

**ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«ЦЕНТР СОЦИАЛЬНО-ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ»**

(ООО "СоцИнформТех")

УТВЕРЖДАЮ:

Генеральный директор

Т. В. Некрасова
« 10 » января _____ 2023 г.



**Программа для ЭВМ
Программный комплекс
«Ранняя помощь»**

Руководство администратора по установке программного комплекса

Листов 16

г. Тула

Оглавление

1. Общее описание	3
2. Техническое описание ПК.....	4
3. Программные и аппаратные требования	5
4. Установка автоматизированной системы	6
5. Установка Nginx.....	7
5.1 Настройка Firewall	9
5.2 Проверка доступа.....	9
5.3 Настройка Nginx	9
6. Установка PostgreSQL	13
7. Установка ПК:.....	16

1. Общее описание

Программный комплекс «Ранняя помощь» (далее - ПК) предназначен для автоматизации процессов при предоставлении органами исполнительной власти субъектов Российской Федерации, органами местного самоуправления и организациями-операторами информации об исполнении возложенных на них индивидуальной программой ранней помощи (далее – ИПРП) мероприятий с целью формирования единой базы данных.

2. Техническое описание ПК

Программный комплекс построен по трехзвенной архитектуре на Mono software platform, и функционирует в среде Linux совместимых операционных систем.

Программно-техническое решение ПК реализовано на основе трехзвенной архитектуры, включающей в себя:

- уровень хранения и обработки данных: PostgreSQL Server;
- уровень приложений: NGINX и Mono software platform;
- уровень представления данных: браузер на клиентской машине.

Доступ пользователей к ПК осуществляется по принципу «тонкого клиента».

Установка может быть выполнена по одному из следующих вариантов:

1) Одномашинная установка

Все три компонента – PostgreSQL Server, NGINX, Mono и ПК – должны быть установлены на одной машине. В этом случае система установлена и работает на одной машине. Локальная сеть не требуется.

2) Установка в небольших сетях (до 200 рабочих мест)

В сети организуется сервер, на который устанавливаются серверные компоненты – PostgreSQL Server, NGINX, Mono и ПК.

Клиентский компонент - браузер – находится на каждом рабочем месте в сети, и с него происходит подключение к серверу.

3) Установка в ЦОДе (200-2000 рабочих мест)

В ЦОДе разворачивается 2 сервера:

- сервер приложений – на нем устанавливается NGINX, Mono и ПК
- сервер баз данных PostgreSQL Server.

Клиентский компонент - браузер – находится на каждом рабочем месте в сети, и с него происходит подключение к серверу приложений.

3. Программные и аппаратные требования

Так как технические требования изменяются постоянно, то ниже приведены требования в обобщенном виде по состоянию на 2021 год. Для получения актуальных требований обратитесь к разработчику.

ПК обеспечивает работу специалистов с базой данных на следующем техническом обеспечении:

1) Сервер приложений (веб-сервер):

- 8 ядер, 16 Гб RAM, свободного места на диске от 250Гб;
- Linux OS (Linux- совместимые операционные системы такие как Ubuntu Server 20.04, Debian 10, Astra linux), Nginx;

2) Сервер баз данных:

- 16 ядер, 32 Гб RAM;
- HDD объемом 1Тб SSD, обеспечивающий производительность не ниже: последовательное чтение/запись: 400Мбит/с, случайное многопоточное чтение/запись (блоки 4K): 80Mb/s;
- Linux OS (Linux- совместимые операционные системы такие как Ubuntu Server 20.04, Debian 10, Astra linux);

3) Автоматизированные рабочие места пользователей:

- 32-разрядный (x86) или 64-разрядный (x64) процессор с тактовой частотой 2 ГГц или выше;
- 2 Гб (для 32-разрядной системы) или 4 Гб (для 64-разрядной системы) оперативной памяти;
- 16 Гб (для 32-разрядной системы) или 20 Гб (для 64-разрядной системы) свободного места на жестком диске;
- используемые браузеры: Internet Explorer 10, FireFox 64, Opera 57, Chrome 61 и более поздние версии.

