|  |  |  |
| --- | --- | --- |
| **Установка сервера Apache** | | |
| **$ sudo apt update**  **$ sudo apt upgrade** | | Получение и установка последних обновлений системы |
| **$ sudo apt install apache2** | | Установка Апач |
| **$ sudo systemctl enable apache2** | | Добавление Апач в автозагрузку.  Для ручного запуска Апач использовать команды:  **$ sudo systemctl start apache2**  или  **$ sudo service apache2 start** |
| **$ sudo a2enmod expires**  **$ sudo a2enmod headers**  **$ sudo a2enmod rewrite**  **$ sudo a2enmod ssl** | | Включение в Апач дополнительных модулей.  Для отключения использовать команды:  **$ sudo a2dismod expires**  **$ sudo a2dismod headers**  **$ sudo a2dismod rewrite**  **$ sudo a2dismod ssl** |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| <VirtualHost \*:80>  ServerName ***host2.*** ***staffcontrol.pp.ua***  ServerAlias ***www.*** ***staffcontrol.pp.ua***  ServerAdmin webmaster@localhost  DocumentRoot ***/var/www/host2.localhost/blog/public***  <Directory "***/var/www/host2.localhost/blog/public***">  Options FollowSymLinks  AllowOverride All  Order allow,deny  Allow from all  </Directory>  ErrorLog ${APACHE\_LOG\_DIR}/error.log  CustomLog ${APACHE\_LOG\_DIR}/access.log combined    </VirtualHost>  # vim: syntax=apache ts=4 sw=4 sts=4 sr noet | | Содержимое файла настройки виртуальных хостов.  /etc/apache2/hosts-available/***host2.ubuntu.conf***  или  /etc/apache2/sites-available/***host2.ubuntu.conf***  ServerName ***адрес сервера***  ServerAlias ***альтернативный адрес сервера***  ServerAdmin ***эл.почта админа (неважно)***  DocumentRoot ***корневая папка с файлами сервера***  \* При использовании laravel – ***/blog/public*** , где «blog» это название проекта, «**public» это стандартная папка** laravel;  - иначе указывать папку где находится файл index.php или index.html |
| **$ sudo a2ensite host2.ubuntu** | | Активировать виртуальный хост.  Для деактивации использовать команду:  **$ sudo a2dissite host2.ubuntu** |
| 127.0.0.1 localhost  127.0.0.1 host2.ubuntu  127.0.0.1 www.host2.ubuntu  127.0.1.1 ubuntu-VirtualBox  # The following lines are desirable for IPv6 capable hosts  ::1 ip6-localhost ip6-loopback  fe00::0 ip6-localnet  ff00::0 ip6-mcastprefix  ff02::1 ip6-allnodes  ff02::2 ip6-allrouters  192.168.0.104 host1.ubuntu  192.168.0.104 www.host1.ubuntu  192.168.0.104 staffcontrol.pp.ua  192.168.0.104 www. staffcontrol.pp.ua | | Содержимое файла настройки доменных имен.  */etc/****hosts*** |
| **$ ifconfig** | | проверить сетевые настроки |
| # Copyright (c) 1993-2009 Microsoft Corp.  #  # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.  #  # This file contains the mappings of IP addresses to host names. Each  # entry should be kept on an individual line. The IP address should  # be placed in the first column followed by the corresponding host name.  # The IP address and the host name should be separated by at least one  # space.  #  # Additionally, comments (such as these) may be inserted on individual  # lines or following the machine name denoted by a '#' symbol.  #  # For example:  #  # 102.54.94.97 rhino.acme.com # source server  # 38.25.63.10 x.acme.com # x client host  # localhost name resolution is handled within DNS itself.  # 127.0.0.1 localhost  # ::1 localhost    192.168.0.105 host1.ubuntu #For Apache  192.168.0.105 host2.ubuntu #For Apache  192.168.0.105 staffcontrol.pp.ua #For Apache  192.168.0.105 www.host1.ubuntu #For Apache  192.168.0.105 www.host2.ubuntu #For Apache  192.168.0.105 www. staffcontrol.pp.ua #For Apache | | Содержимое файла настройки доменных имен на клиенте (WINDOWS).  *c:\Windows\System32\drivers\etc\hosts* |
