

**Руководство по администрированию и установке  
«Plus7 Forsage Platform»**

## Содержание

1.	Введение .....	4
2.	Глоссарий .....	5
3.	Необходимое ПО для работы системы .....	6
3.1.	РНР 7.3 .....	6
3.1.1.	Модули РНР, необходимые для работы системы .....	6
3.1.2.	Конфигурация системы .....	7
3.2.	АРАСНЕ .....	7
3.2.1.	Установка Apache .....	7
3.2.2.	Настройка Apache .....	7
3.2.3.	Перезапуск Apache .....	8
3.3.	Graphviz .....	9
3.4.	СУБД PostgreSQL .....	9
3.4.1.	Создание пользователя и схемы БД .....	10
3.4.2.	Создание базы данных .....	10
3.4.3.	Создание первого пользователя в системе .....	11
3.4.4.	Заливка в БД тестовых данных .....	13
4.	Конфигурация системы .....	14
4.1.	Конфигурация пути для вызова ПО graphviz .....	14
4.2.	Дистрибутив системы .....	14
4.3.	Настройки, хранящиеся в конфигурационном массиве \$sys_settings. ....	14
4.3.1.	Параметры соединения с БД .....	14
4.3.1.	Параметры двухфакторной аутентификации .....	14
4.3.1.	Параметры отправки SMS-сообщений .....	15
4.3.2.	Параметры отправки email-сообщений .....	15
4.3.1.	Параметры регистрации новых пользователей системы .....	16
5.	Конфигурация файловой системы web-сервера .....	17
6.	Подсистема запуска фоновых процессов системы на стороне сервера .....	18
7.	Состав дистрибутива .....	19
8.	Установка обновления системы .....	20
8.1.	Порядок установки обновлений .....	20

**История внесения изменений в документ**

<b>Дата</b>	<b>Версия</b>	<b>Изменения</b>	<b>Автор</b>
30.06.2019	1.0	Создание документа.	Сафронов Д. Г.
06.05.2022	1.1	Обновление документа.	Захаренко В. В.

## **1. Введение**

Платформа «Plus7 Forsage Platform» предоставляет возможность быстрого создания пользовательских интерфейсов.

Настоящий документ содержит описание и инструкции по администрированию данной платформы.

## 2. Глоссарий

Термин	Определение
БД	База данных
СУБД	Система управления базы данных
ПО	Программное обеспечение

### 3. Необходимое ПО для работы системы

#### 3.1. PHP 7.3

Система написана с использованием PHP версии 7.3, но для операционной системы RedOS, будет использована версия (7.4.11).

##### 3.1.1. Модули PHP, необходимые для работы системы

Для работы системы под операционной системой в RedOS использовались следующие модули для PHP:

- php
- php-cli
- php-common
- php-gd
- php-json
- php-mbstring
- php-mcrypt (php-pecl-mcrypt)
- php-pdo
- php-xdebug (php-pecl-xdebug)
- php-pgsql
- php-soap
- php-xml
- php-xmlrpc

Пример команды установки для dnf:

```
dnf install php
```

Проверим что пакет установлен:

```
dnf list installed php
```

После установки Apache проверяем работу php, для этого создадим файл в директории виртуального хоста и проверим работу:

```
mcedit /web/site1.ru/www/index.php
```

```
<?php phpinfo(); ?>
```

```
chown apache. /web/site1.ru/www/index.php
```

Заходим по адресу <http://site1.ru/index.php>

PHP Version 5.4.16



### 3.1.2. Конфигурация системы

После установки часто возникает вопрос, а где хранятся настройки php? Традиционно они находятся в едином файле настроек, **php.ini лежит в /etc**, прямо в корне. Там можно редактировать глобальные настройки для все виртуальных хостов. Персональные настройки каждого сайта можно сделать отдельно в файле конфигурации виртуального хоста, который мы сделали раньше. Можно сделать несколько полезных настроек:

```
mcedit /etc/httpd/conf.d/sitel.ru.conf
```

Добавляем в самый конец, перед </VirtualHost>

```
php_admin_value date.timezone 'Europe/Moscow'
```

```
php_admin_value max_execution_time 60
```

```
php_admin_value upload_max_filesize 30M
```

Для применения настроек нужно сделать restart apache. Теперь в выводе phpinfo можно увидеть изменение настроек.

