

ООО «НТЦ ТРАНССИСТЕМОТЕХНИКА»

Программное обеспечение
Цифровая транспортно-логистическая платформа (ЦТЛП)

Руководство по установке

2024 г.

Оглавление

1. ОБЩИЕ СВЕДЕНИЯ	4
2. СОСТАВ ДИСТРИБУТИВА ЦТЛП	4
3. ТРЕБОВАНИЯ К ПОДГОТОВКЕ ПРОГРАММНО-АППАРАТНОГО СЕРВЕРНОГО КОМПЛЕКСА	4
4. РАЗВЕРТЫВАНИЕ СУБД POSTGRESQL	5
5. УСТАНОВКА СЕРВЕРА ПРИЛОЖЕНИЙ WILDFLY	7
6. УСТАНОВКА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЦТЛП	10
7. УСТАНОВКА HTTP СЕРВЕРА NGINX	10
8. НАСТРОЙКА ПОЛЬЗОВАТЕЛЬСКОЙ ССЫЛКИ ДЛЯ РАБОТЫ С ЦТЛП В БРАУЗЕРЕ РАБОЧЕГО МЕСТА ПОЛЬЗОВАТЕЛЕЙ ЦТЛП	12
9. НАСТРОЙКА ИНФОРМАЦИОННОГО ВЗАИМОДЕЙСТВИЯ В РЕЖИМЕ АСУ-АСУ С АС ОАО «РЖД»	12

АННОТАЦИЯ

Документ содержит информацию, необходимую для установки программного обеспечения «Цифровая транспортно-логистическая платформа» (ЦТЛП).

1. Общие сведения

Программное обеспечение ЦТЛП реализовано в трехуровневой архитектуре с использованием централизованного сервера базы данных, сервера приложений и браузера на рабочих местах пользователей. Количество пользователей системы определяется производительностью серверного комплекса ЦТЛП.

Условием применения ЦТЛП является действующий режим информационного взаимодействия в режиме АСУ-АСУ с автоматизированным системам ОАО «РЖД».

2. Состав дистрибутива ЦТЛП

Состав и содержание дистрибутивного носителя данных:

- Сборка СУБД PostgreSQL;
- Серверные модули приложений;
- Клиентские модули приложений.

Архив с дистрибутивом предоставляется Заказчику в соответствии с лицензионным договором.

Установка и настройка СУБД PostgreSQL, сервера приложений WildFly, HTTP сервера Nginx, серверных и клиентских модулей ЦТЛП, базы данных ЦТЛП, выполняется однократно, при запуске ЦТЛП в эксплуатацию.

3. Требования к подготовке программно-аппаратного серверного комплекса

Перед установкой ПО ЦТЛП, полученного в составе дистрибутива, необходим серверный аппаратно-программный комплекс, подготовленный Заказчиком:

- 1 Установлена операционная система (ОС) RedOS 8.0.
- 2 Создана и настроена учётная запись root.

Сервер должен соответствовать следующим минимальным требованиям:

Сервер	Описание	Назначение		Роль	
		Сервер СУБД Сервер приложений	-		Основной
Аппаратное обеспечение	CPU, кол-во ядер	RAM, ГБ	SWAP, ГБ	Тактовая частота, ГГц	HDD,ГБ
	64-bit (x86_64), 6	8	8	-	500
Логические диски	LUN	Точка монтирования		Логическая емкость, ГБ	
	LUN0	/		50	
	LUN1	/u01		15	
	LUN2	/u02		200	
	LUN3	/u03		35	
LUN4	/u04		200		

4. Развертывание СУБД PostgreSQL

Установите следующие пакеты в ОС:

```
# yum install make gcc libuuid uuid uuid-devel
# yum install readline-devel zlib-devel systemd-devel libxml2-devel
# yum install flex bison
# yum install perl
# yum install lz4 lz4-devel
```

Скопируйте дистрибутив PostgreSQL в раздел /distr/pgsql на сервере ЦТЛП и распакуйте его.

```
# unzip postgresql-17.0-red8.zip
```