| **Обеспечение безопасности Apache с помощью Let's Encrypt в Ubuntu - Пока не работает** | | |
| **$ sudo apt install certbot python3-certbot-apache** | | установить программное обеспечение Certbot  **python3-certbot-apache** – плагин, который интегрирует Certbot с Apache, позволяя автоматизировать получение сертификата и настройку HTTPS на веб-сервере с помощью одной команды |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| **$ sudo ufw status** | | проверить статус брандмауера |
| **$ sudo certbot --apache**  **Enter email address (used for urgent renewal and security notices) (Enter 'c' to**  **cancel):** [*zolotukhinpv@gmail.com*](mailto:zolotukhinpv@gmail.com)  **(A)gree/(C)ancel:***A* | | изменение конфигурации Apache  An unexpected error occurred:  The server will not issue certificates for the identifier :: Error creating new order :: Cannot issue for "host1.ubuntu": Domain name does not end with a valid public suffix (TLD) (and 5 more problems. Refer to sub-problems for more information.) |
| **Создание самоподписанных сертификатов SSL для Apache в Ubuntu** | | |
| **$ sudo a2enmod ssl** | | Активируйте mod\_ssl  **$ sudo a2dismod ssl** |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| **$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt**  **Country Name (2 letter code) [XX]:***UA*  **State or Province Name (full name) [Some-State]:***Donetsk*  ***Locality Name (eg, city) []:****Bakhmut*  **Organization Name (eg, company) [Internet Widgits Pty Ltd]:***noorgname*  **Organizational Unit Name (eg, section) []:***NON*  **Common Name (e.g. server FQDN or YOUR name) []:***host3.ubuntu*  **Email Address []:***zolotukhinpv@gmail.com* | | создать ключ SSL и файлы сертификата   * openssl: это инструмент командной строки, предназначенный для создания сертификатов, ключей и других файлов OpenSSL и управления ими. * req -x509: указывает, что мы хотим использовать управление запросами на подписание сертификатов X.509 (CSR). X.509 — это инфраструктура открытых ключей, используемая стандартами SSL и TLS для управления ключами и сертификатами. * -nodes: предписывает OpenSSL пропустить опцию защиты нашего сертификата кодовой фразой. Для чтения этого файла при запуске сервера без вмешательства пользователя нам потребуется Apache. Кодовая фраза предотвратит это, поскольку в ином случае нам пришлось бы вводить ее после каждого перезапуска. * -days 365: эта опция устанавливает период действия сертификата. Здесь мы устанавливаем срок действия в один год. Многие современные браузеры отклоняют любые сертификаты, срок действия которых превышает один год. * -newkey rsa:2048: указывает, что мы хотим сгенерировать новый сертификат и новый ключ одновременно. Мы не создали требуемый ключ для подписи сертификата на предыдущем шаге, и поэтому нам нужно создать его вместе с сертификатом. Часть rsa:2048 предписывает создать ключ RSA длиной 2048 бит. * -keyout: эта строка указывает OpenSSL, где разместить генерируемый файл закрытого ключа. * -out: указывает OpenSSL, где разместить создаваемый сертификат. |