## 3.2. APACHE

### 3.2.1. Установка Apache

```
dnf install -y httpd
```

Добавляем apache в автозагрузку:

```
systemctl enable httpd
```

Запускаем apache в RedOS:

```
systemctl start httpd
```

Проверяем, запустился ли сервер:

```
netstat -tulnp | grep httpd
```

### 3.2.2. Настройка Apache

На данной платформе будет реализована следующая структура веб хостинга:

/web	раздел для размещения сайтов
/web/ <u>sitel.ru</u> /www	директория для содержимого сайта
/web/ <u>sitel.ru</u> /logs	директория для логов сайта

Создаем подобную структуру:

```
mkdir /web && mkdir /web/sitel.ru && mkdir /web/sitel.ru/www && mkdir /web/sitel.ru/logs
```

```
chown -R apache. /web
```

Дальше редактируем файл конфигурации apache - **httpd.conf** по адресу /etc/httpd/conf. Первым делом проверим, раскомментирована ли строчка в самом конце:

```
IncludeOptional conf.d/*.conf
```

Если нет, раскомментируем и идем в каталог /etc/httpd/conf.d. Создаем там файл **site1.ru.conf**:

```
mcedit /etc/httpd/conf.d/site1.ru.conf
```

Заполняем:

```
<VirtualHost *:80>
  ServerName site1.ru
  ServerAlias www.site1.ru
  DocumentRoot /web/site1.ru/www
  <Directory /web/site1.ru/www>
    Options FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>
  ErrorLog /web/site1.ru/logs/error.log
  CustomLog /web/site1.ru/logs/access.log common
</VirtualHost>
```

### 3.2.3. Перезапуск Apache

Теперь делаем restart apache:

```
systemctl restart httpd
```

Если возникли какие-то ошибки - смотрим лог apache /var/log/httpd/error\_log. Если все в порядке, то проверим, нормально ли настроен наш виртуальный хост. Для этого создадим в папке /web/**site1.ru**/www файл index.html следующего содержания:

```
mcedit /web/site1.ru/www/index.html
```

```
<h1>Апач настроен!</h1>
```

```
chown apache. /web/site1.ru/www/index.html
```

Дальше в винде правим файл hosts, чтобы обратиться к нашему виртуальному хосту. Добавляем туда строчку:

```
192.168.1.25 site1.ru
```

- где 192.168.1.25 ip адрес нашего веб сервера.

Теперь в браузере набираем адрес http://**site1.ru**. Если видим картинку:

## Апач настроен!

значит все правильно настроили. Если какие-то ошибки, то идем смотреть логи. Причем в данном случае не общий лог `httpd`, а лог ошибок конкретного виртуального хоста по адресу `/web/site1.ru/logs/error.log`.

Сразу же обращаю ваше внимание на настройку ротации логов виртуальных хостов. Часто бывает, что, если сразу не настроишь, потом забываешь. Но если сайт с хорошей посещаемостью, то логи будут расти стремительно и могут занять очень много места. Лучше настроить ротацию логов веб сервера сразу же после создания.

Чтобы настроить ротацию логов виртуальных хостов, необходимо отредактировать файл `/etc/logrotate.d/httpd`. Он создается во время установки `apache` и включает в себя настройку ротации стандартного расположения логов. А так как мы перенесли логи каждого виртуального хоста в индивидуальную папку, необходимо добавить эти папки в этот файл:

```
mcedit /etc/logrotate.d/httpd
```

```
/web/*/logs/*.log
/var/log/httpd/*log {
    missingok
    notifempty
    sharedscripts
    delaycompress
    postrotate
    /bin/systemctl reload httpd.service > /dev/null 2>/dev/null || true
    endscript
}
```

Мы добавили одну строку в самое начало файла. Теперь логи всех виртуальных хостов в папке `/web` будут ротироваться по общему правилу.

