

Офис в Москве:
109147, Москва, Таганская улица, 17–23
+7(495)258–71–64 (доб. 20891)
algorithm@sinara-group.com
dtdesk.ru

**Инструкция по установке ПО
автоматизированной системы эксплуатации
технического обслуживания и ремонта
подвижного состава DTtechnoRS для
пользователей, которая размещается на сайте**

Сведения о документе

Версия документа:	1.0
Дата:	26.12.24
Аудитория документа:	Конечные пользователи системы

Оглавление

Введение.....	3
Шаг 1: Обновление пакетов Ubuntu на сервере.....	4
Шаг 2: Установка зависимостей Odoo 17.....	4
Шаг 3. Установка плагинов NPM и CSS	4
Шаг 4. Установка Wkhtmltopdf для генерации pdf-отчетов	4
Шаг 4: Установка и настройка PostgreSQL.....	5
Шаг 5: Создание пользователя Odoo	5
Шаг 6: Установка Odoo из исходного кода.....	5
Шаг 7: Настройка файла конфигурации Odoo.....	6
Шаг 8: Создание файла systemd-юнита Odoo	7
Шаг 9: Клонирование репозитория DTtechnoRS.....	8
Шаг 10: Клонирование репозитория DTtechnoRS.....	8

Введение

Документ содержит описание порядка установки экземпляра программного обеспечения автоматизированной системы эксплуатации технического обслуживания и ремонта подвижного состава DTtechnoRS, предоставленного для проведения экспертной проверки, и включает информацию по установке и настройке платформы.

Шаг 1: Обновление пакетов Ubuntu на сервере

Всегда рекомендуется обновлять и модернизировать существующие пакеты Ubuntu перед установкой любого приложения.

Откройте сеанс терминала и выполните следующие команды, чтобы сначала обновить и модернизировать вашу систему.

```
sudo apt update
sudo apt upgrade
```

Шаг 2: Установка зависимостей Odoo 17

Поскольку Odoo написан на Python, вам необходимо установить все зависимости Python. Выполните следующие команды, чтобы установить все зависимости.

```
sudo apt-get install -y python3-pip python3-dev python3-venv libxml2-dev libxslt1-dev
zlib1g-dev libsasl2-dev libldap2-dev build-essential libssl-dev libffi-dev libmysqlclient-dev
libjpeg-dev libpq-dev libjpeg8-dev libbcms2-dev libblas-dev libatlas-base-dev -y
```

Шаг 3. Установка плагинов NPM и CSS

После установки зависимостей мы установим управление пакетами для языка программирования JavaScript:

```
sudo apt-get install -y npm
sudo ln -s /usr/bin/nodejs /usr/bin/node
sudo npm install -g less less-plugin-clean-css
sudo apt-get install -y node-less
```

Шаг 4. Установка Wkhtmltopdf для генерации pdf-отчетов

Установите Wkhtmltopdf, инструмент командной строки с открытым исходным кодом для преобразования HTML в отчеты PDF. Он необходим для генерации отчетов в Odoo. Выполните следующие команды для установки Wkhtmltopdf.

```
Wget https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6.1-3/wkhtmltox_0.12.6.1-3.jammy_amd64.deb
sudo apt install ./wkhtmltox_0.12.6.1-3.jammy_amd64.deb
```

Шаг 4: Установка и настройка PostgreSQL

Odoo работает с базой данных PostgreSQL. Установка PostgreSQL в Ubuntu довольно проста. Просто выполните следующую команду, чтобы установить PostgreSQL на свой сервер.

```
sudo apt install postgresql -y
```

После установки запустите и включите службу PostgreSQL с помощью следующей команды.

```
sudo systemctl start postgresql && sudo systemctl enable postgresql
```

Чтобы проверить статус, выполните следующую команду.

```
sudo systemctl status postgresql
```

Теперь создайте пользователя PostgreSQL, сопоставив имя пользователя Ubuntu, которое мы создали на шаге 3

```
sudo su - postgres -c "createuser -s odoo17"
```

Команда переключается на пользователя Postgres и выполняет команду createuser с флагом -s, чтобы создать нового пользователя PostgreSQL с именем odoo17.

Шаг 5: Создание пользователя Odoo

Хотя вы можете запускать все команды как пользователь root, всегда используйте определенного пользователя, это защитит ваш сервер от взлома. Создайте пользователя с отдельным домашним каталогом для установки Odoo. Используйте следующую команду для создания нового пользователя (имя пользователя должно быть идентичным, в нашем случае мы назовем его odoo17)

```
sudo useradd -m -U -r -d /opt/odoo17 -s /bin/bash odoo17
```

(Вам необходимо использовать sudo, если вы работаете в Ubuntu или любом другом пользователе)

Установите пароль пользователя для odoo17

```
sudo passwd odoo17
```

Добавьте созданного пользователя в группу sudo

```
sudo adduser odoo17 sudo
```

Шаг 6: Установка Odoo из исходного кода

Для установки Odoo сначала переключитесь на пользователя, которого вы создали ранее (шаг 5).

```
sudo su odoo17
```

Теперь клонируйте исходный код Odoo 17 из его официального репозитория.

```
git clone https://www.github.com/odoo/odoo --depth 1 --branch 17.0 /opt/odoo17/odoo17
```

Теперь измените каталог на исходный каталог Odoo.

```
cd /opt/odoo17
```

Теперь создайте виртуальную среду Python для установки пакетов Python, которые потребуются Odoo.

```
python -m venv odoo17-venv
```

```
source odoo17-venv/bin/activate
```

```
pip install --upgrade pip
```

```
pip install wheel
```

```
pip install -r odoo17/requirements.txt
```