| <VirtualHost \*:***443***>  SSLEngine on  SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt  SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key | | Изменим порт с 80 на 443 содержимом файла настройки виртуальных хостов.  /etc/apache2/hosts-available/***host2.ubuntu.conf***  Добавляем в содержимое файла настройки виртуальных хостов.  /etc/apache2/hosts-available/***host2.ubuntu.conf***  или  /etc/apache2/sites-available/***host2.ubuntu.conf*** |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| <VirtualHost \*:8**0**>  ServerName staffcontrol.pp.ua  Redirect / https:// staffcontrol.pp.ua /  </VirtualHost> | | Создайте в конце файла еще один блок VirtualHost для запросов через порт 80. Используйте директиву ServerName для привязки вашего доменного имени или IP-адреса. Затем используйте Redirect для перенаправления всех запросов на SSL VirtualHost. Не забудьте добавить косую черту в конце |
|  | |  |
|  | | |
| **Установка PHP** | | |
| **$ sudo apt update**  **$ sudo apt upgrade** | | Получение и установка последних обновлений системы |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| **$ sudo apt install php7.4** | | Установка PHP. Версия PHP может быть другой (новей). |
| **$ sudo apt install libapache2-mod-php7.4** | | Дополнительно устанавливается для работы с Апач |
| **$ sudo apt install php-curl**  **$ sudo apt install php-json**  **$ sudo apt install php-cgi**  **$ sudo apt install php-gd**  **$ sudo apt install php-zip**  **$ sudo apt install php-mbstring**  **$ sudo apt install php-xml**  **$ sudo apt install php-xmlrpc** | | Установка дополнительных модулей PHP |
| **$ sudo ufw allow in 80/tcp** | | Добавить порт 80 в исключения брандмауэра |
| **$ sudo vi /etc/php/7.4/apache2/php.ini** | | файл настройки PHP.  /etc/php/7.4/apache2/***php.ini*** |
| **Установка Postgresql+phppgadmin** | | |
| **$ sudo apt update**  **$ sudo apt upgrade** | | Получение и установка последних обновлений системы |
| **$ sudo systemctl restart apache2** | | Перезагрузка Апач |
| **$ sudo apt install postgresql-12** | | Установка Postgresql. Версия Postgresql может быть другой (новей). |
| **$ sudo apt install php7.4-pgsql**  **$ sudo apt install postgresql-contrib**  **$ sudo apt install phppgadmin** | | Установка дополнительных модулей PHP для работы с Postgresql.  Установкаphppgadmin. |
| **$ sudo su**  **su postgres** | | Смена пользователя на **postgres – системного пользователя** Postgresql |
| **psql** | | Входим в консоль Postgresql |
| **ALTER USER postgres WITH PASSWORD `rtub`;** | | Смена пароля пользователя **postgres на «rtub»** |
| **exit** | | Выходим с консоли Postgresql |
| **Su ubuntu** | | Смена пользователя на **ubuntu** |
| **$ sudo passwd postgres**  **Новый пароль: rtub**  **Повторите ввод нового пароля: rtub**  **Passwd: пароль успешно обновлен** | | Смена пароля пользователя **postgres** |
| **listen\_addresses = '\*'**  **password\_encryption = md5** | | Содержимое файла настройки доменных имен.  */etc/postgresql/12/main/postgresql.conf* |
| **# Database administrative login by Unix domain socket**  **local all postgres peer**  **# TYPE DATABASE USER ADDRESS METHOD**  **# "local" is for Unix domain socket connections only**  **local all all peer**  **# IPv4 local connections:**  **host all all 127.0.0.1/32 md5**  **# IPv6 local connections:**  **host all all ::1/128 md5**  **# Allow replication connections from localhost, by a user with the**  **# replication privilege.**  **local replication all peer**  **host replication all 127.0.0.1/32 md5**  **host replication all ::1/128 md5**  ***host all all 192.168.0.106/24 md5***  ***host all all 0.0.0.0/0 md5***  ***host all all 0.0.0.0/0 trust*** | | Содержимое файла настройки доменных имен.  */etc/postgresql/12/main/pg\_hba.conf*  *Необходимо чтобы получить доступ с удаленной машины* |
| **#Require local**  **Order deny,allow**  **Deny from all**  **Allow from 127.0.0.1**  **Allow from 192.168.0.0/16**  **Allow from 172.16.0.0/12**  **Allow from 10.0.0.0/8** | | Содержимое файла настройки доменных имен.  */etc/apache2/conf-available/phppgadmin.conf* |
