , :

```
{
  "status": "OK",
  "accessToken": "Logged out",
  "refreshToken": null
}
```

- , :

```
{
  "status": "FAIL",
  "error": "Invalid JWT"
}
```

access token /api/signin/token :

```
POST http://localhost:9097/api/signin/token HTTP/1.1
Accept: application/json
Authorization: Bearer eyJhb...
```

:

```
curl -XPOST -H "Authorization: Bearer eyJhbGc..." 'http://100.110.1.222:9097/api/signin/token/'
```

- , :

```
{
  "status": "OK",
  "accessToken": "eyJhbGc...",
  "refreshToken": "eyJhb..."
}
```

- , :

```
{
  "status": "FAIL",
  "error": "Invalid JWT"
}
```

## /etc/eltex-doors/application.conf

### /etc/eltex-doors/application.conf

```
# Mysql datasource properties
database {
  host = localhost
  port = 3306
  name = eltex_doors
  user = eltexdoors
  password = eltexpassword

  pool {
    # Time to wait for a connection
    connectionTimeout = 10s
    # Time to wait for connection validation
    validationTimeout = 3s

    min = 1
    max = 5
  }
}

# Basic account credentials
auth {
  username = user
  password = password
}

# Authentication keys paths
keys {
  private = /etc/eltex-doors/keys/private.pem
  public = /etc/eltex-doors/keys/public.pem
}

# Tokens lifetime
tokens {
  clearTimeout = "**/60 * * * * *"

  lifetime {
    access = 15m
    refresh = 30d
  }
}
```

• :

```
database {
  host = localhost
  port = 3306
  name = eltex_doors
  user = eltexdoors
  password = eltexpassword
...

```

• :

```
connectionTimeout = 10s
```

- :

```
validationTimeout = 3s
```

- :

```
min = 1
max = 5
```

- auth-service:

```
auth {
  username = user
  password = password
}
```

- :

```
keys {
  private = /etc/eltex-doors/keys/private.pem
  public = /etc/eltex-doors/keys/public.pem
}
```

- :

```
clearTimeout = "**/60 * * * * *
```

- :

```
lifetime {
  access = 15m
  refresh = 30d
}
```

## /etc/default/eltex-doors

.

### /etc/default/eltex-doors

```
# AP Monitoring service

# HTTP Port for use by Eltex Doors
PORT=9097

# Initial size of Java heap
JAVA_INIT_HEAP=32m
# Maximum size of Java heap
JAVA_MAX_HEAP=256m

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

- :

```
# HTTP Port for use by Eltex Doors
PORT=9097
```

- , :

```
# Initial size of Java heap
JAVA_INIT_HEAP=32m
```

- , :

```
# Maximum size of Java heap
JAVA_MAX_HEAP=256m
```

- java:

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

## /etc/eltex-doors/log4j2.xml

### /etc/eltex-doors/log4j2.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<Configuration monitorInterval="600">
  <Properties>
    <Property name="rootLevel">DEBUG</Property>

    <Property name="baseDir">/var/log/eltex-doors</Property>
    <Property name="maxFileSize">5 MB</Property>
    <Property name="accumulatedFileSize">100 MB</Property>
    <Property name="lastModified">7d</Property>
    <Property name="maxCount">10</Property>
    <Property name="logPattern">%d{ISO8601} [%t] %-5p %logger{1} %C{1}.%M(line:%L). %m%n</Property>

    <Property name="consoleLevel">${env:CONSOLE_LEVEL:-OFF}</Property>

    <Property name="gelfLevel">${env:GELF_LEVEL:-OFF}</Property>
    <Property name="gelfHost">${env:GELF_HOST:-udp:lab3-test.eltex.loc}</Property>
    <Property name="gelfPort">${env:GELF_PORT:-12201}</Property>
  </Properties>

  <Appenders>

    <Console name="STDERR" target="System.err">
      <PatternLayout pattern="${logPattern}" />
    </Console>

    <RollingFile name="RollingFile"
      fileName="${baseDir}/doors.log"
      filePattern="${baseDir}/log/doors-%i.log.gz">
      <PatternLayout pattern="${logPattern}" />
      <Policies>
        <SizeBasedTriggeringPolicy size="${maxFileSize}" />
        <OnStartupTriggeringPolicy />
      </Policies>
      <DefaultRolloverStrategy max="${maxCount}">
        <Delete basePath="${baseDir}" maxDepth="3">
          <IfFileName glob="*/doors-*.log.gz">
```

```

        <IfAny>
            <IfAccumulatedFileSize exceeds="${accumulatedFileSize}"/>
            <IfLastModified age="${lastModified}"/>
        </IfAny>
    </IfFileName>
</Delete>
</DefaultRolloverStrategy>
</RollingFile>

<Gelf name="Gelf" host="${gelfHost}" port="${gelfPort}" version="1.1" facility="eltex-doors"
    extractStackTrace="true" originHost="%host{fqdn}" maximumMessageSize="8192">
    <Field name="thread" pattern="%t"/>
    <Field name="level" pattern="%level"/>
    <Field name="severity" pattern="%-5level"/>
    <Field name="logger" pattern="%logger{1}"/>
    <Field name="location" pattern="%C{1}.%M(line:%L)"/>
</Gelf>
</Appenders>

<Loggers>
    <Root level="${rootLevel}">
        <AppenderRef ref="RollingFile"/>
        <AppenderRef ref="STDERR" level="${consoleLevel}"/>
        <AppenderRef ref="Gelf" level="${gelfLevel}"/>
    </Root>

    <Logger name="org.eltex.softwlc.doors" level="INFO"/>
    <Logger name="org.springframework" level="ERROR"/>
    <Logger name="org.eclipse.jetty" level="ERROR"/>
    <Logger name="com.zaxxer" level="ERROR"/>

</Loggers>
</Configuration>

```

- :

```
<Configuration monitorInterval="600">
```

- :

```
<Property name="baseDir">/var/log/eltex-doors</Property>
```

- ( , ,):

```
<Property name="maxFileSize">5 MB</Property>
```

- ( +). :

```
<Property name="accumulatedFileSize">100 MB</Property>
```

- , , :

```
<Property name="lastModified">7d</Property>
```

- , , :

```
<Property name="maxCount">10</Property>
```

- Graylog ( , ,):



```
<Property name="gelfLevel">OFF</Property>
<Property name="gelfHost">udp:lab3-test.eltex.loc</Property>
<Property name="gelfPort">12201</Property>
```

- :

```
<Property name="rootLevel">DEBUG</Property>
```