



ОБЩЕСТВО
С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

129085, г. Москва,
ул. Годовикова, д. 9, стр. 17.
+7 (495) 784-70-00
ineed@bftcom.com, www.bftcom.com

ИНСТРУКЦИЯ ПО УСТАНОВКЕ И НАСТРОЙКЕ ЭКЗЕМПЛЯРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ЗАПУСКУ СИСТЕМЫ

БФТ.УПРАВЛЕНИЕ АКТИВАМИ

На 23 листах

2025

СОДЕРЖАНИЕ

Установка системы «БФТ.Управление активами»	3
Установка программы	3
Установка комплекта инструментов сервера приложений	3
Установка СУБД	4
Установка сервера приложений	6
Установка дополнений	12
Создание базы данных приложения	13
Установка сервера геоданных	14
Настройка компонентов	16
Установка и настройка облачного хранилища вложений S3	18
Настройка программы на условия конкретного применения	20
Настройка программы на условия конкретного применения	20
Настройка программы на условия конкретного применения для комплекса интеграции	21
Запуск Системы хранения и предоставления электронных документов («БФТ.Управление активами»)	22

Установка системы «БФТ.Управление активами»

Установка программы

Ниже приведено описание действий по установке программы для операционной системы Linux.

Операции в ОС следует обязательно выполнять под пользователем суперюзер (напр. Administrator)

Последовательность установки программы для операционной системы Linux (с использованием интерфейса командной строки):

Порядок установки на примере системы Linux Ubuntu 20.04

****Важно!**** Обязательным требованием является наличие в операционной системе русской локали. Перед установкой следует убедиться, что в операционной системе она присутствует.

Перед началом установки окружения необходимо в обязательном порядке перейти в папку «Пакеты для предустановки» в предоставленном наборе дистрибутивов и установить все пакеты оттуда.

Либо установку окружения, настройку и запуск приложения производить на системе, установленной из приложенного образа в наборе дистрибутивов.

Установка комплекта инструментов сервера приложений

Порядок действий по установке комплекта инструментов сервера приложений на примере OpenJDK.17:

1. Перейти в директорию /usr/lib

```
cd /usr/lib
```

2. Создать там каталог jvm

```
sudo mkdir jvm
```

3. Из предоставленного набора дистрибутивов скопировать архив

openjdk-17.0.13_linux-x64_bin.tar в директорию /usr/lib/jvm

```
sudo cp jdk-17.0.13_linux-x64_bin.tar.gz /usr/lib/jvm/
```

4. Перейти в каталог /usr/lib/jvm и распаковать архив

```
cd /usr/lib/jvm
```

```
sudo tar -xvf jdk-17.0.13_linux-x64_bin.tar.gz
```

5. Убедиться, что распаковка прошла успешно и удалить архив

```
sudo rm jdk-17.0.13_linux-x64_bin.tar.gz
```

6. Далее необходимо установить переменную \$JAVA_HOME и добавить путь к java в переменную \$PATH. Для этого следует открыть файл /etc/profile в любом редакторе, например

```
sudo gedit /etc/profile
```

и прописать их ниже:

```
export JAVA_HOME=/usr/lib/jvm/jdk-17.0.13
```

```
export PATH=$PATH:$JAVA_HOME/bin
```

7. Далее следует перезагрузить систему и проверить, что java 17 установлена

```
java --version
```

должна быть выведена версия java:

```
java 17.0.13
```

Установка СУБД

Порядок действий по установке СУБД на примере PostgreSQL 14.17:

1. Для установки PostgreSQL 14 из исходников потребуются следующие пакеты, доступные из ОС: gcc make, libreadline-dev, zlibc zlib1g-dev — для их установки следует воспользоваться командой apt install

```
apt install gcc make
```

```
apt install libreadline-dev
```

```
apt install zlibc zlib1g-dev
```

2. Создать на жестком диске папку, скопировать туда и распаковать архив postgresql-14.17.tar.gz из набора предоставленных дистрибутивов

```
mkdir pg
```

```
cp postgresql-14.17.tar.gz ~/pg
```

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

```
cd ~/pg
```

```
tar xvf postgresql-14.17.tar.gz
```

3. Для установки СУБД последовательно выполнить:

```
sudo ./configure
```

```
sudo make
```

```
sudo make install
```

