

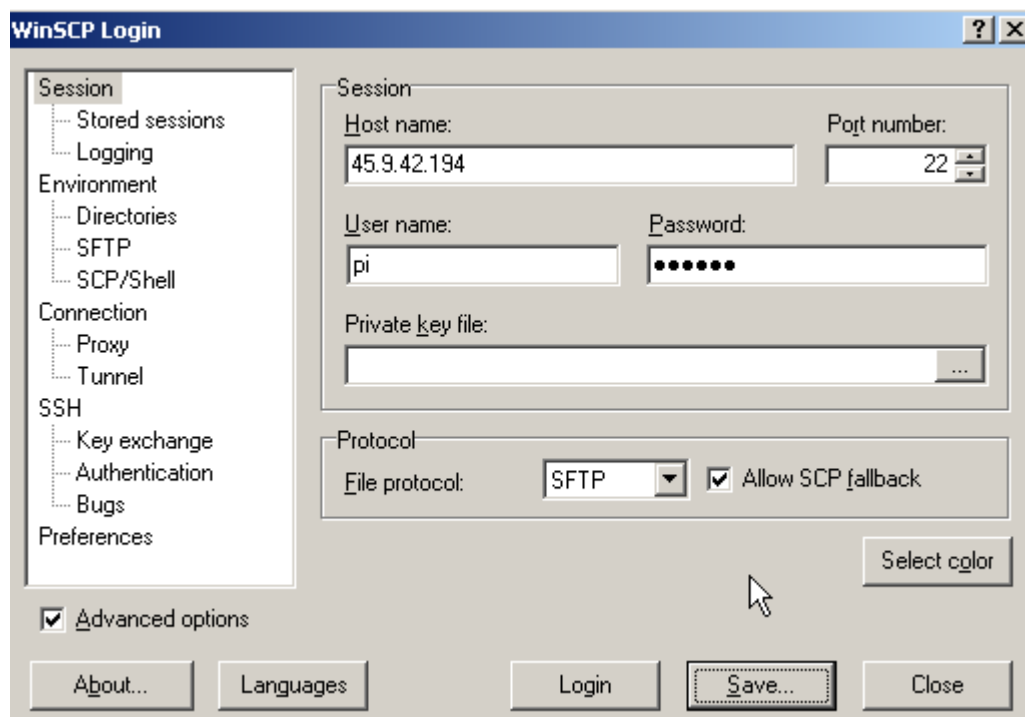
Конструктор бота 'creator' позволяет построить структурированного бота с помощью самого бота, в приложении Телеграмм, не влезая в коды скрипта. Для этого у бота есть система нехитрых команд, которые позволяют:

- добавлять/удалять кнопки бота различного типа;
- редактировать кнопки различного типа;
- перемещать кнопки вверх/вниз;
- перемещать кнопки на другой уровень вместе со всеми ветками;
- создавать и редактировать текст, выводимый по нажатию кнопки;
- загружать/удалять файлы различного типа;
- выдавать статистику посещений;
- рассылать всем подписчикам важную информацию;

