

# Инструкция по установке подсистемы работы с централизованными договорами.

На 21 листе.  
Версия 1.1.



## Содержание

<b>1</b>	<b>ОБЩИЕ ПОЛОЖЕНИЯ .....</b>	<b>4</b>
1.1	ПОЛНОЕ НАИМЕНОВАНИЕ СИСТЕМЫ И ЕЕ УСЛОВНОЕ ОБОЗНАЧЕНИЕ .....	4
1.2	ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ .....	4
<b>2</b>	<b>НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ .....</b>	<b>5</b>
2.1	НАЗНАЧЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ .....	5
2.2	УРОВЕНЬ ПОДГОТОВКИ ПЕРСОНАЛА.....	6
<b>3</b>	<b>АРХИТЕКТУРНЫЕ РЕШЕНИЯ .....</b>	<b>9</b>
3.1	АРХИТЕКТУРНЫЕ РЕШЕНИЯ СИСТЕМЫ .....	9
3.2	ПОРЯДОК ЗАГРУЗКИ ДАННЫХ И ПРОГРАММ.....	10
<b>4</b>	<b>УСТАНОВКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ .....</b>	<b>11</b>
4.1	Установка программного обеспечения подсистемы работы с централизованными договорами .....	11
4.1.1	Установка операционной системы и общесистемных компонентов 11	
4.1.2	Установка OpenJDK 11 .....	11
4.1.2.1	Установка OpenJDK 11 .....	11
4.1.2.2	Настройка Java .....	11
4.1.2.3	Переменная JAVA_HOME .....	11
4.2	УСТАНОВКА ТОМСАТ 8 .....	12
4.2.1	Создание пользователя Tomcat .....	12
4.2.2	Установка Tomcat .....	12
4.2.3	Настройка прав.....	13
4.2.4	Создание файла systemd.....	13
4.2.5	Настройка Tomcat .....	14
4.3	УСТАНОВКА СУБД POSTGRESQL 9.5+ .....	15
4.3.1	Установка PostgreSQL .....	15
4.3.2	Создание базы данных в PostgreSQL.....	16
4.3.3	Открытие внешнего доступа к серверу .....	16
4.4	УСТАНОВКА И ОБНОВЛЕНИЕ ПРИЛОЖЕНИЯ.....	17
4.4.1	Обновление приложения.....	17
4.4.2	Обновление базы данных.....	18
4.4.3	Запуск новой версии приложения.....	18
4.5	НАСТРОЙКА ПРИЛОЖЕНИЯ .....	19
4.5.1	Настройки соединения с базой данных .....	19
4.5.2	Параметры развертывания.....	19
4.5.3	Настройка свойств безопасности .....	20
4.5.4	Настройка параметров подключения к системе оператора ЭДО	20

4.5.5	Настройка расписания выполнения фоновых задач .....	20
4.5.6	Настройка подключения к хранилищу для файлов.....	20
4.5.7	Настройка подключения к SMTP-серверу .....	21
4.5.8	Настройка параметров автоматического завершения одностороннего расторжения договоров .....	21
4.6	ПРОВЕРКА ДОСТУПНОСТИ ПРИЛОЖЕНИЯ .....	21

# 1 Общие положения

## 1.1 Полное наименование Системы и ее условное обозначение

Полное наименование: Подсистема работы с централизованными договорами.

Краткое наименование: Система.

## 1.2 Определения, обозначения и сокращения

В документе используются термины и сокращения, приведенные в таблице ниже.