Удалите библиотеку psycopg2 и установите psycopg2-binary

```
pip uninstall psycopg2
```

```
pip install psycopg2-binary
```

Дополнительно требуется установить библиотеки:

```
pip install docxtpl XlsxWriter openpyxl
```

После завершения задач установки виртуальной среды Python деактивируйте виртуальное окружение и создайте каталоги аддонов odoo и файл журнала логов Odoo.

```
deactivate
```

```
mkdir /opt/odoo17/odoo17-custom-addons
```

```
chown -R odoo17:odoo17 /opt/odoo17/odoo17-custom-addons
```

```
sudo mkdir -p /var/log/odoo17
```

```
sudo touch /var/log/odoo17/odoo17.log
```

```
sudo chown -R odoo17:odoo17 /var/log/odoo17
```

Шаг 7: Настройка файла конфигурации Odoo

Теперь создайте файл конфигурации Odoo. Этот шаг наиболее важен для работы Odoo в соответствии с вашими требованиями. Создайте файл конфигурации в каталоге «etc», выполнив следующую команду.

```
sudo touch /etc/odoo17.conf
```

```
sudo nano /etc/odoo17.conf
```

Теперь добавьте следующие строки в файл.

```
[options]
```

```
; Database operations password:
```

```
admin_passwd = admin_passwd
```

```
db_host = False
```

```
db_port = False
```

```
db_user = odoo17
```

```
db_password = False
```

```
addons_path = /opt/odoo17/odoo17/addons, /opt/odoo17/odoo17-custom-addons
```

```
logfile = /var/log/odoo17/odoo17.log
```

```
log_level = debug
```

Не забудьте заменить admin_passwd на надежный пароль.

Теперь обновите владельца и разрешение файла конфигурации, который вы только что создали.

```
sudo chown odoo17:odoo17 /etc/odoo17.conf
```

Эта команда изменяет владельца и группу файла /etc/odoo17.conf на odoo17:odoo17, гарантируя, что пользователь odoo17 и группа odoo17 будут иметь соответствующие разрешения для доступа и изменения файла конфигурации.

Шаг 8: Создание файла systemd-юнита Odoo

Мы создадим файл systemd для простого управления службой odoo. Выполните следующую команду, чтобы создать файл systemd.

```
sudo touch /etc/systemd/system/odoo17.service
```

Теперь откройте файл systemd, выполнив команду

```
sudo nano /etc/systemd/system/odoo17.service
```

Добавьте следующие строки

```
[Unit]
```

```
Description=odoo17
```

```
After=network.target postgresql@16-main.service
```

```
[Service]
```

```
Type=simple
```

```
SyslogIdentifier=odoo17
```

```
PermissionsStartOnly=true
```

```
User=odoo17
```

```
Group=odoo17
```

```
ExecStart=/opt/odoo17/odoo17-venv/bin/python3 /opt/odoo17/odoo17/odoo-bin -c  
/etc/odoo17.conf
```

```
StandardOutput=journal+console
```

```
[Install]
```

WantedBy=multi-user.target

Теперь перезагрузите daemon, запустите и включите службу Odoo

sudo systemctl daemon-reload

sudo systemctl start odoo17 && sudo systemctl enable odoo17

Теперь проверьте статус службы и выполните следующую команду

sudo systemctl status odoo17

Теперь службы Odoo17 запущены и работают. Для доступа к Odoo вам нужно включить порт 8069 (по умолчанию Odoo использует порт 8069, также измените порт, если хотите).

http://”your server ip”:8069 – по данному адресу в любом браузере будет открываться ваш экземпляр DTtechnoRS.

Шаг 9: Клонирование репозитория DTtechnoRS

Перейдите в каталог дополнительный аддонов Odoo

cd /opt/odoo17/odoo17-custom-addons

Распакуйте переданный архив репозитория

unzip sinara_maintenance_repairs-main.zip

Шаг 10: Клонирование репозитория DTtechnoRS

Перейдите по адресу <http://”YOUR SERVER IP”:8069>

Создайте базу данных. Для этого в открывшейся форме вам необходимо заполнить следующие поля:

Master Password – Пароль, который вы указали в конфигурационном файле odoo17.conf.

Email и Password – Учетные данные администратора

Language – язык по умолчанию



Master Password	<input type="password" value="....."/>
Database Name	<input type="text" value="odoo"/>
Email	<input type="text" value="admin@admin.ru"/>
Password	<input type="password" value="..."/>
Phone Number	<input type="text"/>
Language	<input type="text" value="Russian / русский язык"/>
Country	<input type="text" value="Russian Federation"/>
Demo Data	<input type="checkbox"/>

[Create database](#) [or restore a database](#)

Перейдите в Приложения и активируйте приложение Repairs Management.

Приложения

Приложения Обновление списка приложений Применение запланированных обновлений Модуль импорта

Приложения

Категории: Все, Продажи, Услуги, Бухгалтерия, Склад, Производство, Сайт, Маркетинг, HR

Repairs management sinara_maintenance_repairs Активировать Узнать больше	Продажи sale_management Активировать Узнать больше
CRM crm Активировать Узнать больше	MRP II mrp_workorder Узнать больше Обновить
Бухгалтерия account_accountant Узнать больше Обновить	Знание knowledge Узнать больше Обновить
Проект project Активировать Узнать больше	eCommerce website_sale Активировать Узнать больше
Расписания timereport_mid	Расходы hr_expense

После выполнения всех шагов, описанных выше у вас будет готовый к работе программного обеспечения автоматизированной системы эксплуатации технического обслуживания и ремонта подвижного состава DTtechnoRS