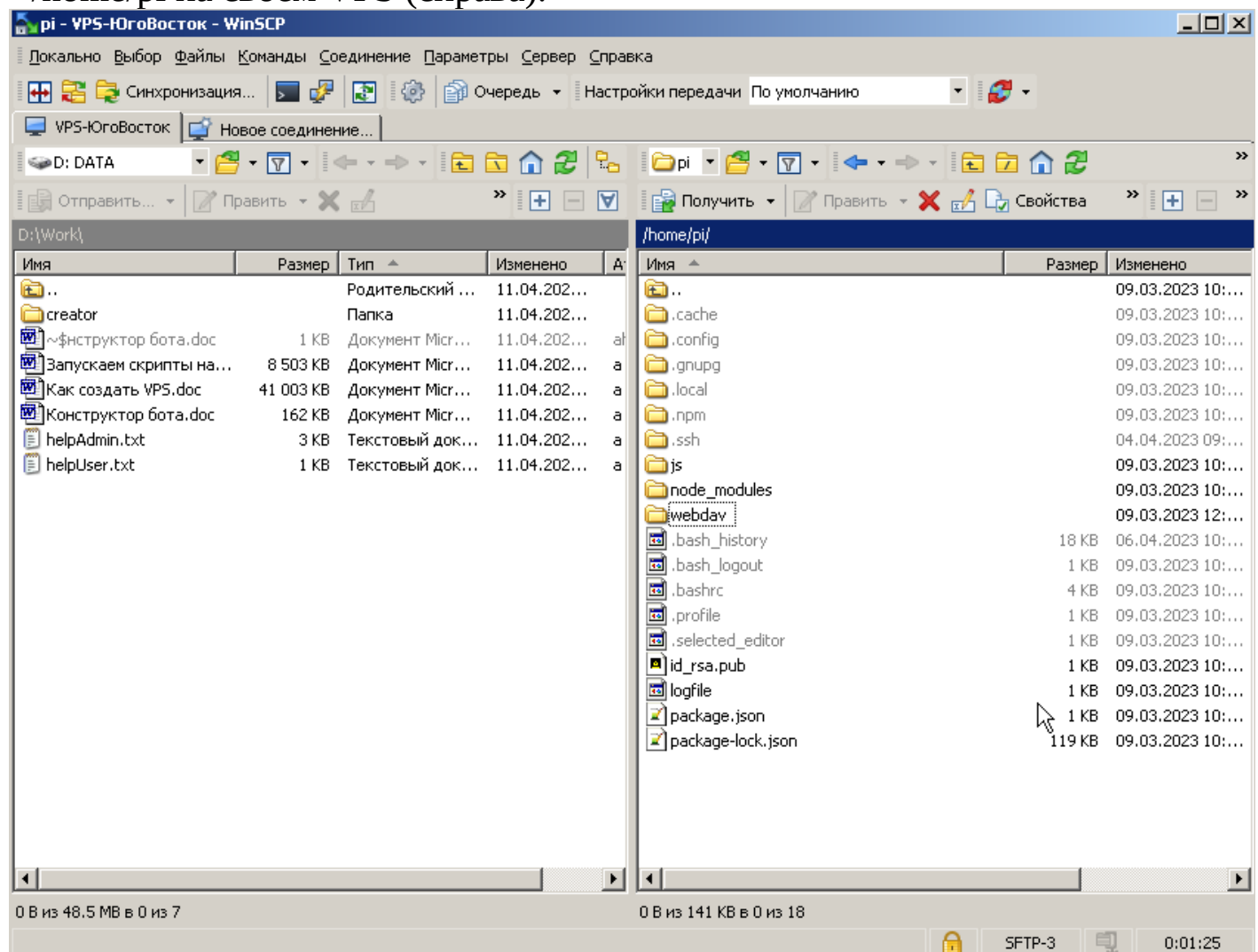
Но сначала нужно загрузить скрипт бота на сервер и корректно запустить его. Этим мы сейчас и займемся.

Итак, сервер у нас уже есть, доступ к нему мы имеем – самое время загрузить на него скрипт нашего бота. Делать это мы будем на компьютере с ОС Windows. Для начала нам нужно скачать скрипт с облачного диска по ссылке: <https://cloud.mail.ru/public/qxJx/CWJQ7b3Lf> . Там скачиваем папку – creator – в архивированном виде в удобное для нас место на диске (у меня это D:\work), извлекаем из архива в этом же месте, и удаляем *.zip файлы за ненадобностью. Имеем у себя на windows-компьютере 1 папку – creator.