Дополнительно рекомендуется выделить в сети сервер хранения резервных копий баз данных: около 5Тб дискового пространства.

Канал связи между серверами не менее 10 Гбит/с. Канал связи от клиентской машины до сервера не менее 50 Мбит/с.

4. Установка автоматизированной системы

Программное обеспечение, используемое при эксплуатации системы:

- **Операционная система**, предназначенная для организации серверов в вычислительных сетях с современным оборудованием: Linux (Linux- совместимые операционные системы такие как Ubuntu Server 20.04, Debian 10, Astra linux).
- **Nginx** - бесплатный программный веб-сервер, служащий для балансировки нагрузки на сайт.
- **Mono** - реализация технологии .NET Framework от Microsoft в виде открытого программного обеспечения
- **PostgreSQL** – сервер баз данных. Система работает на версиях сервера 12 и 13.
- **Браузер**: Internet Explorer 10, FireFox 64, Opera 57, Chrome 61 и более поздние версии.

Устанавливать ПО следует строго в описанной последовательности:

- Установка Linux
- Установка NGINX
- Установка Mono
- Установка PostgreSQL Server

Установка Системы производится из терминала.

В качестве каталога установки будем использовать /var/www/aspnet/

Создание каталога

```
mkdir /var/www
```

```
mkdir /var/www/aspnet
```

5. Установка Nginx

Nginx (или **Engine X**) - это бесплатный, свободный и мощный HTTP- и прокси-сервер с открытым исходным кодом и архитектурой на основе обработки событий. Он написан на языке программирования C и может работать в Unix-подобных системах.

Перед любой установкой всегда обновляйте локальный репозиторий, чтобы убедиться, что вы загружаете последнюю версию программного обеспечения.

Далее все действия будут выполняться от имени root если вы работаете под отдельным пользователем, необходимо перед командами добавлять sudo.

Внесем изменения в список репозиториев.

```
vi /etc/apt/sources.list
```

удалим "s" в <http://download.astralinux.ru/astra/stable/orel/repository/> orel
main contrib non-free

Загрузим списки пакетов из репозиториев и обновим их.

```
apt update
```

Установка mc (Midnight Commander файловый менеджер).

```
apt install mc
```

На данном этапе мы можем установить пакет.

```
apt-get install mono-complete
```

Проверим версию пакета mono которую мы установили.

```
mono --version
```

Должна быть 4

Начинаем обновление пакета mono.

```
sudo apt install apt-transport-https dirmngr gnupg ca-certificates
```

Добавьте репозиторий Mono в вашу систему.

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys  
3FA7E0328081BFF6A14DA29AA6A19B38D3D831EF
```

```
echo "deb https://download.mono-project.com/repo/debian stable-stretch
main" | sudo tee /etc/apt/sources.list.d/mono-official-stable.list
```

Временно заблокируем доступ к официальному репозиторию Astra.

```
# deb http://download.astralinux.ru/astra/stable/orel/repository/ orel main
contrib non-free
```

Обновим списки пакетов.

```
apt update
```

Установим их более новые версии.

```
apt upgrade
```

Проверим версию пакета mono .

```
mono --version
```

Версия должна быть не меньше 6.12. Теперь необходимо установить nginx. Добавим соответствующий репозиторий

```
deb [trusted=yes] http://nginx.org/packages/debian/ stretch nginx
```

Обновим списки пакетов.

```
apt update
```

Запускаем установку nginx.

```
apt install nginx
```

Чтобы запустить сервис Nginx, выполните.

```
systemctl start nginx
```

Затем необходимо добавить программу в автозагрузку.