Термин / Сокращение	Определение
БД	База данных
ЕИС	Официальный сайт единой информационной системы в сфере закупок
MS	Microsoft
Оператор ЭДО	Оператор электронного документооборота
ОС	Операционная система
ПО	Программное обеспечение
ПК	Портативный компьютер
Разработка, доработка	Процесс, подразумевающий разработку кода, в соответствии с требованиями, описанными в Функциональном Дизайне. Результатом является пакет документов и кода.
Разработчик	АО «Панорама»
SMTP	Simple Mail Transfer Protocol
Система	Подсистема работы с централизованными договорами
СУБД	Система управления базами данных
СХД	Система хранения данных
ЦОД	Центр обработки данных
ЭДО	Электронный документооборот, система электронного документооборота
ЭП	Электронная подпись

## **2 Назначение и условия применения программного обеспечения**

### **2.1 Назначение программного обеспечения**

**Подсистема работы с централизованными договорами предусматривает следующие цели:**

- Реализация работы с множеством карточек нижестоящих (дополняющих) договоров к основному договору (централизованному);
- Формирование дополнительных соглашений с привязкой к нижестоящим договорам, а также к основному договору (централизованному);
- Формирование привязки множества карточек нижестоящих (дополняющих) договоров с основным договором (централизованным);
- Контроль лимита суммы основного (централизованного) договора в разрезе нижестоящих (дополняющих) договоров;
- Визуальное представление нижестоящих (дополняющих) договоров;
- Визуальное представление сведений по основному (централизованному) договору с учетом сведений по нижестоящим (дополняющим) договорам с возможностью поиска;
- Формирование отчетов по нижестоящим (дополняющим) договорам, по основному (централизованному) договору с возможностью выгрузки.

**Подсистема работы с централизованными договорами обеспечивает информационное взаимодействие:**

- С Заказчиком с целью передачи следующих видов информации по основным (централизованным) и нижестоящим (дополняющим) договорам в части:
  - а) создания/ размещения/ подписания договора;
  - б) уточнения/ аннулирования договора;
  - в) сроки/ статусы исполнения обязательств/ договора;
  - д) извещений об оплате по договору/ штрафа/ пени/ неустойки и т.д.
  - е) формирование отчетов/ аналитических данных по договору/ группе договоров.
- С Исполнителем с целью передачи следующих видов информации:
  - а) внесение изменений/ отклонения/ подписания договора с Заказчиком;

b) перечень заявок, принятых от Заказчика/ Третьей стороны на исполнение обязательств по договору;

c) создание заявки на исполнение обязательств;

d) сроки/ статусы исполнения обязательств/ договора;

e) с извещений об оплате по договору/ штрафа/ пени/ неустойки и т.д.

f) формирование отчетов/ аналитических данных по договору/ группе договоров.

- С Оператором ЭДО «ФТ-Крипто» (далее Оператор ЭДО) - используется для выполнения операции подписания документов электронной подписью (далее ЭП). Данные передаются в формате Оператора ЭДО.
- С Официальным сайтом единой информационной системы в сфере закупок (ЕИС) – используется для загрузки информации в Систему о заключенном/исполненном/расторгнутом договоре по централизованной закупке;
- С почтовым сервером (SMTP);
- С прочими системами, подключенные к Системе посредством API – Система не требует подключения внешних систем, но допускает такую интеграцию.

## 2.2 Уровень подготовки персонала

Для штатной эксплуатации Системы необходимо привлечение следующих групп персонала:

### **Обслуживающий персонал:**

- Администратор Системы;
- Администратор баз данных (БД);
- Специалист по техническому обслуживанию.

Пользователи (из числа администраторов Системы) осуществляют функции мониторинга и поддержки работы в Системе.

## **Рекомендуемая численность персонала для эксплуатации Системы:**

- Системный администратор – 1 штатная единица;
- Администратор баз данных – 1 штатная единица;
- Специалист по техническому обслуживанию – 1 штатная единица.

### **Системный администратор.**

Основными обязанностями системного администратора являются: установка, настройка и мониторинг работоспособности программного обеспечения.

Системный администратор должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию программных средств, применяемых в Системе, а также должен иметь профессиональные знания и практический опыт в области системного администрирования. Обязательны знакомство и практический опыт установки и администрирования серверных операционных систем семейства Windows, программного обеспечения Крипто Про.

### **Администратор баз данных.**

Администратор баз данных Системы должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию Системы.