Скопируйте скрипты управления конфигурацией PostgreSQL в /u01/pgsql и распакуйте их:

```
# cd /u01/pgsql
# unzip pgs-scripts-lin.zip
```

Создайте пользователя ОС postgres:

```
# groupadd -g 776 postgres
# useradd -g postgres -u 777 postgres
# passwd postgres
```

Выполните скрипт root.sh из раздела управления конфигурацией PostgreSQL:

```
# cd /u01/pgsql/scripts
# ./root.sh
```

Добавьте в конец файла /home/postgres/.bash_profile следующие строки:

```
# PostgreSQL Settings
export PGS_HOME=/u01/pgsql/pgs17
export PGS_DATA=/u02/pgsql/pgsdata17
LD_LIBRARY_PATH=$PGS_HOME/lib:/lib:/usr/lib
export LD_LIBRARY_PATH
PATH=$PGS_HOME/bin:$PATH:$HOME/.local/bin:$HOME/bin
export PATH
```

Создание службы для автоматического старта СУБД PostgreSQL. Скопируйте файл /u01/pgsql/scripts/templ/pgsql.service в /etc/systemd/system и выполните:

```
# systemctl daemon-reload
# systemctl enable postgresql.service
```

Установите PostgreSQL:

```
# cd /distr/pgsql/postgresql-17.0-red8
# make install
# ln -sf /u01/pgsql/pgs17/bin/* /usr/sbin/
```

Установите необходимые расширения PostgreSQL:

```
# cd /u01/pgsql/scripts/shell
# ./pgs_contrib.sh
```

Выполните инициализацию СУБД из под пользователя postgres:

```
# su - postgres
$ cd $PGS_HOME/bin
$ ./initdb --pgdata=$PGS_DATA --wal-segsize=64 --data-checksums --encoding=UTF8
```

Выполните настройку PostgreSQL:

Переименуйте файлы \$PGS_DATA/pg_hba.conf в \$PGS_DATA/pg_hba.orig.conf и \$PGS_DATA/postgresql.conf в \$PGS_DATA/postgresql.orig.conf.

Скопируйте /u01/pgsql/scripts/templ/postgresql.conf в \$PGS_DATA.

Скопируйте файлы /u01/pgsql/scripts/templ/pg_hba.conf и /u01/pgsql/scripts/templ/pg_hba.loc.conf в \$PGS_DATA.

Установите пароль для административной учётной записи PostgreSQL:

```
$ psql -c "alter user postgres with password '*****'"
```

Создайте новую базу данных, пользователей и табличные пространства:

```
$ cd /u01/pgsql/scripts/roles
$ psql < ctp_rw.sql
$ psql < ctp_ro.sql
$ psql < ctp.sql
```

```
$ cd /u01/pgsql/scripts/space
$ psql < ctp-ts.sql
$ cd /u01/pgsql/scripts/bases
$ psql < ctp-p01.sql
```

Запустите СУБД:

```
# systemctl start postgresql
```

Проверьте журнал СУБД на отсутствие ошибок в нём `$PG_DATA/log`.

5. Установка сервера приложений WildFly

Установите Open JDK в `/u01/java/jdk21x64`:

```
# mkdir -p /u01/java
# chmod -R 775 /u01/java
# cd /u01/java
# tar -zxvf jdk-21_linux-x64_bin.tar.gz
# alternatives --install /usr/bin/java java /u01/java/jdk21x64/bin/java 3
# alternatives --config java
# alternatives --install /usr/bin/jar jar /u01/java/jdk21x64/bin/jar 3
# alternatives --install /usr/bin/javac javac /u01/java/jdk21x64/bin/javac 3
# alternatives --set jar /u01/java/jdk21x64/bin/jar
# alternatives --set javac /u01/java/jdk21x64/bin/javac
# java -version
```

Создайте файл `/etc/profile.d/jdk.sh`:

```
export J2SDKDIR=/u01/java/jdk21x64
export J2REDIR=/u01/java/jdk21x64
export PATH=$PATH:/u01/java/jdk21x64/bin
export JAVA_HOME=/u01/java/jdk21x64
```