4. Создать каталог для хранения данных

```
sudo mkdir /usr/local/pgsql/data
```

5. Создать пользователя postgres. Во время создания система попросит назначить пароль и прочие пользовательские данные

```
sudo adduser postgres
```

6. Сделать его владельцем каталога data и установить права

```
chown postgres /usr/local/pgsql/data/
```

```
chmod 750 /usr/local/pgsql/data/
```

7. Добавить в /etc/profile переменную PGDATA и путь к PostgreSQL

```
export PGDATA=/usr/local/pgsql/data
```

```
export PATH=$PATH:/usr/local/pgsql/bin
```

8. Зайти под пользователем postgres

```
su - postgres
```

9. Проинициализировать кластер баз данных

```
initdb -k
```

10. Изменить аутентификацию по паролю:

```
sudo gedit /usr/local/pgsql/data/pg_hba.conf
```

В блоке:

```
host    all    all    127.0.0.1/32    ident
host    all    all    ::1/128         ident
```

Заменить данные на:

```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

Установка сервера приложений

Порядок установки сервера приложений на примере на примере Tomcat 9.0.93+ (для обозначения каталога установки любой копии Tomcat будет использоваться <ТМС>):

1. Создать каталог

```
sudo mkdir /opt/_Tomcat
```

2. Создать группу tomcat

```
sudo groupadd tomcat
```

3. Создать пользователя **tomcat**

```
sudo useradd -M -s /sbin/nologin -g tomcat -d /opt/_Tomcat
tomcat
```

4. В каталоге /opt/_Tomcat создать каталог application1-8080 (указание порта в названии каталога, по которому будет работать приложение предусмотрено специально — это поможет при отладке и в дальнейшей работе с сервером)

```
sudo mkdir /opt/_Tomcat/application1-8080
```

5. Из предоставленного набора дистрибутивов скопировать и распаковать архив apache-tomcat-9.0.100.tar.gz в каталог /opt/_Tomcat/application1-8080:

```
sudo tar xvf apache-tomcat-9.0.100.tar.gz -C
/opt/_Tomcat/application1-8080 --strip-components=1
```

6. Установить права доступа:

```
sudo chown -R tomcat:tomcat /opt/_Tomcat
sudo find /opt/_Tomcat/* -type d -exec chmod 770 {} \;
sudo find /opt/_Tomcat/* -type f -name *.sh -exec chmod 770 {}
\;
```

7. Создать каталоги для журналов

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

```
sudo mkdir -p /var/log/tomcat/application1-8080/archiv
sudo rmdir /opt/_Tomcat/application1-8080/logs
sudo ln -s /var/log/tomcat/logs
sudo ln -s /var/log/tomcat/application1-8080
/opt/_Tomcat/application1-8080/logs
sudo chmod -R 770 /var/log/tomcat
sudo chown -R tomcat:tomcat /var/log/tomcat
sudo restorecon -Rv /var/log/tomcat
```

8. Создать скрипт для архивации журналов (архивирует все журналы за определённую дату и сохраняет архивы в каталоге <ТМС>/logs/archiv в течение недели):

```
sudo mkdir -p /root/_Scripts/
sudo touch /root/_Scripts/arch_log_date-tomcat.sh
```

и привести его к виду:

```
#!/bin/bash

mkdir -p $1/archiv
find $1/*$(date --date '-1 day' +%Y-%m-%d)*.* -exec tar -r -f
$1/archiv/$(date --date '-1 day' +%Y-%m-%d).tar --remove-files
{} \; && find $1/archiv/*.tar -exec gzip {} \; && find
$1/archiv/*.
tar.gz -mtime +5 -delete

chown $2:$3 $1/archiv/*.tar.gz
chmod 640 $1/archiv/*.tar.gz
```

9. Для logrotate создать файл-конфигурацию ротации основного файла журнала Tomcat-а (при активной работе с приложениями в Tomcat он увеличивается очень быстро) /etc/logrotate.d/tomcat следующего содержания:

```
/var/log/tomcat/application1-8080/catalina.out
{
    rotate 10
    size 200M
    compress
    notifempty
    missingok
    copytruncate
    su tomcat tomcat
}
```

10. В файл конфигурации "планировщика" /etc/crontab добавить строки:

```
0 */12 * * * root logrotate --force /etc/logrotate.d/tomcat >
/dev/null 2>&1
0 1 * * * root /root/_Scripts/arch_log_date-tomcat.sh
/var/log/tomcat/application1-8080 tomcat tomcat > /dev/null
2>&1
```

По умолчанию в конфигурационных файлах Tomcat (Tomcat/opt/_Tomcat/application1-8080/conf/server.xml) в качестве основного порта доступа указан порт 8080.

11. При необходимости настройки работы по HTTPS необходимо добавить дополнительные настройки:

```
<Connector port="8443" SSLEnabled="true"
maxHttpHeaderSize="8192"
    maxThreads="150" minSpareThreads="25"
    maxSpareThreads="200"
    enableLookups="false" disableUploadTimeout="true"
    acceptCount="100" scheme="https" secure="true"
    clientAuth="false" sslProtocol="TLS"
    keystoreFile="/opt/_Tomcat/application1-
8080/conf/ssl/keystore.p12" keystorePass="*****"
    keystoreType="PKCS12"
    keyAlias="application1"/>
```

И создать сами ключи:

```
sudo mkdir -p /opt/_Tomcat/application1-8080/conf/ssl
sudo keytool -genkeypair -keyalg RSA -keysize 2048 -storetype
PKCS12 -keystore keystore.p12 -validity 3650 -alias application1
-file /opt/_Tomcat/application1-8080/conf/ssl/keystore.p12
```

12. В приложении предусмотрен механизм компрессии данных для ускорения их передачи по сети. Чтобы ее включить, в файле конфигурации сервера Tomcat/opt/_Tomcat/application2-8081/conf/server.xml указать настройки для HTTP:

```
# <Server port="8005"
<Server port="8006"
```

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

```
# <Connector port="8080" protocol="HTTP/1.1"
# connectionTimeout="20000"
# compressibleMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/javascript,application/json,application/xml"
# compression="on" compressionMinSize="8192" useSendfile="false"
# maxThreads="3800"
# maxSwallowSize="105906176"
# maxParameterCount="1000"
# redirectPort="8443" />
<Connector port="8081" protocol="HTTP/1.1"
    connectionTimeout="20000"
    compressibleMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/javascript,application/json,application/xml"
    compression="on" compressionMinSize="8192" useSendfile="false"
    maxThreads="3800"
    maxSwallowSize="105906176"
    maxParameterCount="1000"
    redirectPort="8444" />
### В последних сборках Tomcat 9.0 следующие строки отсутствуют
# <Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
<Connector port="8010" protocol="AJP/1.3" redirectPort="8444" />
```

ВАЖНО!!! Параметр `compressibleMimeType` до какой-то версии/сборки именовался `compressableMimeType` (сменилась одна буква: а → і), поэтому, если при запуске Tomcat в его журнале `catalina.out` появляются сообщения вида

```
24-Jan-2024      08:56:19.264      WARNING      [main]
org.apache.tomcat.util.digester.SetPropertiesRule.begin      Match
[Server/Service/Connector] failed to set property [compressableMimeType] to
```

[text/html,text/xml,text/plain,text/css,text/javascript,application/javascript,application/json,application/xml]

значит надо проверить написание данного параметра.

При необходимости настройки работы по HTTPS необходимо добавить дополнительные блок параметров:

```
### после
<Connector port="8080"....
### добавляем
### для Application1
<Connector port="8443" protocol="HTTP/1.1" scheme="https" secure="true"
  SSLEnabled="true" sslProtocol="TLSv1.2" clientAuth="false"
  SSLVerifyClient="optional"
  connectionTimeout="20000"
  compressibleMimeType="text/html,text/xml,text/plain,text/css,text/javascript,application/javascript,application/json,application/xml"
  compression="on" compressionMinSize="8192" useSendfile="false"
  maxThreads="3800"
  maxParameterCount="1000"
  maxSwallowSize="105906176"
# при использовании отдельных файлов сертификатов
  SSLCertificateFile="/opt/_Tomcat/application1-8080/conf/ssl/tls.crt"
  SSLCertificateKeyFile="/opt/_Tomcat/application1-8080/conf/ssl/tls.key"
# при использовании криптоконтейнеров
  keystoreFile="/opt/_Tomcat/application1-8080/conf/ssl/keystore.p12" keystorePass="*****" keystoreType="PKCS12"
  keyAlias="application1"
/>
```

13. В интерактивном режиме указать требуемые параметры и получить по окончании файл-хранилище с ключами (закрытым (секретным) и публичным). В файл <TMC>/conf/context.xml перед закрывающим тегом </Context> добавить строку:

```
...
  <Resources cachingAllowed="true" cacheMaxSize="100000"
  cacheTtl="2000" />
</Context>
```

14. Создать файлы systemd-юнитов для запуска Tomcat-ов в качестве сервиса. Для Application1 /etc/systemd/system/tomcat-application1-8080.service:

```
# Systemd unit file for Tomcat - Application1
[Unit]
Description=Apache Tomcat Web Application Container
After=syslog.target network.target

[Service]
Type=forking

###-- Каталог установки JDK
### OpenJDK - JRE
# Environment='JAVA_HOME=/usr/lib/jvm/jre'
Environment='JAVA_HOME=/usr/lib/jvm/jdk-17.0.13'
### OpenJDK - JRE
# Environment='JAVA_HOME=/usr/java/latest/jre'

Environment='CATALINA_PID=/opt/_Tomcat/application1-8080/temp/tomcat.pid'
Environment='CATALINA_HOME=/opt/_Tomcat/application1-8080'
Environment='CATALINA_BASE=/opt/_Tomcat/application1-8080'
Environment='CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'

WorkingDirectory=/opt/_Tomcat/application1-8080
ExecStart=/opt/_Tomcat/application1-8080/bin/startup.sh
ExecStop=/bin/kill -15 $MAINPID

User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target
```

15. Указать systemd, чтобы он прочитал новые юниты:

```
sudo systemctl daemon-reload
```

16. Запустить, проверить, поставить в автозапуск:

```
sudo systemctl enable tomcat-application1-8080.service  
sudo systemctl start tomcat-application1-8080.service  
sudo systemctl status tomcat-application1-8080.service
```

17. Если запуск Tomcat прошёл успешно, можно разворачивать приложение. При необходимости передавать какие-либо параметры в приложение, их необходимо предварительно прописывать в файл <ТМС>/conf/catalina.properties.

18. Для установки приложения необходимо выбрать файл app.war из предоставленного набора дистрибутивов и поместить его в папку /opt/_Tomcat/application1-8080/webapps

Установка дополнений

Порядок установки дополнений на примере PostGIS:

1. Для установки PostGIS потребуются дополнительно установленные пакеты: libxml2-dev, libgeos-dev, libproj-dev, libgdal-dev, postgres-server-dev-12, доступные из ОС по команде `sudo apt install`

```
sudo apt install libxml2-dev  
sudo apt install libgeos-dev  
sudo apt install libproj-dev  
sudo apt install libgdal-dev  
sudo apt install postgres-server-dev-12
```

2. Из предоставленного набора дистрибутивов нужно скопировать на жесткий диск и распаковать архив `postgis-3.0.12.tar.gz`

```
tar -xvf postgis-3.0.12.tar.gz
```

3. Перейти в распакованный каталог

```
cd postgis-3.0.12
```

4. Последовательно выполнить команды для установки PostGIS:

```
./configure  
make  
make install
```

5. Далее необходимо установить и настроить взаимодействие PostgreSQL и PostGIS для возможности использовать расширения при настройке баз данных. Из предоставленного набора необходимо установить на компьютер дистрибутив postgresql-14-postgis-3-scripts_3.2.1+dfsg-1.pgdg20.04+1_all.deb.

Создание базы данных приложения

1. Создать базу geo_saumi3 и выполнить на ней:

```
CREATE EXTENSION postgis;  
CREATE EXTENSION postgis_raster;  
CREATE EXTENSION postgis_sfcgal;  
CREATE EXTENSION address_standardizer;  
CREATE EXTENSION fuzzystrmatch;  
CREATE EXTENSION postgis_topology;  
CREATE EXTENSION postgis_tiger_geocoder;
```

Таблицы для слоев с геоданными в данной БД следует создавать через интерфейс гео-сервера. Структура таблиц должна соответствовать результату выполнения скрипта ниже, и в обязательном порядке содержать поля cad_num и geom.