### **Основными обязанностями администратора баз данных являются:**

- установка, модернизация, настройка параметров программного обеспечения систем управления базами данных (СУБД);
- оптимизация функционирования прикладных баз данных по времени отклика, скорости доступа к данным;
- резервное копирование и аварийное восстановление данных;
- конфигурирование и настройка программно-технических средств Системы;
- разработка, управление и реализация эффективной политики доступа к информации, хранящейся в прикладных базах данных;
- ввод и поддержание в актуальном состоянии общих разделов базы данных (классификаторов).

Роли системного администратора и администратора баз данных могут быть совмещены в одну роль.

## **Специалист по техническому обслуживанию.**

Основными обязанностями специалиста по техническому обслуживанию являются:

- модернизация, настройка и мониторинг работоспособности комплекса технических средств (серверов, ПК);
- конфигурирование и настройка программно-технических средств;
- диагностика типовых неисправностей;
- замена базовых узлов периферийных устройств, имеющих ограниченный ресурс;
- настройка локальной компьютерной сети и Интернета;
- контроль доступа к сетевым ресурсам;
- настройка сетевого окружения.

### **Квалификация обслуживающего персонала должна позволять:**

- использовать стандартные возможности применяемых типовых средств вычислительной техники, ОС, СУБД и другого системного ПО;
- работать с архиваторами, дисковыми утилитами, антивирусными программами и программами резервного копирования;
- определять источник сбоя функционирования и отказа Системы;
- восстанавливать работоспособность Системы после сбоя или отказа;
- проводить регламентные работы и техническое обслуживание Системы.

### 3 Архитектурные решения

#### 3.1 Архитектурные решения Системы

**Пользователями Системы являются:**

1. Заказчики, договоры которых исполняются в Системе;
  2. Исполнители и третьи стороны, являющиеся сторонами исполняющихся договоров в Системе.
  3. Операторы Системы (сотрудники техподдержки).
- Технологическая схема приведена на рисунке 1.

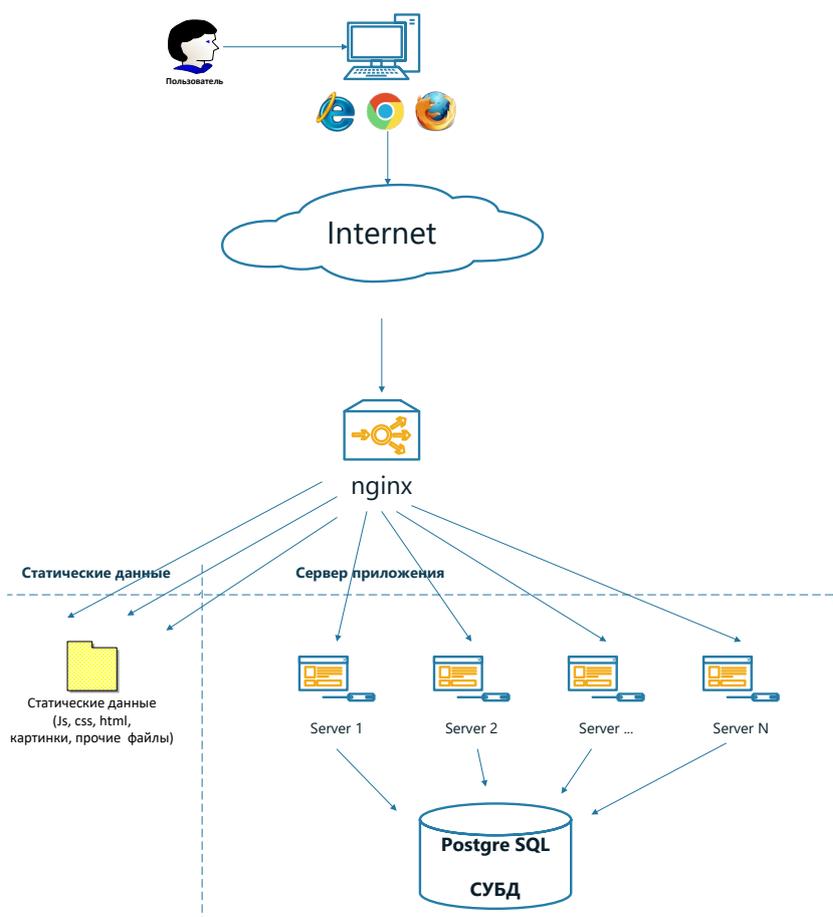


Рисунок 1 – Технологическая схема.