Создайте файл `/etc/profile.d/jdk.csh`:

```
setenv J2SDKDIR /u01/java/jdk21x64
setenv J2REDIR /u01/java/jdk21x64
setenv PATH ${PATH}:/u01/java/jdk21x64/bin
setenv JAVA_HOME /u01/java/jdk21x64
```

Скопируйте скрипты управления конфигурацией WildFly в `/u01/wildfly` и распакуйте их:

```
# cd /u01/wildfly
# unzip wfy-scripts-lin.zip
```

Создайте пользователя ОС wildfly:

```
# useradd wildfly
# passwd wildfly
```

Выполните скрипт root.sh из раздела управления конфигурацией WildFly:

```
# cd /u01/wildfly/scripts
# ./root.sh
```

Добавьте в конец файла /home/wildfly/.bash_profile следующие строки:

```
export JBOSS_HOME=/u01/wildfly/wfy33
export JAVA_HOME=/u01/java/jdk21x64
export WFY_SCRIPTS=/u01/wildfly/scripts
```

Распакуйте дистрибутив с wildfly в каталог /u01/wildfly/wfy33

```
# cd /u01/wildfly/wfy33
# unzip wildfly-33.0.2.Final.zip
```

Настройте аутентификацию:

```
# $WFY_HOME/bin/add-user.sh
What type of user do you wish to add?
  a) Management User (mgmt-users.properties)
  b) Application User (application-users.properties)
(a):
```

Далее скрипт предложит вам ввести данные нового пользователя:

```
Username: <username>
Password: *****
```

Скопируйте файлы asucor-jvm1.conf и asugui-jvm1.conf из раздела управления конфигурацией в /etc/wildfly

```
# cp /u01/wildfly/scripts/cfg/service/asucor-jvm1.conf /etc/wildfly
# cp /u01/wildfly/scripts/cfg/service/asugui-jvm1.conf /etc/wildfly
```

Скопируйте файлы asucor.conf, asugui.conf, asucor.sh, asugui.sh из раздела управления конфигурацией в \$WFY_HOME/bin

```
# cp /u01/wildfly/scripts/cfg/jvm/asucor.conf /u01/wildfly/wfy33/bin
# cp /u01/wildfly/scripts/cfg/jvm/asugui.conf /u01/wildfly/wfy33/bin
```

```
# cp /u01/wildfly/scripts/bin/asucor.sh /u01/wildfly/wfy33/bin
# cp /u01/wildfly/scripts/bin/asugui.sh /u01/wildfly/wfy33/bin
```

Настройка автоматического запуска серверов wildfly.

Скопируйте файлы asucor-jvm1.service и asugui-jvm1.service из раздела управления конфигурацией в /etc/systemd/system:

```
# cp /u01/wildfly/scripts/cfg/service/asucor-jvm1.service /etc/systemd/system
# cp /u01/wildfly/scripts/cfg/service/asugui-jvm1.service /etc/systemd/system
# systemctl daemon-reload
# systemctl enable asucor-jvm1
# systemctl enable asugui-jvm1
```

Настройте сервера wildfly.

Создайте две копии раздела \$WFY_HOME/standalone и назовите их:

\$WFY_HOME/asucor-jvm1

\$WFY_HOME/asugui-jvm1

Скопируйте в них следующие файлы из раздела управления конфигурацией:

```
# cp /u01/wildfly/scripts/cfg/jvm/33/asucor-jm1.xml /u01/wildfly/wfy33/asucor-
jm1/configuration
# cp /u01/wildfly/scripts/cfg/jvm/33/asugui-jm1.xml /u01/wildfly/wfy33/asugui-
jm1/configuration
```

Установка JDBC драйвера.

Создайте раздел /u01/wildfly/wfy33/modules/org/postgresql/main, если его нет:

```
# mkdir -p /u01/wildfly/wfy33/modules/org/postgresql/main
```

И скопируйте в него файл драйвера postgresql-42.7.4.jar и xml файл из раздела управления конфигурацией:

```
# cp postgresql-42.7.4.jar /u01/wildfly/wfy33/modules/org/postgresql/main
# cp /u01/wildfly/scripts/cfg/jdbc/postgres/module.xml
/u01/wildfly/wfy33/modules/org/postgresql/main
```

Запустите сервера wildfly:

```
# systemctl start asucor-jvm1
# systemctl start asugui-jvm1
```

Проверьте журналы серверов wildfly на отсутствие ошибок в них
\$WFY_HOME/asucor-jvm1/log, \$WFY_HOME/asugui-jvm1/log.