| **'default' => env('DB\_CONNECTION', 'pgsql'),**    **'pgsql' => [**  **'driver' => 'pgsql',**  **'url' => env('DATABASE\_URL'),**  **'host' => env('DB\_HOST', '127.0.0.1'),**  **'port' => env('DB\_PORT', '5432'),**  **'database' => env('DB\_DATABASE', 'staff'),**  **'username' => env('DB\_USERNAME', 'postgres'),**  **'password' => env('DB\_PASSWORD', 'rtub'),**  **'charset' => 'utf8',**  **'prefix' => '',**  **'prefix\_indexes' => true,**  **'schema' => 'public',**  **'sslmode' => 'prefer',**  **],** | | настройки в файле **/var/www/host2.localhost/blog/config/database**.php |
| **APP\_NAME=Laravel**  **APP\_ENV=local**  **APP\_KEY=base64:YnTcqnVQ3uXt4khsy9weZGfLt1S4lffRjpshQsjDTvE=**  **APP\_DEBUG=true**  **APP\_URL=http://localhost**  **LOG\_CHANNEL=stack**  **LOG\_LEVEL=debug**  **DB\_CONNECTION=pgsql**  **DB\_HOST=localhost**  **DB\_PORT=5432**  **DB\_DATABASE=staff**  **DB\_USERNAME=postgres**  **DB\_PASSWORD=rtub**  **BROADCAST\_DRIVER=log**  **CACHE\_DRIVER=file**  **QUEUE\_CONNECTION=sync**  **SESSION\_DRIVER=file**  **SESSION\_LIFETIME=120**  **MEMCACHED\_HOST=127.0.0.1**  **REDIS\_HOST=127.0.0.1**  **REDIS\_PASSWORD=null**  **REDIS\_PORT=6379**  **MAIL\_MAILER=smtp**  **MAIL\_HOST=mailhog**  **MAIL\_PORT=1025**  **MAIL\_USERNAME=null**  **MAIL\_PASSWORD=null**  **MAIL\_ENCRYPTION=null**  **MAIL\_FROM\_ADDRESS=null**  **MAIL\_FROM\_NAME="${APP\_NAME}"**  **AWS\_ACCESS\_KEY\_ID=**  **AWS\_SECRET\_ACCESS\_KEY=**  **AWS\_DEFAULT\_REGION=us-east-1**  **AWS\_BUCKET=**  **PUSHER\_APP\_ID=**  **PUSHER\_APP\_KEY=**  **PUSHER\_APP\_SECRET=**  **PUSHER\_APP\_CLUSTER=mt1**  **MIX\_PUSHER\_APP\_KEY="${PUSHER\_APP\_KEY}"**  **MIX\_PUSHER\_APP\_CLUSTER="${PUSHER\_APP\_CLUSTER}"** | | настройки в файле **/var/www/host2.localhost/blog/.env** |
|  | | |
|  | |  |
| **APP\_NAME=Laravel**  **APP\_ENV=local**  **APP\_KEY=**  **APP\_DEBUG=true**  **APP\_URL=http://localhost**  **LOG\_CHANNEL=stack**  **LOG\_LEVEL=debug**  **DB\_CONNECTION=pgsql**  **DB\_HOST=127.0.0.1**  **DB\_PORT=5432**  **DB\_DATABASE=staff**  **DB\_USERNAME=postgres**  **DB\_PASSWORD=rtub**  **BROADCAST\_DRIVER=log**  **CACHE\_DRIVER=file**  **QUEUE\_CONNECTION=sync**  **SESSION\_DRIVER=file**  **SESSION\_LIFETIME=120**  **MEMCACHED\_HOST=127.0.0.1**  **REDIS\_HOST=127.0.0.1**  **REDIS\_PASSWORD=null**  **REDIS\_PORT=6379**  **MAIL\_MAILER=smtp**  **MAIL\_HOST=mailhog**  **MAIL\_PORT=1025**  **MAIL\_USERNAME=null**  **MAIL\_PASSWORD=null**  **MAIL\_ENCRYPTION=null**  **MAIL\_FROM\_ADDRESS=null**  **MAIL\_FROM\_NAME="${APP\_NAME}"**  **AWS\_ACCESS\_KEY\_ID=**  **AWS\_SECRET\_ACCESS\_KEY=**  **AWS\_DEFAULT\_REGION=us-east-1**  **AWS\_BUCKET=**  **PUSHER\_APP\_ID=**  **PUSHER\_APP\_KEY=**  **PUSHER\_APP\_SECRET=**  **PUSHER\_APP\_CLUSTER=mt1**  **MIX\_PUSHER\_APP\_KEY="${PUSHER\_APP\_KEY}"**  **MIX\_PUSHER\_APP\_CLUSTER="${PUSHER\_APP\_CLUSTER}"** | | настройки в файле **/var/www/host2.localhost/blog/**.env |
| **1 Установка Laravel** | | |
| **$ sudo apt install php-cli**  **$ sudo apt install unzip**  **$ sudo apt install php-tokenizer**  **$ sudo apt install php-cbcmath** | |  |
| **$ sudo apt install composer** | | Устанавливаем Composer |
| **composer global require laravel/installer** | | Загружаем laravel |
| **cd /var/www/host2.localhost/** | | Переходим в папку, в которую будет устанавливаться laravel |
| **composer create-project --prefer-dist laravel/laravel blog** | | устанавливаем laravel в папку blog (можно использовать любую) |