### 3.3. Graphviz

Для отображения диаграмм состояний сущностей используется ПО `Graphviz`.

Пример команды установки для `dnf`:

```
dnf install graphviz
```

### 3.4. СУБД PostgreSQL

В работе система использует базу данных СУБД PostgreSQL 12.

Для установки на операционную систему РЕД ОС использовалась следующая инструкция: [Установка Postgresql \(red-soft.ru\)](http://red-soft.ru)

Также, нужно установить дополнительный пакет `contrib`:

```
dnf install postgresql14-contrib.x86_64
```

Для администрирования сервера рекомендуется использовать ПО pgAdmin.  
Подробнее можно прочитать на официальном сайте проекта: <https://www.pgadmin.org>

База данных может развертываться как на том же сервере, что и web, так и на удаленном сервере. В случае развертывания БД на отдельном сервере для достижения максимальной скорости работы рекомендуется обеспечить канал между web-сервером и сервером базы данных с пропускной способностью не менее одного гигабита в секунду и минимальными задержками передачи пакетов.

### 3.4.1. Создание пользователя и схемы БД

Для работы системы требуется создания одного технологического пользователя и одной схемы БД.

Для корректного создания объектов базы данных требуется установить расширение «uuid-oss» в создаваемой схеме БД.

Предположим, мы хотим создать базу со следующими параметрами:

- Имя базы данных: upp\_db
- Имя пользователя: upp\_user
- Пароль пользователя: secure01

Пример скрипта для создания схемы БД и технологического пользователя (выполняется из-под суперпользователя):

```
CREATE DATABASE upp_db;  
CREATE USER upp_user WITH password 'secure01';  
GRANT ALL PRIVILEGES ON DATABASE upp_db TO upp_user;
```

Далее открываем базу данных upp\_db (либо выбрав данную БД в pgadmin, либо командой \connect upp\_db в psql) и выполняем следующие команды:

```
CREATE EXTENSION IF NOT EXISTS "uuid-oss";  
GRANT USAGE ON SCHEMA public TO upp_user;  
GRANT ALL PRIVILEGES ON SCHEMA public TO upp_user;  
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO upp_user;  
GRANT ALL PRIVILEGES ON ALL SEQUENCES IN SCHEMA public TO upp_user;
```

Все объекты БД системы в дальнейшем создаются под технологическим пользователем.

### 3.4.2. Создание базы данных

Для создания таблиц, представлений и необходимых первоначальных данных требуется выполнить под технологическим пользователем БД в созданной базе данных следующий скрипты, идущие в поставке с ПО:

- 01\_tables.sql
- 02\_trigger\_views.sql
- 03\_config\_data.sql
- 03\_weekends.sql

Примеры:



01\_tables.sql



02\_trigger\_views.sql



03\_config\_data.sql



03\_weekends.sql

### 3.4.3. Создание первого пользователя в системе.

Перед созданием пользователя добавим технологического пользователя для возможности админу создать именованного пользователя системы:

```
INSERT INTO users (login, pass, surname, name, second_name, state, speciality) VALUES(
    'user_admin',
    'a94a8fe5ccb19ba61c4c0873d391e987982fbbd3',
    'Администратор',
    'Пользователей',
    'Системы',
    0,
    'МЗК'
);
INSERT INTO user_roles (role_id, user_id) VALUES (
    'user_admin',
    (SELECT id FROM users WHERE login = 'user_admin')
);
```

Далее в conf.php на время требуется закомментировать следующую настройку:

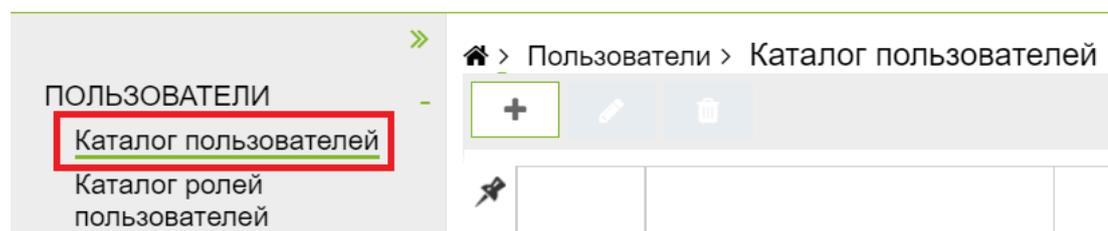
```
// 'two_factor_auth' => array(
// 'max_tries' => 5,
// 'expire_sec' => 180
// ),
```

Это нужно для того, чтобы на время выключить двухфакторную аутентификацию.

Далее требуется зайти на web-интерфейс системы и создать именованного администратора пользователей:

- Имя пользователя: user\_admin
- Пароль: test

После попадания в интерфейс требуется перейти в каталог пользователей:



И создать именованного пользователя:

» > Пользователи > Каталог пользователей > Свойства пользователя > ...

ПОЛЬЗОВАТЕЛИ  
КАТАЛОГ ПОЛЬЗОВАТЕЛЕЙ  
СВОЙСТВА ПОЛЬЗОВАТЕЛЯ  
Роли пользователя  
Каталог ролей пользователей  
НАСТРОЙКИ  
Документы для аккредитации  
Классификатор стран мира  
Виды грузов  
Виды транспорта  
Производственный календарь  
Системные переменные  
Новости  
КОНТРАГЕНТЫ  
Неподтвержденные контрагенты  
Каталог контрагентов

**ОСНОВНЫЕ ДАННЫЕ** РОЛИ ПОЛЬЗОВАТЕЛЯ

*Логин	Valdemar
*Фамилия	Захаренко
*Имя	Вальдемар
*Отчество	Вальдемарович
*Мобильный рабочий телефон	+7 (777) 777-77-77
Рабочий телефон	
Рабочий телефон (добавочный номер)	
Пароль	Задайте новый пароль или сгенерируйте нажав на ключ
E-Mail	
Заблокирован	<input type="checkbox"/>

После нажатия кнопки «Сохранить» требуется открыть созданного пользователя, назначить ему пароль:

Пароль

E-Mail

и на закладке «Роли» установить роль «user\_admin».

» > Пользователи > Каталог пользователей > Свойства пользователя > ...

ОСНОВНЫЕ ДАННЫЕ **РОЛИ ПОЛЬЗОВАТЕЛЯ**

Роли

Назначить роли Отозвать роли

Роль назначена	Код роли	Наименование роли
<input type="checkbox"/>	news_admin	Администратор новостей
<input checked="" type="checkbox"/>	user_admin	Администратор пользователей

После этого требуется выйти в каталог пользователей и удалить временного пользователя user\_admin.

Выход из системы произойдет автоматически.

Включаем назад в conf.php двухфакторную аутентификацию (снимаем комментарий со строк).

После указанных действий новый созданный именованный пользователь может создавать самостоятельно новых пользователей системы.

#### **3.4.4. Заливка в БД тестовых данных**



## 4. Конфигурация системы

Все остальные настройки системы производятся посредством модификации файла `conf.php`, находящегося в корне сайта.

Пример файла `conf.php` приложен к документации в виде файла `php.ini.example`



`php.ini.example`

### 4.1. Конфигурация пути для вызова ПО `graphviz`

Путь указывается в переменной `$graphviz`.

Для `centos` при установке по-умолчанию используется значение:

```
$graphviz = "/usr/bin/dot";
```

### 4.2. Дистрибутив системы

Дистрибутив системы состоит из файлов, находящихся в архиве `www-XXX.rar`, их требуется скопировать на сервер в папку `/web/site1.ru/www/` и заменить файл `conf.php` на `conf.php.example` (с пред настройками).