6. Установка прикладного программного обеспечения ЦТЛП

1. Скопируйте файлы client.war, trefpp.war из дистрибутива в раздел
/u02/wildfly/asudata/app на сервере ЦТЛП.

2. Выполните:

```
# cd /u01/wildfly/scripts/bin
# ./asu-replace.sh client.war
# ./asu-replace.sh trefpp.war
```

7. Установка HTTP сервера nginx

Скопируйте скрипты управления конфигурацией nginx в /u01/nginx и
распакуйте их:

```
# cd /u01/nginx
# unzip ngx-scripts-lin.zip
```

Выполните скрипт root.sh из раздела управления конфигурацией nginx:

```
# cd /u01/nginx/scripts
# ./root.sh
```

Установите следующие пакеты в ОС:

```
# yum groupinstall "Development Tools" -y
# yum install zlib-devel pcre-devel openssl-devel -y
# yum install perl perl-devel perl-ExtUtils-Embed libxslt libxslt-devel libxml2
libxml2-devel gd gd-devel GeoIP GeoIP-devel -y
```

Скопируйте архив с дистрибутивом nginx в /distr/nginx и распакуйте его.

```
# cd /distr/nginx
# unzip nginx-1.26.2-red8.zip
```

Выполните установку nginx:

```
# cd /distr/nginx/nginx-1.26.2-red8
# make install
```

Создайте пользователя ОС nginx:

```
# useradd nginx
# passwd nginx
```

Добавьте в конец файла /home/nginx/.bash_profile следующие строки:

```
# Nginx Settings
export NGX_HOME=/u01/nginx/nginx26
PATH=$NGX_HOME/sbin:$PATH:$HOME/.local/bin:$HOME/bin
export PATH
```

Настройка автоматического запуска nginx.

Скопируйте файлы nginx.service из раздела управления конфигурацией в /etc/systemd/system:

```
# cp /u01/nginx/scripts/tem/nginx.service /etc/systemd/system
# systemctl daemon-reload
# systemctl enable nginx
```

Настройка HTTP сервера.

Скопируйте в папку /u01/nginx/html файлы из раздела управления конфигурацией.

```
# cp /u01/nginx/scripts/htm/* /u01/nginx/html
```

Скопируйте в папку \$NGX_HOME/conf файл nginx.conf из раздела управления конфигурацией.

```
# cp /u01/nginx/scripts/cfg/nginx.conf /u01/nginx/nginx26/conf
```

Скопируйте в папку \$NGX_HOME/conf.d файлы asucor-hosts.conf, asugui-hosts.conf из раздела управления конфигурацией.

```
# cp /u01/nginx/scripts/cfg/asucor-hosts.conf /u01/nginx/nginx26/conf.d
# cp /u01/nginx/scripts/cfg/asugui-hosts.conf /u01/nginx/nginx26/conf.d
```

Запустите сервер nginx:

```
# systemctl start nginx
```

Проверьте журналы сервера nginx на отсутствие ошибок в них
\$NGX_HOME/logs.

8. Настройка пользовательской ссылки для работы с ЦТЛП в браузере рабочего места пользователей ЦТЛП

Запуск головной задачи ЦТЛП в браузере пользователя системы настраивается созданием ссылки: http://адрес_nginx:порт_nginx/client.

9. Настройка информационного взаимодействия в режиме АСУ-АСУ с АС ОАО «РЖД»

Настройка информационного взаимодействия ЦТЛП с АС ОАО «РЖД» в режиме АСУ-АСУ выполняется совместно с Заказчиком, в соответствии с утвержденным ОАО «РЖД» и Заказчиком Проектом технических решений на подключение к АС ОАО «РЖД», в порядке, определенном на <https://cargo.rzd.ru> .