| **$ sudo chown -R www-data:www-data /var/www** | | открыть доступ к файлам проекта |
| **$ sudo php artisan key:generate** | | генерация ключа – после подключения файла .env слетел ключ |
| **установка DEBUG панели** | | |
| **$ sudo composer require barryvdh/laravel-debugbar --dev** | | установка **DEBUG** панели |
| **Laravel** ***экспорт в excel*** | | |
| **cd /var/www/host3.localhost/blog** | | Переходим в папку, в которую устанавлен laravel |
| **composer require maatwebsite/excel** | | Устанавливаем пакет ***maatwebsite/excel*** |
| **php artisan vendor:publish --provider="Maatwebsite\Excel\ExcelServiceProvider"** | | копировать файл конфигурации: |
| **Maatwebsite\Excel\ExcelServiceProvider::class,** | | добавить ServiceProvider к массиву поставщиков в config/app.php |
| **'Excel' => Maatwebsite\Excel\Facades\Excel::class,** | | Вы можете использовать фасад для более короткого кода. Добавьте это к вашим псевдонимам: |
| **2 Laravel** ***авторизация https://codeguida.com/post/678*** | | |
| **$ sudo composer require laravel/ui**  **$ php artisan ui vue --auth** | | обеспечивает быстрый способ формирования всех маршрутов и представлений, необходимых для аутентификации.  !!! Только на чистый проект |
| **$ sudo composer require laravel/socialite** | | добавляем пакет для социальной аутоинтефикации |
|  | |  |
|  | |  |
| **GIT (github.com)** | | |
| **github.com->Create a new repository** | | Создаем новый удаленный репозиторий на сайте github.com, перед этим регистрируемся – **Mad-Deuce/test.git** |
| **Сервер (LINUX)**  **$ sudo apt install git**  **$ sudo git config --global user.name “MAD-DEUCE”**  **$ sudo git config --global user.email** [**zolotukhinpv@gmail.com**](mailto:zolotukhinpv@gmail.com)  **$ git config --list --show-origin** | | Установка GIT  Настройка GIT – Имя глобального пользователя GIT;  Настройка GIT – почта глобального пользователя GIT;  Просмотр настроек; |
| **Сервер (LINUX)**  **cd /var/www/host2.localhost/blog**  **$ sudo git init**  **$ sudo git add .**  **$ sudo git commit -m “first”**    **$ sudo git branch -M main**  **$ sudo git remote add origin https://github.com/Mad-Deuce/test.git**  **$ sudo git push -u origin main** | | переходим в каталог с проектом;  инициализируем GIT контроль в данной папке;  добавляем все файлы и каталоги в отслеживание;  создаем первый коммит (снимок) с коментом **“first”;**  создаем основную ветку**;**  подключаем удаленный репозиторий **test.git, Mad-Deuce** – имя пользователяgithub.com;  отправляем коммит на удаленный репозиторий; |
| **Клиент (Windows)**  **PHPStorm->VCS->Get from Version Control- >GitHub/ Mad-Deuce->test.git->clone**  **PHPStorm->Git->Commit**  **PHPStorm->Git->Push** | | загружаем файлы проекта с удаленного репозитория на сайте github.com в среду **PHPStorm** на другой машине;  фиксируем изменения – коммит (снимок) – в среде **PHPStorm** на другой машине;  отправляем коммит (снимок) на удаленный репозиторий на сайте github.com; |
| **Сервер (LINUX)**  **cd /var/www/host2.localhost/blog**  **$ sudo git pull origin** | | переходим в каталог с проектом;  загружаем изменения – коммит (снимок)– с удаленного репозитория на сайте github.com на сервер **(LINUX)** |
| **cd /var/www/host2.localhost/blog**  **$ sudo git add .**  **$ sudo git commit -m “other comment”**  **$ sudo git push -u origin main** | | **Изменения сделаны на сервере-добавлен файл**  переходим в каталог с проектом;  добавляем все файлы и каталоги в отслеживание;  создаем коммит (снимок) с коментом **“ other comment ”;**  отправляем коммит на удаленный репозиторий; |