### 4.3. Настройки, хранящиеся в конфигурационном массиве `$sys_settings`.

Основные конфигурационные настройки системы задаются в массиве `$sys_settings`.

Настройки, не указанные ниже в данном документе, менять не следует – они заданы для возможности расширения системы в будущем, и их изменения могут привести к ошибкам в работе ПО.

#### 4.3.1. Параметры соединения с БД.

Параметры соединения с БД задаются в разделе `'connection'`.

Параметр	Необходимые действия
<code>connection.db</code>	Наименование схемы БД.
<code>connection.user</code>	Имя технологического пользователя БД.
<code>connection.password</code>	Пароль технологического пользователя для доступа к БД.
<code>connection.host</code>	IP-адрес или доменное имя сервера БД.

#### 4.3.2. Параметры двухфакторной аутентификации

Параметры задаются в разделах `'two_factor_auth'` и `'sms'`

Параметр	Необходимые действия
<code>two_factor_auth.max_tries</code>	Максимальное количество попыток входа с использование одноразового пароля. По истечению данного лимита пользователь будет заблокирован.

Параметр	Необходимые действия
two_factor_auth.expire_sec	Время в секундах действия высланного по SMS одноразового пароля.
<b>sms.code</b>	<b>Настройки значения формируемого одноразового кода</b>
sms.code.min_value	Минимальное значение кода
sms.code.max_value	Максимальное значение кода

#### 4.3.3. Параметры отправки SMS-сообщений

Параметры задаются в разделе 'sms'.

Параметр	Необходимые действия
<b>sms.smtp</b>	<b>Настройки для отсылки SMS-сообщения посредством протокола SMTP</b>
sms.smtp.host	Имя SMTP сервера
sms.smtp.username	Имя пользователя
sms.smtp.password	Пароль пользователя
sms.smtp.port	Порт SMTP сервера
sms.smtp.from_address	Адрес, от имени которого будет осуществляться отправка сообщений
sms.smtp.to_domain	Домен, используемый для формирования адреса отправки – XXXX@домен, где XXX-номер телефона
sms.mode	Тип отправки сообщений. Поддерживаемые значения: <ul style="list-style-type: none"> <li>• smtp</li> </ul>
sms.max_send_count	Максимальное количество попыток отправки сообщений процессом отправки SMS-сообщений.

#### 4.3.4. Параметры отправки email-сообщений

Параметры задаются в разделе 'email'.

Параметр	Необходимые действия
<b>email.from</b>	<b>Настройки имени, от лица которого производится отсылка.</b>
email.from.address	Адрес, от имени которого будет осуществляться отправка сообщений.
email.from.name	Имя, которое будет подставлено в адресе отправки.
email.smtp.host	Имя SMTP сервера
email.smtp.auth_type	Тип аутентификации
email.smtp.username	Имя пользователя

Параметр	Необходимые действия
email.smtp.password	Пароль пользователя
email.smtp.port	Порт SMTP сервера
email.smtp.secure	Тип отправки сообщений. Поддерживаемые значения: <ul style="list-style-type: none"> <li>• tls</li> <li>• пустое значение</li> </ul>
email.max_send_count	Максимальное количество попыток отправки сообщений процессом отправки email-сообщений.

#### 4.3.5. Параметры регистрации новых пользователей системы

Параметры задаются в разделе 'register\_new\_user'.

Параметр	Необходимые действия
register_new_user.max_tries	Максимальное количество попыток отправки одноразовых паролей на телефон и e-mail.
register_new_user.expire_sec	Время в секундах действия высланного по SMS одноразового пароля.
register_new_user.password_generating	<b>Настройки генерации пароля пользователя.</b> <b>Пароль всегда начинается с 1 буквы, далее идут в произвольном порядке указанные в параметрах символы, буквы и спец-символы.</b>
register_new_user.password_generating.chars_count	Количество символов a-Z в пароле.
register_new_user.password_generating.digits_count	Количество символов 0-9 в пароле.
register_new_user.password_generating.special_count	Количество спец-символов в пароле: -=!@#\$\$%^&*()_+.,/<>?:[]{}