```
systemctl enable nginx
```

Если все было сделано правильно, то, открыв адрес сервера, на который вы устанавливали Nginx, вы увидите страницу по умолчанию.

Не забывайте удалять ненужные сторонние репозитории после того, как установка из них завершена.

5.1 Настройка Firewall

Nginx включает в себя служебные файлы `firewalld`, которые указывают, что служба использует порты 80 (HTTP) и 443 (HTTPS) для веб-трафика. Поэтому вам необходимо открыть и включить эти порты, чтобы разрешить постоянный доступ.

Откройте порт HTTP и HTTPS с помощью команд:

```
sudo firewall-cmd --permanent --zone=public --add-
service=http --add-service=https
sudo firewall-cmd --reload
sudo firewall-cmd --list-services --zone=public
```

5.2 Проверка доступа

Используйте `netstat`, чтобы вывести список всех открытых портов и убедиться, что вы успешно открыли 80 и 443:

```
netstat -tulpn
```

Найдите IP-адрес и скопируйте его. Чтобы увидеть IP-адрес вашего сервера, введите в терминале следующую команду:

```
ip addr
```

Затем откройте веб-браузер и вставьте IP-адрес (или имя домена) в строку URL. Откроется страница приветствия Nginx, подтверждающая, что вы успешно установили и настроили сервер.

5.3 Настройка Nginx

Здесь и далее используем имя компьютера `webserver_name` и путь к комплексу `/var/www/webserver_name/`

Приведем конфиг к такому виду:

```
/etc/nginx/nginx.conf
user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}
```

```

http {

    ##
    # Basic Settings
    ##

    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2; # Dropping SSLv3, ref: POODLE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##

    access_log /var/log/nginx/access.log;
    error_log /var/log/nginx/error.log;

    ##
    # Gzip Settings
    ##

    gzip on;

    # gzip_vary on;
    # gzip_proxied any;
    # gzip_comp_level 6;
    # gzip_buffers 16 8k;

```

```

# gzip_http_version 1.1;
# gzip_types text/plain text/css application/json application/javascript text/xml
application/xml application/xml+rss text/javascript;

##
# Virtual Host Configs
##

include /etc/nginx/conf.d/*.conf;
include /etc/nginx/sites-enabled/*;
}

```

```

#mail {
#   # See sample authentication script at:
#   # http://wiki.nginx.org/ImapAuthenticateWithApachePhpScript
#
#   # auth_http localhost/auth.php;
#   # pop3_capabilities "TOP" "USER";
#   # imap_capabilities "IMAP4rev1" "UIDPLUS";
#
#   server {
#       listen    localhost:110;
#       protocol  pop3;
#       proxy     on;
#   }
#
#   server {
#       listen    localhost:143;
#       protocol  imap;
#       proxy     on;
#   }
#}

```

Создаем файл `mccedit /etc/nginx/conf.d/aspnet.conf` с содержимым:

```

server {
    listen 80;
    server_name localhost;
    access_log /var/log/nginx/access.log;
    root /var/www/webserver_name/;
    client_max_body_size 20M;
    charset UTF-8;

    location / {

```

```

        index index.html index.htm default.aspx Default.aspx;
        fastcgi_index default.aspx;
        fastcgi_pass 127.0.0.1:9000;
        include /etc/nginx/fastcgi_params;
    }
}

```

В файл `/etc/nginx/fastcgi_params` добавляем следующие строки:

```

fastcgi_param PATH_INFO      "";
fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;

```

создаем папку `/var/www/webserver_name/`

```

mkdir -p /var/www/webserver_name
chown www-data:www-data /var/www/webserver_name/
chmod 777 /var/www/webserver_name/

```

В эту папку устанавливаем комплекс.

Сделаем проверку.

```
nginx -t
```

Ошибок быть не должно.

Затем делаем рестарт:

```
nginx -s reload
```

Установим `fastcgi-mono-server4` и `mono-xsp4`

```
apt-get install mono-fastcgi-server4 mono-xsp4
```

6. Установка PostgreSQL

Данная инструкция разработана для Astra Linux Special Edition 1.6
Версию ОС можно проверить с помощью команды.

