

Для установки проекта требуется пустой сервер с минимальной конфигурацией Intel Celeron 2 ГГц, 2 ядра, 8 ГБ DDR3, 2x480 Gb HDD (Raid0) либо аналогичной конфигурации с установленной OS Centos 7.last. Если вы планируете устанавливать базу данных на отдельном сервере - рекомендованная минимальная конфигурация аналогична основному серверу.

Сотрудник, занимающийся установкой системы должен обладать достаточной квалификацией и опытом работы с linux / centos, знать основы развертывания и работы web окружения (apache, nginx, mysql).

Проект устанавливается в три этапа:

1. Установка окружения - установка на пустой сервер всех необходимых программ и баз данных для корректного функционирования проекта.
2. Перенос данных из облачной версии проекта (только при внедрении системы), либо с предыдущего сервера (рабочий перенос).
3. Подготовка и запуск проекта.

Установка окружения на пустой сервер

1. Действия выполнять под root
2. Установите Ansible (<https://www.ansible.com/>) и скопируйте в директорию его установки файлы из директории /serversetup проекта (<https://yadi.sk/d/rjeSnPkC8e12xA>). Вы можете скачать архив по ссылке, либо запросить архив у компании MACRO по адресу webmaster@macrocrm.ru.
3. Требуемая для работы системы базы данных (macro_main, macro_lib) и пользователи (системный пользователь macro, под которым будут создаваться директории и файлы системы, пользователи mysql macro_main и root будут созданы автоматически)
4. Задайте в `serversetup/playbook.yml` следующие настройки:
 - mysql_root_password: пароль пользователя root к базе данных macro
 - macro_main_password: пароль пользователя macro_main базы данных macro
 - macropassword: пароль системного пользователя macro
 - macro_path: путь до директории в которой необходимо разместить рабочие файлы
 - mysql_datadir: путь до директории в которой будут размещаться рабочие файлы базы данных
 - macro_domain: домен, под которым будет работать система.
5. Запустите команду `ansible-playbook --connection=local --inventory 127.0.0.1, playbook.yml` и дождитесь окончания установки
6. Система среди прочего установит и сконфигурирует MySQL, Apache и Nginx (+letsencrypt), настроит cron задачи.
7. Версии устанавливаемого программного обеспечения указаны в `serversetup/playbook.yml`
8. Если вы планируете устанавливать базу данных на отдельном сервере, необходимо самостоятельно установить и сконфигурировать (см файл `serversetup/playbook.yml`) на отдельном сервере mysql требуемой версии.
9. Убедитесь, что директории `/var/lib/nginx` `/var/cache/nginx` принадлежат пользователю, под которым работает Nginx.
10. Поставьте корневой директории пользователя права 0755
11. Отредактируйте файл `/etc/php.ini`, найдите в нем следующие параметры и поставьте им значения:
 - a. `expose_php = Off`
 - b. `post_max_size = 50M`
 - c. `upload_max_filesize = 50M`
12. Перезапустите сервис httpd: `systemctl restart httpd`

Установка системы

а) Перенос данных с облачного сервера на новый сервер компании.

Выполняется силами компании MACRO при внедрении системы и первичной установке проекта на сервере компании.

Внимание! Особенности системы не предусматривают самостоятельную первичную установку проекта на сервер компании.

Инструкция для сотрудников MACRO:

1. Добавьте публичный SSH ключ облачного сервера на новый сервер (инструкция по созданию публичного ключа: <https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-centos7-ru>)
2. На рабочем сервере создать дамп _lib базы данных:


```
mysqldump --opt --skip-lock-tables _lib | zip > _lib-date +` \%d_ \%m_ \%Y_ \
%H_ \%M` .sql.zip
```
3. Скопируйте дамп на новый сервер


```
scp -P 22 _lib-01_07_2021_05_41.sql.zip username@new.server:/home/
username/
```
4. На рабочем сервере создайте резервную копию данных переносимой компании: в рабочей директории проекта под пользователем проекта выполните


```
php console.php company:export {id компании}
```

 резервная копия будет располагаться в директории `/storage/export/{url компании}/`
5. Скопируйте резервную копию на новый сервер.

Инструкция для сотрудников компании, занимающихся установкой системы на новый сервер:

1. Предполагается наличие на новом сервере резервной копии рабочей базы данных, базы _lib, а также архива файлов.
2. Импортируйте дамп _ib на новом сервере


```
unzip -p _lib-01_07_2021_05_41.sql.zip | mysql -u macro_main -p macro_lib
```
3. Импортируйте дамп рабочей базы данных _main из резервной копии
4. Скопируйте в директорию `/public_html/upload/` файлы из резервной копии

б) Перенос данных проекта с вашего старого сервера на новый сервер

В случае, если вы не делаете snapshot вашего сервера, то при переезде на новый сервер после подготовки окружения необходимо сделать:

1. На предыдущем сервере сделайте бэкап баз данных `macro_*`
2. На новом сервере импортируйте эти бэкапы
3. Перенесите директорию с файлами `/public_html/upload/` со старого сервера на новый

Подготовка и запуск проекта

1. Создайте на новом сервере ключ `ssh-keygen -t rsa` с дефолтными установками, без пароля.
2. Для добавления публичного ключа в репозиторий обновлений (`git@bitbucket.org:macrocrm/macroencrypted.git`) необходимо написать письмо на адрес `webmaster@macrocrm.ru`, приложить ключ и указать для какого сервера (домен, ip-адрес) необходимо предоставить доступ.
3. В каталоге пользователя `<userdir>/data/www` выполните `git clone git@bitbucket.org:macrocrm/macroencrypted.git`.
4. Создайте директорию `<userdir>/data/www/public_html/config/prod`
 - a. Если вы переносите с облачной версии, тогда скопируйте туда из `public_html/config/default/` файлы:
 - i. `config.php`
 - ii. `db*.php`
 - iii. `deploy.php`
 - iv. `email.php`
 - v. `phinx.php`
 - vi. `security.php`
 - b. Если вы переносите данные со старого сервера просто скопируйте директорию `<userdir>/data/www/public_html/config/prod` со старого сервера.
 - c. Если вы планируете устанавливать базу данных на отдельный сервер, просто укажите в файлах конфига `db*.php` IP адрес этого сервера и настройте на сервер базы данных доступ.
5. Если вы переносите с облачной версии - укажите в этих файлах актуальные текущему серверу настройки (домены, доступы к базе данных, почте и т.д.).
6. В файле `deploy.php` в настройке `deploy_local` укажите в качестве сервера обновлений `macrocrm.ru` и ключ обновлений, полученный от разработчиков MACRO. Наличие данного ключа позволяет проводить разработчикам удаленное систематическое обновление системы.
7. Сохранить в корне рабочей директории (`/data/www/`) файл `macrolic.txt` с лицензионным ключом продукта внутри. Файл `macrolic.txt` предоставляет компания MACRO вместе с лицензионным соглашением.
8. Далее выполните команды от имени `root`

```
localectl set-locale LANG=en_US.UTF-8
```

#в адресной строке в корне проекта

```
npm install
```

#если еще не установлен, то

```
npm install -g gulp-cli
```
9. В директории проекта (`data/www`) выполните команды от имени рабочего пользователя системы

```
composer install
```

```
composer deploy
```
10. Если вы переносите данные со старого сервера, перенесите директорию с файлами `<userdir>/data/www/public_html/upload/` со старого сервера на новый. В случае переноса проекта из облачной версии, скопируйте из резервной копии файлы в указанную директорию.

11. Проект должен открыться по требуемому доменному имени.