**Новый тестовый сервер.**

Особенности:

* Выделенный сервер (4 ядра, 10Gb памяти, 320Gb) – по производительности «шустрее» чем существующий itech-test. Объем диска в будущем можно увеличить.
* Связка nginx + apache + php-fpm – самое быстрое решение на данный момент, без потери совместимости с .htaccess файлами
* Выделенный ip адрес – не будет путаницы с портами (21 – ftp, 22-ssh/sftp, 80 – http, 443 –https, 3306 mysql)
* Сервер настроен для Использования CMS Bitrix, потому что чаще всего используется она.
* Возможность программистам самим создавать для себя dev-зоны из тестовых.
* Для экономии места и идентичности контента, общие папки создаются «симлинками».

**Создание пользователя**

Создание пользователя осуществляется с помощью команды user\_add.sh, запускается через sudo.

Использование: sudo user\_add.sh имя\_пользователя.

Пример: sudo user\_add.sh myusername – создаст пользователя myusername

С помощью команды создается пользователь со всеми необходимыми папками и настройками, доступами к базе данных.

В выводе команды описывается логин пользователя и его пароль к mysql/ssh.

По умолчанию используется командная оболочка bash, и доступны все утилиты, установленные в системе:

* Системы управления версиями: git, svn, hg
* Файловый менеджер: mc
* Текстовые редакторы: vim, nano
* …и можем поставить другой софт по желанию

Создавать пользователей могут только администраторы сервера, так же можем выдать права руководителям отделов.

**Удаление пользователя**

Удаление пользователя осуществляется с помощью команды user\_del.sh, запускается через sudo.

Использование: sudo user\_add.sh имя\_пользователя.

Пример: sudo user\_del.sh myusername – удалит пользователя myusername

**ВНИМАНИЕ: удаляются все данные пользователя – настройки, папки, dev-зоны, базы данных.**

 **Создание dev-зон из тестового сайта.**

Создание dev-зоны пользователя осуществляется с помощью команды create\_dev\_site.sh, запускается через sudo.

Команда доступна самим пользователям, т.е. могут создавать dev-зоны сами из присутствующих на сервере тестовых сайтов.

Команда выполняется в интерактивных окнах.

Пример использования: sudo create\_dev\_site.sh

1. 1 Шаг – выбор тестового сайта из которого делается dev-зона.

Выбираются только сайты на CMS Bitrix присутствующие на сервере.

Для создания dev-зон других CMS требуется обратится в ITECH.service.

Выбор по списку сайтов осуществляется стрелочками вверх и вниз, помечается выбранный домен клавишей «пробел».

После выбора и нажатия ОК происходит копирование файлов, прогресс можно наблюдать внизу. Пока не красиво, но что ни будь придумаем.

1. 2 Шаг – создание базы данных.

После копирования файлов будет предложено создать базу данных.

Если не создавать базу данных, то на dev-зона будет использовать основную базу тестового сайта.

1. Если выбрано создать отдельную базу данных.

То будет предложено выбрать ее имя.

Базы данных создаются с префиксом имени пользователя в начале.

Желательно вводить только английские буквы, база создастся, но не известно, как на это отреагируют скрипты.

После ввода имени базы, база данных будет скопирована и автоматически прописана в конфигурационные файлы CMS Bitrix.

1. Ну и последнее окно, подтверждающее что все выполнено.

Особенности:

* Для экономии места и идентичности контента не копируется папка upload и bitrix (при наличии папки local).
* Не производятся работы с гитом, т.е. не переключаются ветки, не создаются новые. GIT будет идентичен тестовому.

**Смена версии php для dev-зоны**

Смена версии php пользователя осуществляется с помощью команды change\_dev\_php\_version.sh, запускается через sudo.

Команда доступна самим пользователям, т.е. могут создавать dev-зоны сами из присутствующих на сервере тестовых сайтов.

Команда выполняется в интерактивных окнах.

Пример использования: sudo change\_dev\_php\_version.sh

1. 1 Шаг – выбираем dev-зону для которой будем менять версию php

1. Выбираем версию php

**Прочитать обязательно.**

* Все скрипты «заточены» под Bitrix. Для создания dev-зон не Bitrix-сайтов нужно обращаться в ITECH.service.
* Скрипты не умеют создавать и «рулить» мультисайтовостью у битрикса, поэтому за настройкой так же в ITECH.service
* Пользователи не могут менять пароли, что бы не было соблазна поставить слабые пароли, так же от этого зависит автоматика, т.к. на пароли завязаны некоторые вещи. За сменой пароля в ITECH.service
* Пользователи не могут сами менять параметры php, так как завязано на fpm, и из-за неправильно установленного параметра может все сломаться.
* Работа с тестовыми сайтами (с основными, не dev-зонами) происходит только по доступам к основным тестовым сайтам, под логином пользователя прав не хватит.

Как реализовать эти вещи придумаем, бета версия же.