**Алгоритм обработки запросов:**

1. Пользователь Системы через браузер вводит запрос (открывает страницу, загружает или выгружает документ);
2. Сервер nginx распределяет полученные запросы от пользователей.
  - а) Если клиентом затребованы статические данные (JS, CSS, HTML и прочие ресурсы веб-приложения Системы), то обращение идет к серверу статических данных;

б) Если клиентом затребованы динамические данные, то nginx формирует запрос на свободный сервер приложений Системы, который, в свою очередь, обращается к СУБД.

с) Доступ к СУБД может быть реализован как с использованием одной СУБД, так и двух реплик: реплика master (read-write) и реплика read-only. В приложении реализована поддержка нескольких connection string (для master и slave отдельно).

### **Система состоит из следующих компонентов:**

- Веб-сервер;
- Сервер приложений;
- Сервер СУБД;
- Криптосервер.

Для работы администраторов используются АРМ администраторов Системы.

В качестве технологической платформы веб-сервера используются продукт семейства Linux (например, CentOS).

## **3.2 Порядок загрузки данных и программ**

**Порядок подготовки сервера для работы с Системой включает следующие этапы:**

1. Установка OpenJDK. Порядок установки описан в [пункте 4.1](#) настоящего документа;
2. Установка Tomcat 8. Порядок установки описан в [пункте 4.2](#) настоящего документа;
3. Установка СУБД PostgreSQL 9.5+. Порядок установки описан в [пункте 4.3](#) настоящего документа;
4. Установка и обновление приложения. Порядок установки описан в [пункте 4.4](#) настоящего документа:
  - ROOT\_centralizov dogovory.zip - серверное приложение Системы;
  - client\_centralizov dogovory.zip - клиентское приложение Системы;
  - db\_centralizov dogovory.zip - приложение для установки/обновления базы данных.
5. Настройка приложения. Порядок установки описан в [пункте 4.5](#) настоящего документа;
6. Проверка доступности приложения. Порядок установки описан в [пункте 4.6](#) настоящего документа.

## **4 Установка программного обеспечения**

### **4.1 Установка программного обеспечения Подсистемы работы с централизованными договорами**

#### **4.1.1 Установка операционной системы и общесистемных компонентов**

Установка программ и настройка операционной системы и общесистемных компонентов осуществляются стандартным образом. При настройке программного обеспечения, серверам необходимо присваивать имена, IP-адреса в соответствии с настоящим документом.

Ниже описан процесс подготовки сервера к запуску приложения. Описывается процесс установки приложения на сервер под управлением ОС семейства Linux (например, CentOS).

#### **4.1.2 Установка OpenJDK 11**

##### **4.1.2.1 Установка OpenJDK 11**

**Для установки OpenJDK 11, выполнить:**

```
cd /tmp
wget https://download.java.net/java/GA/jdk11/9/GPL/openjdk-11.0.2_linux-x64_bin.tar.gz
tar xzvf openjdk-11.0.2_linux-x64_bin.tar.gz
sudo mv jdk-11.0.2 /usr/lib/jvm/java-11-openjdk-amd64/
```

##### **4.1.2.2 Настройка Java**

На одном сервере может одновременно находиться несколько установок Java. Вы можете выбрать версию Java, которая будет использоваться Системой по умолчанию. Для этого существует команда `update-alternatives`, управляющая символическими ссылками, которые используются в различных командах.

```
sudo update-alternatives --install /usr/bin/java java /usr/lib/jvm/java-11-openjdk-amd64/bin/java 2
sudo update-alternatives --install /usr/bin/javac javac /usr/lib/jvm/java-11-openjdk-amd64/bin/javac 2
sudo update-alternatives --config java
```

Команда вернет информацию о всех установленных пакетах java. Теперь можно выбрать номер версии, которая будет использоваться по умолчанию.