```
CREATE TABLE [имя таблицы: layer1] (  
  
    id    BIGINT PRIMARY KEY,  
  
    cad_num VARCHAR,  
  
    geom  geometry  
  
);
```

2. Создать базу ice_esb

3. Создать базу данных saumi3 (в данном примере на ней будет развернуто приложение)

```
CREATE DATABASE saumi3;
```

4. Кроме того, для успешного восстановления БД из дампа, потребуются создать следующих пользователей и настроить одного из них

```
CREATE USER ice_user;
```

```
CREATE USER admin;
```

```
ALTER USER admin WITH SUPERUSER;
```

```
ALTER DATABASE saumi3 OWNER TO admin;
```

```
GRANT ALL PRIVILEGES ON DATABASE "saumi3" to admin;
```

5. Восстановить базу из дампа

```
pg_restore -d saumi3 dump_with_clean_config.tar.gz
```

Установка сервера геоданных

1. Из предоставленного набора дистрибутивов распаковать архив с GeoServer - geoserver-2.20.7-bin.zip - в папку /lib а затем распаковать архив geoserver-2.20.7-patches в папку /lib/geoserver-2.20.7

```
sudo unzip geoserver-2.20.7-bin.zip
```

2. Настроить GeoServer:

В файле /lib/geoserver-2.20.7/start.ini установить значение параметра

```
jetty.port=8585
```

В файле /lib/geoserver-2.20.7/webapps/geoserver/WEB-INF/web.xml раскомментировать следующие фильтры:

```
<filter>
```

```
<filter-name>cross-origin</filter-name>
```

```
<filter-  
class>org.eclipse.jetty.servlets.CrossOriginFilter</filter-class>
```

...

```
</filter>
```

```
<filter-mapping>
```

```
<filter-name>cross-origin</filter-name>
```

```
<url-pattern>/*</url-pattern>
```

```
</filter-mapping>
```

3. Запустить GeoServer из файла:

```
/lib/geoserver-2.20.7/bin/startup.sh
```

4. Авторизоваться

Пройти в браузере по ссылке:

```
http://{имя_хоста}:8585/geoserver
```

Логин: admin

Пароль: geoserver

5. Создать рабочую область:

Данные -> Рабочие области -> Добавить новую рабочую область

Name: drr

URI: com.bftcom.geoserver.drr

По умолчанию: true

6. Настроить хранилище:

Создать новое хранилище (Данные -> Хранилища -> Добавить новое хранилище -> PostGIS)

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

Указать созданную рабочую область drt, параметры базы данных geo_saum3, ввести название хранилища, нажать "Сохранить"

7. Настроить слои:

Создать слой (Данные -> Слои -> Добавить новый слой).
Указать созданное хранилище в качестве источника

Напротив нужной таблицы, соответствующей слою, нажать кнопку опубликовать. Данная таблица будет источником данных для слоя

В разделе "Охваты" нажать кнопку "Вычислить по данным" и "вычислить из родного охвата"

Нажать "Сохранить"

Настройка компонентов

Порядок настройки компонентов на примере сервера приложений Tomcat:

1. Для всех компонентов в указанный файл <TMC>/conf/catalina.properties надо добавить параметр. <TMC> - это домашний каталог конкретного экземпляра установленной копии Tomcat (в данном примере это /opt/_Tomcat/application1-8080). Добавить параметр:

```
ice.projectRoot=/opt/_Tomcat/application1-8080
```

2. Создать соответствующий каталог:

```
sudo mkdir -p <TMC>/ice
```

```
sudo chown tomcat:tomcat <TMC>/ice
```

```
sudo chmod 750 <TMC>/ice
```

3. Так же создать файл **/spring-shell.log** (его необходимость зависит от параметров, указанных в <TMC>/conf/catalina.properties):

```
sudo touch /spring-shell.log
```

```
sudo chown tomcat:tomcat /spring-shell.log
```

```
sudo chmod 750 /spring-shell.log
```

4. Открыть доступ к порту:

1.1.i. на межсетевом экране firewalld:

```
# открываем доступ к контейнерам и интерфейсу управления  
для указанной сети
```

```
# добавляем требуемые порты 8080
```

```
sudo firewall-cmd --permanent --new-service=tomcat-http
```

```
sudo firewall-cmd --permanent --service=tomcat-http --add-  
port=8080/tcp
```

```
sudo firewall-cmd --permanent --add-service=tomcat-http
```

```
sudo firewall-cmd --reload
```

```
### Далее - примеры (!) для настройки более "тонкой"  
настройки доступа:
```

```
sudo firewall-cmd --permanent --add-rich-rule 'rule family="ipv4"  
source address="192.168.1.0/24" service name="http" accept'
```

```
# открываем полный доступ с сервера мониторинга Zabbix
```

```
sudo firewall-cmd --permanent --add-rich-rule 'rule family="ipv4"  
source address="192.168.19.34/32" accept'
```

1.1.ii. в SELinux:

5. Проверить, что SELinux активен:

```
sudo sestatus
```

6. Если в ответе на эту команду в строке "SELinux status:" будет enabled - проверяем целевые порты и добавляем по необходимости:

```
# Посмотреть, открыты ли порты наших серверов Tomcat
```

```
sudo semanage port -l | grep -E "8080"
```

7. Так же проверяем, если ли блокировки со стороны SELinux:

```
sudo cat /var/log/audit/audit.log | grep denied | grep catalina.sh
```

8. Если в выводе будут строки, то создаём и применяем на основе их разрешающее правило:

```
sudo cat /var/log/audit/audit.log | grep denied | grep catalina.sh |  
audit2allow -M my_tomcat  
sudo semodule -i my_tomcat.pp
```

9. При случае, когда утилита semanage не установлена на сервере - установить её (и повторить затем предыдущие команды):

```
sudo yum install polycoreutils-python
```

10. Положить файл application.properties в папку с /opt/_Tomcat/application1-8080.

Важное замечание: файл application.properties подготавливается для каждой инсталляции приложения отдельно, его шаблон можно взять в исходниках проекта.

В приложении предусмотрен механизм компрессии данных для ускорения их передачи по сети, чтобы ее включить, следует добавить настройки:

```
server.compression.enabled=true  
server.compression.min-response-size=2048  
server.compression.mime-  
types=application/pdf,application/json,application/xml,text/html,  
application/javascript,text/css,text/xml,text/plain
```

11. Прописать полный путь до этого файла в /opt/_Tomcat/application1-8080/bin/startup.sh:

```
export CATALINA_OPTS="-  
Dspring.config.location=file:полный_путь_до_application.prop  
erties"
```

Установка и настройка облачного хранилища вложений S3

1) Подготовка:

```
wget https://dl.min.io/server/minio/release/linux-amd64/minio -O  
/usr/local/sbin/minio
```

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

```
chmod +x /usr/local/sbin/minio  
useradd -r minio -s /sbin/nologin -m -d /data/minio  
chown minio:minio /usr/local/sbin/minio  
  
mkdir -p /etc/minio  
chown minio:minio /etc/minio  
mkdir -p /data/minio  
chown minio:minio /data/minio
```

2) конфиг сервиса:

* ключи сгенерировать самостоятельно

```
cat >/etc/default/minio <<EOF  
MINIO_VOLUMES="/data/minio"  
MINIO_OPTS="-C /etc/minio --address 0.0.0.0:9000"  
MINIO_ACCESS_KEY="УКАЗЫВАЕМ_ACCESS_KEY"  
MINIO_SECRET_KEY="УКАЗЫВАЕМ_SECRET_KEY"  
EOF
```

3) конфиг-файл «демона» (сервиса):

```
cat >/etc/systemd/system/minio.service <<EOF  
[Unit]  
Description=MinIO Server  
Documentation=https://docs.min.io  
Wants=network-online.target  
After=network-online.target  
AssertFileIsExecutable=/usr/local/sbin/minio  
[Service]  
WorkingDirectory=/data/minio/  
User=minio  
Group=minio  
EnvironmentFile=/etc/default/minio  
ExecStart=/usr/local/sbin/minio server $MINIO_OPTS  
$MINIO_VOLUMES  
Restart=always  
LimitNOFILE=65536  
TimeoutStopSec=infinity  
SendSIGKILL=no  
[Install]  
WantedBy=multi-user.target  
EOF
```

4) запуск сервиса:

```
systemctl daemon-reload
systemctl enable minio
systemctl start minio
```

Для корректного сохранения вложений в облачном сервисе необходимо настроить взаимодействие с ним в основном приложении:

- В разделе «Настройки» -> «Хранилища вложений» необходимо добавить новое хранилище, в настройках которого указать значение «Облачное хранилище S3» в поле «Тип хранилища», заполнить обязательные поля, убедиться, что флаг «Включен» установлен, а также добавить сущности, вложения из которых должны быть сохранены в хранилище.

Создание новой записи: Хранилище вложений

* Идентификатор:

* Отображаемое имя:

* Тип хранилища: Облачное хранилище S3

* Поведение при отсутствии связи: Сохранение в БД без уведомления

Включен:

* Endpoint:

* Bucket:

Region:

* Access Key ID:

* Secret Key:

Отменить Сохранить

Настройка программы на условия конкретного применения

Настройка программы на условия конкретного применения

Настройка программы на условиях конкретного применения для комплексов:

1 Управление активами.

Требуется выполнить следующие настройки для начала работы с компонентом:

- восстановить базу данных для приложения из дампа

1. установить war-архив приложения на сервер приложения Apache Tomcat.
2. настроить файл `application.properties`:
 - настройки подключения к базе;
 - настройки подключения к сервису интеграции (группа параметров `prg.integration.*`);
 - настройки подключения к сервису конвертации (параметр `ice.converter.url`).

Инициализация базы приложения происходит автоматически при первом старте.

Настройка программы на условия конкретного применения для комплекса интеграции

Требуется выполнить следующие настройки для начала работы с компонентом:

- восстановить базу данных для приложения из дампа
 - установить war-архив приложения на сервер приложения Apache Tomcat.
 - настроить файл `catalina.properties` (см. содержимое ниже):
 - настройки подключения к базе
 - настроить файл `application.properties`:
 - настройки подключения к базе;
 - настройки подключения к очереди `jsm`.
 - установить расширение PostGis для базы данных PostgreSQL
 - установить сервер геоданных ГеоСервер
 - настроить хранилище и слои для ГеоСервер

Инициализация базы приложения происходит автоматически при первом старте.

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

Для настройки программы на условия конкретного применения выполняются следующие действия:

- Настроить файл `catalina.properties`:

Параметры подключения

```
spring.profiles.active=postgresql
```

```
spring.datasource.url=jdbc:postgresql://ServerName:5432/db_name
```

```
spring.datasource.username=*****
```

```
spring.datasource.password=*****
```

```
spring.shell.interactive.enabled=false
```

- настроить файл `logback.xml`:
- настройка необходимости журналирования обмена:
`logger name="Camellog" level="OFF" или "INFO"`
- настройки электронной подписи
- настройка через пользовательский интерфейс используемых слоев сервера геоданных и привязка слоев к объектам приложения
- настроить разделе Информационные системы подключение к модулю Подсистема информационного взаимодействия. Указать способ использования адаптера - JMS
- настроить подпись (ЭП-ОВ)

Запуск Системы хранения и предоставления электронных документов («БФТ.Управление активами»)

Работа в платформе доступна только для зарегистрированных пользователей.

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«БЮДЖЕТНЫЕ И ФИНАНСОВЫЕ ТЕХНОЛОГИИ»

Для перехода к окну авторизации в строке адреса браузера вводится адрес сервера системы. Откроется окно авторизации пользователя:

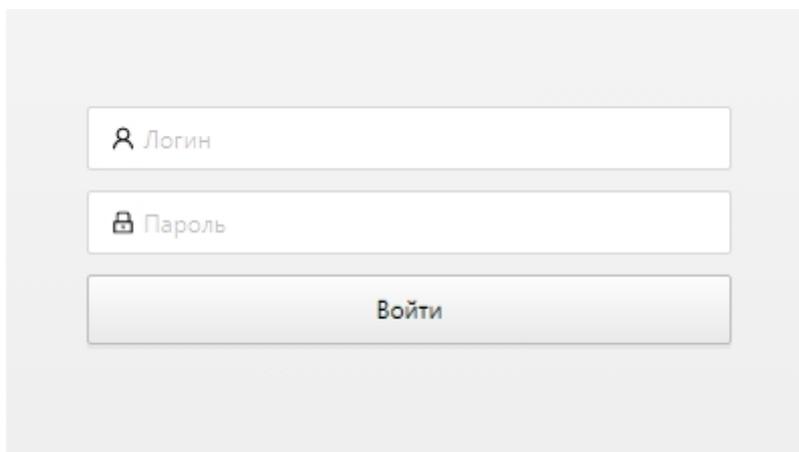


Рисунок 1 – Окно входа в систему

Для входа в систему необходимо указать следующую информацию (Рисунок 1):

- **Логин** - имя пользователя (Administrator).
- **Пароль** - пароль пользователя (Administrator).

Далее нажимается кнопка **Войти**.

В случае ввода верной информации (указаны зарегистрированный в системе пользователь и правильный пароль) осуществится вход в главное окно системы. При вводе неверного **Пользователя** или **Пароля** на экране появится сообщение об ошибке «**Неправильное имя пользователя или пароль**».