```
lsb_release -a
```

или

```
cat /etc/astra_version
```

Внимание крайне не рекомендуется использовать для администрирования пользователя root. В системе необходимо создать отдельного пользователя и осуществлять установку и дальнейшее администрирование, используя эту учетную запись. Пользователь, выполняющий установку, должен иметь права на выполнение команды `sudo`. Дополнительную информацию можно найти в "Документации ОС", "Руководство администратора, часть 1" в разделе "1. АДМИНИСТРИРОВАНИЕ ОС".

Рекомендуется PostgreSQL не ниже версии 12. Рассмотрена установка и настройка PostgreSQL версии 14.

Для установки необходимо подключить официальные репозитории PostgreSQL.

Astra Linux SE 1.6 построена на ОС Debian 9 (Stretch). Для добавления репозитория выполните:

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

Установите ключ проверки подлинности:

```
wget -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo  
apt-key add -
```

В результате установки может возникнуть ошибка "ОШИБКА: Сертификат «www.postgresql.org» неизвестно кем выпущен." Для ее устранения необходимо:

1. Проверить, что установлен пакет `ca-certificates`.

```
dpkg -s ca-certificates
```

Если пакет не установлен, установить:

```
sudo apt install ca-certificates
```

2. Проверить наличие корневого сертификата DST Root CA X3.

```
awk -v cmd='openssl x509 -noout -subject' ' /BEGIN/{close(cmd)};{print |  
cmd}' < /etc/ssl/certs/ca-certificates.crt | grep "DST Root CA X3"
```

У сертификата DST Root CA X3 истек срок действия 30 сентября 2021 года.

Его наличие в системе может приводить к проблемам при доступе к сайтам, использующим сертификат от Let's Encrypt.

Для исключения сертификата выполните команды:

```
sudo sed -i 's/mozilla/DST_Root_CA_X3.crt!/mozilla/DST_Root_CA_X3.crt/' /etc/ca-certificates.conf
sudo update-ca-certificates
```

После этого повторите установку ключа, проблема не должна возникнуть

```
wget -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
```

Обновите список репозиториев:

```
sudo apt update
```

Для установки PostgreSQL выполните команду.

```
sudo apt install -t stretch-pgdg postgresql-14
```

Запустите БД и включите автозагрузку, после чего проверьте статус.

```
sudo systemctl enable postgresql
sudo systemctl start postgresql
systemctl status postgresql
```

Для проверки работоспособности переключитесь на пользователя postgres, автоматически созданного при установке:

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

Проверьте запрос из консоли, запросив версию:

```
psql -c "SELECT version();"
```

Войдите в консоль PostgreSQL, выполнив команду psql и, например, запросите список баз данных, командой \l

```
psql
```

Для настройки PostgreSQL войдите в консоль PostgreSQL. Внимание, обязательно установите пароль для пользователя postgres

```
\password postgres
```

По умолчанию доступ к серверу баз данных PostgreSQL осуществляется только с локального хоста.

```
ss -tunelp | grep 5432
```

Для разрешения доступа из сети нужно отредактировать файл postgresql.conf.

```
sudo nano /etc/postgresql/14/main/postgresql.conf
```

Найдите строку

```
#listen_addresses = 'localhost'
```

Раскомментируйте ее и замените 'localhost' на '*'.

```
listen_addresses = '*'
```

Перезапустите сервис.

```
sudo systemctl restart postgresql
```

Проверьте, что порт прослушивается для всех адресов

```
ss -tunelp | grep 5432
```

Отредактируйте файл pg_hba.conf

```
echo "host all all all md5" | sudo tee -a /etc/postgresql/14/main/pg_hba.conf
```

Перезапустите сервис

```
sudo systemctl restart postgresql
```

7. Установка ПК:

ПК может быть установлен только, если на компьютере уже установлены вышеперечисленные компоненты (Операционная система Linux Server, Nginx, Mono, PostgreSQL сервер).

Скачайте с FTP-сервера разработчика и разархивируйте файл инсталлятора. Содержимое папки скопируйте в каталог /var/www/webserver_name/ созданный в ходе настройки Nginx (см. п. 5.3 данной инструкции).

Далее нужно перезапустить nginx воспользовавшись командой от имени рута:

```
service nginx restart
```

Также нужно перезапустить Mono:

```
systemctl restart mono
```

После установки ПК функционирует корректно.