##### **4.1.2.3 Переменная JAVA\_HOME**

```
sudo update-alternatives --config java
```

Скопируйте путь необходимой версии программы и откройте `/etc/environment` в текстовом редакторе:

```
sudo nano /etc/environment
```

Добавьте следующую строку в конец файла (замените условный путь к программе только что скопированным путём):

```
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"
```

Сохраните и закройте файл (Ctrl + O, Ctrl + X). Чтобы обновить настройки, введите:

```
source /etc/environment
```

Убедитесь, что переменная среды установлена. Для этого введите:

```
echo $JAVA_HOME
```

Команда должна вернуть установленный вами путь.

## 4.2 Установка Tomcat 8

### 4.2.1 Создание пользователя Tomcat

В целях безопасности сервер Tomcat следует запускать с правами обычного пользователя (т.е. без root-доступа). Для этого создайте нового пользователя и группу. Чтобы создать группу Tomcat, введите:

```
sudo groupadd tomcat
```

После этого нужно создать пользователя Tomcat, который должен состоять в одноименной группе. Домашний каталог пользователя – /opt/tomcat (в него будет установлен Tomcat); оболочка – /bin/false (чтобы никто не мог открыть учётную запись):

```
sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat
```

### 4.2.2 Установка Tomcat

На данный момент Tomcat 8 проще всего установить вручную из бинарного релиза. На странице загрузки найдите последнюю версию Tomcat. В разделе Binary Distributions найдите список Core и скопируйте ссылку на tar.gz. Перейдите в каталог /tmp на сервере. Этот каталог предназначен для временных пакетов.

```
cd /tmp
```

Чтобы загрузить архив Tomcat, используйте следующую команду (приведенная ниже ссылка может отличаться):

```
curl -O http://apache-mirror.rbc.ru/pub/apache/tomcat/tomcat-8/v8.5.37/bin/apache-tomcat-8.5.37.tar.gz
```

Установите Tomcat в каталог /opt/tomcat. Создайте этот каталог и извлеките в него архив:

```
sudo mkdir /opt/tomcat  
sudo tar xzvf apache-tomcat-8*.tar.gz -C /opt/tomcat --strip-components=1
```

#### 4.2.3 Настройка прав

Пользователь Tomcat должен иметь доступ к установке Tomcat. Откройте каталог Tomcat:

```
cd /opt/tomcat
```

Дайте пользователю Tomcat право на чтение, запись и исполнение файлов внутри папки tomcat:

```
sudo chmod 770 -R /opt/tomcat/*
```

Установите владельца и группу для каталога tomcat:

```
sudo chown tomcat:tomcat -R /opt/tomcat
```

Для удобства администрирования добавьте текущего пользователя (informadmin) в группу tomcat:

```
sudo usermod -G tomcat informadmin
```

#### 4.2.4 Создание файла systemd

Чтобы запустить Tomcat как сервис, нужно создать service-файл systemd. Серверу Tomcat нужно знать, где находится установка Java. При необходимости это путь можно переопределить в JAVA\_HOME.

Для создания service-файла откройте tomcat.service в каталоге /etc/systemd/system:

```
sudo nano /etc/systemd/system/tomcat.service
```

Внесите в файл следующий скрипт, при необходимости изменив JAVA\_HOME; также можно изменить настройки распределения памяти, которые указаны в CATALINA\_OPTS:

```
[Unit]  
Description=Apache Tomcat Web Application Container  
After=network.target
```

```
[Service]
Type=forking
Environment=JAVA_HOME=/usr/lib/jvm/java-8-oracle/jre
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-Xms1024M -Xmx2048M -server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
User=tomcat
Group=tomcat
[Install]
WantedBy=multi-user.target
```

Сохраните и закройте файл. После этого нужно перезапустить демон systemd:

```
sudo systemctl daemon-reload
```

Теперь сервис Tomcat готов к запуску. Для запуска сервиса используйте:

```
sudo systemctl start tomcat
```

Убедитесь, что при запуске не произошло ошибок:

```
sudo systemctl status tomcat
```

Для обработки запросов Tomcat использует порт 8080. Чтобы разблокировать трафик этого порта, введите:

```
sudo ufw allow 8080
```

Теперь брандмауэр пропускает трафик Tomcat. Чтобы получить доступ к стандартной странице сервиса, откройте ссылку: [http://server\\_domain\\_or\\_IP:8080](http://server_domain_or_IP:8080).