| **Клиент (Windows)**  **PHPStorm->Git->Pull** | | загружаем изменения – коммит (снимок)– с удаленного репозитория на сайте github.com на **Клиент (Windows) в PHPStorm** |
| **Установка FTP сервера**  - для связки PHPStorm на Винде и виртуального сервера | | |
| **$ sudo apt update**  **$ sudo apt upgrade** | | Получение и установка последних обновлений системы |
| **$ sudo apt-get install vsftpd** | | установка VSFTPD-сервера |
| **$ sudo systemctl start vsftpd** | | Запуск ftp сервера **vsftpd** |
| **$ sudo systemctl enable vsftpd** | | Добавление ftp сервера **vsftpd** в автозагрузку. |
| **$ sudo ufw allow 20/tcp**  **$ sudo ufw allow 21/tcp**  **$ sudo ufw status** | | Открываем порты 20 и 21 – Если у вас установлен фаервол ufw, а такая ситуация возникает, когда вы пытаетесь установить FTP на Ubuntu Server, нужно открыть порты 20 и 21 для нормальной работы |
| **$ sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.original** | | делаем бэкап оригинального файла конфигурации |
| **sudo adduser phpstormuser**  **Новый пароль : 200107**  **Повторите ввод нового пароля : 200107** | | создадим нового пользователя,  и пароль для него |
| **cd /var/www/host2.localhost**  **$ sudo chown -R phpstormuser:phpstormuser blog** | | переходим в каталог с хоста;  назначаем владельцем всех файлов и папок пользователя **phpstormuser** и группу **phpstormuser** |
|  | | |
|  | |  |
| anonymous\_enable = NO  local\_enable = YES  write\_enable = YES    chroot\_local\_user=YES    local\_root=/var/www/host3.localhost/StaffControl/  local\_umask = 022  dirmessage\_enable = YES  xferlog\_enable = YES  xferlog\_std\_format=YES  connect\_from\_port\_20 = YES  listen=YES  listen\_ipv6=NO  pam\_service\_name=vsftpd | | Настройки **vsftpd** - файл «/etc/vsftpd.conf»  – отключаем анонимный вход;  – разрешаем использовать имена локальных пользователей для входа;  – для авторизованных пользователей разрешаем команды, позволяющие изменять файловую систему;  – данная опция позволяет поместить локального пользователя в chroot() «заточение», выше которого (по дереву каталогов) он не сможет подняться    – установим значение **umask** для новых файлов, создаваемых по FTP; **пока неактивно**  – включаем сообщение о необходимости выбрать каталог после регистрации;  – записывать в лог файл все транзакции по передаче файлов и использовать стандартный формат лога;  – использовать порт 20 для передачи данных вместо случайного, это нужно для нормальной работы фаервола;  – указываем, что нужно ожидать входящих соединений;  – listen и listen\_ipv6 должны быть разные  – использовать PAM-библиотеки |
| **$ sudo systemctl restart vsftpd** | | Перезапуск ftp сервера **vsftpd** |
| **Клиент (Windows)**  **PHPStorm->File->Settings->Build, Execution, Deployment-> Deployment->”+”->FTP:**  **New server name: *host3.ubuntu***  ***Connection->***  **Host: *192.168.0.105* Port: *21***  **User name: *phpstormuser***  **Password: *200107***  **Root path: /**  **Mapping->**  **Local path: C:\Users\admin\PhpstormProjects\StaffControl**  **Deployment path: /**  **Web path: /** | |  |
|  | |  |
| **BUGs** | | |
| sudo chmod -R 777 /var/www | touch(): Unable to create file /var/www/host3.localhost/StaffControl/storage/framework/laravel-excel/laravel-excel-Jsw4hiRMJsxi4rbYtjxEbb3R1qBFmJlo.html because Permission denied | |
| php artisan config:clear | trying to access array offset on value of type null  если вы ввели свой client\_id и secret\_id config/services.php, то запустите | |
| **Other** | | |
| **sudo apt-get install meld** | | Установка приложения для сравнивания файлов и каталогов |