Если у пользователя нет возможности регистрации нового пользователя через sms оповещения или email, то нужно ее выключить, для этого в файле conf.php изменить значение test\_mode'=>true:

```
'register_new_user' => [
  'sms' => [
    'stat_path' => "../temp/sms",
    'max_tries' => 10,
    'expire_sec' => 300,
  ],
  'email' => [
    'stat_path' => "../temp/email",
    'max_tries' => 5,
    'expire_sec' => 300,
  ],
  'password_generating' => [
    'chars_count' => 5,
    'digits_count' => 2,
    'special_count' => 1,
  ],
],
'test_mode'=>true,
```

## 5. Конфигурация файловой системы web-сервера

Для корректной работы web-сервера необходимо дать права на доступ и запись в каталог `webint/templates_c` для пользователя `apache` под операционной системой RedOS.

## 6. Подсистема запуска фоновых процессов системы на стороне сервера

В комплект поставки системы входят набор скриптов, запускаемых из-под главного скрипта **crontab.sh**, который требуется запускать планировщиком событий RedOS (crontab) раз в 1 минуту.



01-send\_email.php

Данные скрипт обеспечивают периодическое выполнение процедур, необходимых для корректной работы системы, а также отсылку SMS и E-Mail уведомлений системой.

Настройки планировщика:

```
* * * * * /home/develop/www/crontab/crontab.sh /home/develop/www/glencore
```

Где в данном примере:

[/home/develop/www/crontab/crontab.sh](#) – полный путь к скрипту

[/home/develop/www/glencore](#) – полный путь к каталогу с web

Скрипт в работе использует часть функционала системы из каталог web-сервера.

## 7. Состав дистрибутива

Дистрибутив системы состоит из следующих файлов:

Параметр	Необходимые действия
crontab-XXX.rar	Дистрибутив подсистемы запуска фоновых процессов.
sql-XXX.rar	Скрипт для создания таблиц БД, представлений и заполнения справочников.
www-XXX.rar	Дистрибутив web-сервиса.
conf.php.example	Пример настройки системы. Файл следует переименовать в conf.php и положить в папку с web-сервисом.

## 8. Установка обновления системы.

Обновление присылается в виде следующих файлов:

Параметр	Необходимые действия
crontab-XXX.rar	Обновление подсистемы запуска фоновых процессов.
sql-XXX.rar	Обновления скриптов для создания таблиц БД, представлений и заполнения справочников.
www-XXX.rar	Обновление web-сервиса.
change.txt.sql	SQL-скрипт для внесения изменения в БД.

### 8.1. Порядок установки обновлений

При установке обновлений следует действовать по следующему алгоритму:

1. Остановить web-сервер apache;
2. Сделать резервную копию файла crontab (`crontab -l >file.crontab`);
3. Остановить подсистему выполнения фоновых процессов (`crontab -r`);
4. Сделать резервную копию базы данных;
5. Сделать резервную копию каталогов с web-сервисом и подсистемой запуска фоновых процессов;
6. Выполнить все команды из скрипта «change.txt.sql» по одной, убедившись, что нет ошибок выполнения;
7. Выполнить скрипт 02\_trigger\_views.sql для пересоздания представлений и служебных процедур, убедиться, что нет ошибок;
8. Удалить из каталога с web-сервисом все файлы, кроме «conf.php» и содержимого папки «docs»;
9. Развернуть присланное обновление из файла «www-XXX.rar» в каталог с web-сервисом;
10. Удалить из каталога с подсистемой запуска фоновых процессов все файлы;
11. Развернуть присланное обновление из файла «crontab-XXX.rar» в каталог с подсистемой запуска фоновых процессов;
12. Запустить подсистему выполнения фоновых процессов основываясь на сохраненной копии в файле «file.crontab» (`crontab file.crontab`);
13. Запустить web-сервер apache.