Также необходимо в настройках сети и/или на локальной машине обеспечить доступ к скачиванию файлов (модуль/ content) по внешнему адресу стенда. Решается это либо на уровне сетевого администратора, либо добавлением внешнего адреса в файл hosts (не рекомендуется).

#### 4.2.5 Настройка Tomcat

Для изменения порта, на котором будет работать tomcat, необходимо отредактировать файл server.xml:

```
sudo nano /opt/tomcat/conf/server.xml
```

Заменяем порт 8080 на нужный нам порт, например, на 80:

```
<Connector port="80" protocol="HTTP/1.1"
```

И открываем порт наружу:

```
sudo ufw allow 80
```

Если необходимо, создаем пользователей Tomcat:

```
sudo nano /opt/tomcat/conf/tomcat-users.xml
```

и заменяем содержимое файла на:

```
<?xml version="1.0" encoding="UTF-8"?>
<tomcat-users xmlns="http://tomcat.apache.org/xml"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
  version="1.0">
<role rolename="tomcat"/>
<user username="dms" password="<password_here>" roles="tomcat,manager-script,admin-
script,admin-gui,manager-status,manager-gui"/>
</tomcat-users>
```

Перезапускаем Tomcat:

```
sudo systemctl restart tomcat
```

Также можно оставить Tomcat на порту 8080, но перенаправить трафик с 80 на 8080

```
sudo iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080
sudo iptables -t nat -I OUTPUT -p tcp -o lo --dport 80 -j REDIRECT --to-ports 8080
```

Если мы собираемся получать доступ с Tomcat Manager (например, с билд-сервера), нужно закомментировать VALVE в файле context.xml, либо добавить туда регулярку, учитывающую наш адрес билд-сервера

```
sudo nano /opt/tomcat/webapps/manager/META-INF/context.xml
```

Для приложения необходимо выдать пользователю права и сделать владельцем для папки /var/dms (или см. конфиги)

```
sudo chmod 770 -R /var/dms/*
sudo chown tomcat:tomcat -R /var/dms
```

## 4.3 Установка СУБД PostgreSQL 9.5+

### 4.3.1 Установка PostgreSQL

Для установки создайте файл источника репозитория для PostgreSQL с указанием в нем версии операционной системы

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'
```

Устанавливаем необходимые компоненты:

```
sudo apt-get install wget ca-certificates  
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -  
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get install postgresql-9.5 pgadmin3
```

#### 4.3.2 Создание базы данных в PostgreSQL

Выполним команды от имени postgres

```
sudo -i -u postgres
```

Запустим psql

```
psql
```

Создадим тестовую базу данных и тестового пользователя с помощью команды CREATE DATABASE <имя бд>:

```
create database dms;
```

Создание пользователя CREATE USER пользователь WITH password 'пароль':

```
create user dms with password '123';
```

Выделение прав с помощью команды GRANT ALL privileges ON DATABASE имя БД TO пользователь:

```
grant all privileges on database dms to dms;
```

Если необходимо сделать пользователя суперпользователем (с расширенными правами), используем команду:

```
alter user dms WITH SUPERUSER;
```

Для выхода из psql используйте \q Для выхода из под postgres пользователя используйте exit.

#### 4.3.3 Открытие внешнего доступа к серверу

Возможно, потребуется изменить pg\_hba.conf и / или pg\_ident.conf, postgresql.conf, чтобы разрешить внешний доступ:

```
sudo nano /etc/postgresql/9.5/main/pg_hba.conf
```

в конце файла после локальных правил, добавьте:

```
host all all 0.0.0.0/0 md5
```

Чтобы разрешить postgresql слушать все адреса, необходимо изменить файл postgresql.conf:

```
sudo nano /etc/postgresql/9.5/main/postgresql.conf
```

и раскомментировать свойство listen\_addresses, поменяв его на '\*':

```
listen_addresses = '*'
```

Перезапустите сервер БД:

```
sudo service postgresql restart
```

## 4.4 Установка и обновление приложения

**В комплект поставки входят следующие файлы:**

- ROOT\_centalizov dogovory.zip - серверное приложение Системы;
- client\_centralizov dogovory.zip - клиентское приложение Системы;
- db\_centralizov dogovory.zip - приложение для установки/обновления базы данных.

### 4.4.1 Обновление приложения

Останавливаем Tomcat:

```
sudo systemctl stop tomcat
```

Извлекаем новую версию приложения из архивов ROOT\_centalizov dogovory.zip и client\_centralizov dogovory.zip в соответствующие папки (ROOT\_centalizov dogovory и client\_centralizov dogovory):

```
sudo unzip /var/dms/distr/ROOT_centalizov dogovory.zip -d /opt/tomcat/webapps/  
ROOT_centalizov dogovory
```

```
sudo unzip /var/dms/distr/client_centralizov dogovory.zip -d /opt/tomcat/webapps/  
client_centralizov dogovory
```

Меняем группу-владельца для новых файлов на tomcat:

```
sudo chgrp -R tomcat /opt/tomcat/webapps
```

При необходимости указания собственных настроек сервера создаем или меняем файл `custom.properties`:

```
sudo nano /opt/tomcat/webapps/ROOT_centalizov_dogovory/WEB-INF/classes/custom.properties
```

Все возможные для изменения настройки описаны ниже, в разделе Настройка приложения.

Сохраняйте конфигурационный файл `custom.properties` перед обновлением приложения каждый раз при установке новой версии.

#### 4.4.2 Обновление базы данных

Извлекаем скрипт обновления базы данных:

```
sudo unzip /var/dms/distr/db_centralizov_dogovory.zip -d /tmp/db_centralizov_dogovory
```

Перейдите в директорию, в которую извлекли файлы из архива:

```
cd /tmp
```

Запустите файл `update.sh`:

```
./update.sh
```

База данных должна успешно обновиться.

#### 4.4.3 Запуск новой версии приложения

Запустите сервис `tomcat`

```
sudo systemctl restart tomcat
```

## 4.5 Настройка приложения

Все возможные для задания настройки, которые можно задать в конфигурационном файле, перечислены ниже.

### 4.5.1 Настройки соединения с базой данных

Настройки конфигурационного файла зависят от используемой базы данных. Ниже приведены настройки для базы PostgreSQL:

```
#jdbc driver for selected database
jdbc.driverClassName=org.postgresql.Driver
#jdbc connection url for selected database
jdbc.url=jdbc:postgresql://95.213.206.108:5432/dms
#database user name
jdbc.user=dms
#database user password
jdbc.password=123
#hibernate dialect to be used, depends from selected database
hibernate.dialect=ru.dms.core.commons.hibernate.CustomPostgreSQL9Dialect
#hibernate configuration that depends from selected database, quotation for identifiers (false for
PostgreSQL always) and policy for id generation (false for PostgreSQL always)
hibernate.globally_quoted_identifiers=false
hibernate.id.new_generator_mappings=false
#jdbc connection url for quartz database, can be same as jdbc.url
jobs.quartz.dataSource.URL= jdbc:postgresql://95.213.206.108:5432/dms
#quartz database user name
jobs.quartz.dataSource.user=dms
#quartz database user password
jobs.quartz.dataSource.password=123
#quartz database jdbc driver
jobs.quartz.dataSource.driver=org.postgresql.Driver
#quartz database jdbc driver delegate
jobs.quartz.dataStore.driverDelegateClass=org.quartz.impl.jdbcjobstore.PostgreSQLDelegate
```

Вы можете переопределить любое из свойств. Например, для задания имени пользователя базы данных вы должны задать параметр `jdbc.user`.

### 4.5.2 Параметры развертывания

В параметрах ниже указать IP-адрес веб-сервера:

```
#external url of application (used by users)
web_client_centralizov_dogovory.application.url=http://95.213.206.108/c3/#
#API external url
deployment.api.external.url=http://95.213.206.108/
#Installation owner: R
deployment.owner= R
```

### 4.5.3 Настройка свойств безопасности

```
#allowed authentication providers for installation (LOCAL)
auth.provider=LOCAL
#expiration time for refresh token (5days in example)
token.refresh.expiration=P5D
#expiration time for authentication token (4hours in example)
token.access.expiration=PT4H00M
#access token issuer (app address usually)
token.access.issuer=https://skd.digitalpanorama.ru/
#access token secret key (plain format in example)
token.access.secret=plain:gY+SspAVnqqiCBAvNqXRSEtKreM0iMt6NdFSjUeMI4=
#files download token expiration time (2minutes in example)
token.download.expiration=PT2M
```

### 4.5.4 Настройка параметров подключения к системе ЭДО

Настройте параметры подключения к серверу Оператора ЭДО, если требуется:

```
#EDO exchange properties
edo.base.url=https://eds-pro-stable.ft-crypto.ru
edo.service.url=https://eds-pro-api-stable.ft-crypto.ru/api/ExchangeDocument
edo.service.login=<edoLogin>
edo.service.password=<edoPassword>
```

### 4.5.5 Настройка расписания выполнения фоновых задач

```
#background tasks schedule configurations in cron format
#EDO messages receive schedule
cron.edo.messages.receive=0 */1 * * * ?
#send messages to EDO schedule
cron.edo.messages.send=30 */1 * * * ?
#EDO messages process schedule
cron.edo.messages.process=45 */1 * * * ?
#auto complete document events job schedule
cron.execution.documents.events.execute=0 15 1 * * ?
#sending of pending outgoing email messages
cron.email.outgoing.send=0 * * * * ?
#delete old EDO outgoing messages schedule
cron.edo.outgoing.messages.delete=0 20 1 * * ?
#delete old EDO incoming messages schedule
cron.edo.incoming.messages.delete=0 40 1 * * ?
#finalize single side contract terminations after settlement period
cron.contract.termination.finalize=0 25 1 * * ?
#remove old system logs (more than 3 days)
cron.system.logs.delete=0 0 4 * * ?
```

### 4.5.6 Настройка подключения к хранилищу для файлов

В параметрах ниже указать адрес сервера и папку для хранения пользовательских файлов:

```
#full path to folder for uploaded files
local_storage.base_directory=C:/dms_storage
```

```
#base url for files download (content module deployed in tomcat is used in example, nginx
configuration is possible and recommended for production)
secured_link.base_url=http://95.213.206.108/content
#secret word for secured link generation in application and reading in files server (secured link
module for nginx or content module for tomcat)
secured_link.secret=secret
```

#### 4.5.7 Настройка подключения к SMTP-серверу

В параметрах ниже указать параметры подключения к SMTP-серверу для отправки писем:

```
mail.transport.protocol=smtp
mail.smtp.auth=true
mail.smtp.starttls.enable=true
mail.smtp.host=smtp-server-url
mail.smtp.port=25
mail.smtp.user=SMTP-login
mail.smtp.password=SMTP-password
mail.debug=true
mail.from=mail@domain.ccc
mail.batch.size=500
```

#### 4.5.8 Настройка параметров автоматического завершения одностороннего расторжения договоров

```
#single side contract termination settlement period duration
contract.termination.defection.correction.duration=10
```

### 4.6 Проверка доступности приложения

После настройки сервера и установки приложения на нем, сделать:

1. Открыть браузер
2. Ввести ссылку на установленный выше веб-сервер;
3. Убедиться, что на экране отобразилась страница входа;
4. Ввести произвольные логин и пароль;
5. Убедиться, что на экране отобразилась страница логина и сообщение

«Неверное имя пользователя или пароль».