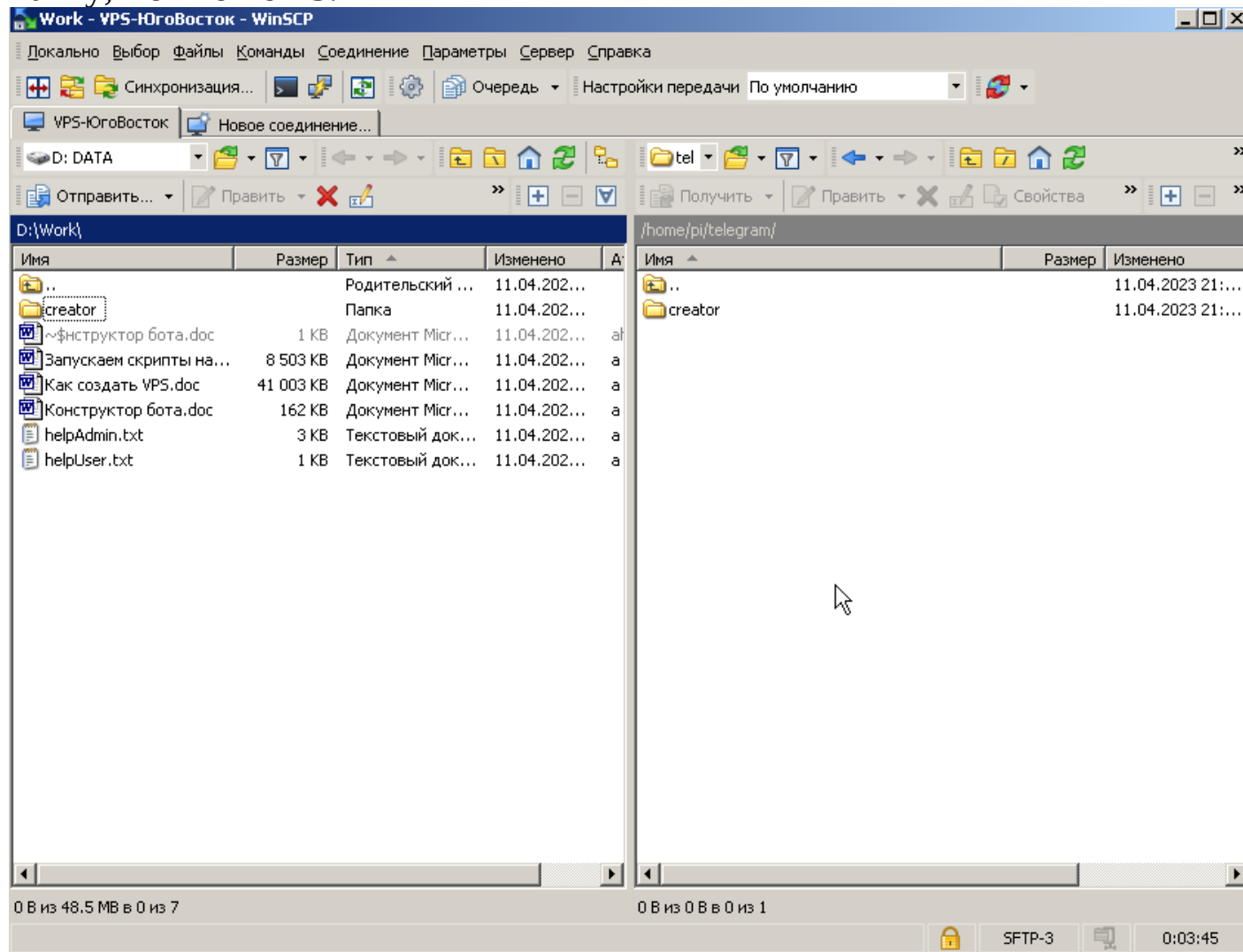
Работать с файлами на сервере VPS через терминал не очень удобно, гораздо проще это делать с помощью программы WinSCP. Скачиваем и устанавливаем ее себе на компьютер, запускаем. Создаем новую сессию, нажав на кнопку New. Заполняем поля как на скриншоте и нажимаем Save, чтобы сохранить сессию под удобным для нас псевдонимом. Чтобы не вводить потом пароль каждый раз, можно поставить галочку напротив Save password. Само собой ваш IP и пароль будут отличными от моего...☺



Все готово, нажимаем Login и попадаем в домашнюю папку пользователя pi - /home/pi на своем VPS (справа).



Правой кнопкой мыши на свободном месте справа выбираем New - Directory и создаем папку telegram. Заходим туда и копируем в это место нашу папку со скриптами из левой половины. Можно просто перетаскивать папку, можно по F5.



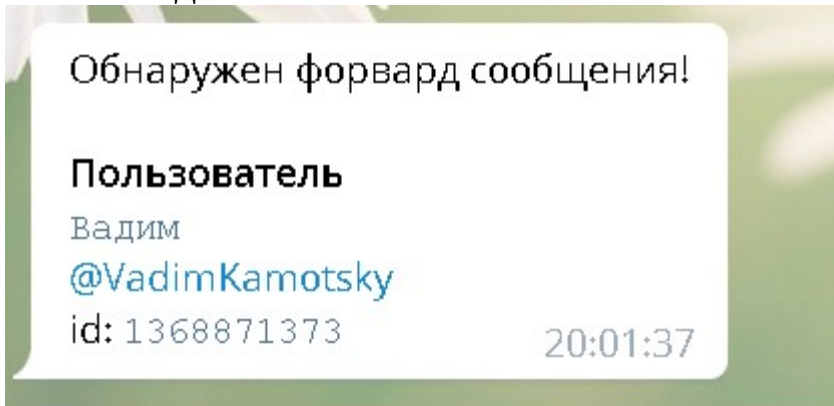
Скрипты на месте, теперь будем создавать ботов в Telegram. Сам по себе бот ничего особо делать не умеет, его можно представить просто как пользователь без номера телефона. Но он дает своему владельцу точку входа в систему телеграмм. Этим мы и воспользуемся.

Чтобы не раздувать портянку описанием, прошу создать бота самостоятельно, набрав в гугле – как создать бота @BotFather. Рекомендую только придумать красивое и запоминающееся имя на латинице. Наша цель – получить токен бота (типа входной билет). Копируем токен и переходим в winSCP в папку `/creator/Token`. Находим там файл `token_bot.json`. Двойным кликом открываем файл в редакторе и вставляем токен бота в указанное место, строго между кавычек, без пробелов и прочих левых знаков. Это важно, ибо в ином случае вы не получите доступ к своему боту из скрипта.

1

```
{"token": "сюда надо вписать токен бота"}_
```

Токен бота нужно хранить в тайне, ибо кто знает токен, тот хозяин бота. Теперь нам нужно прописать в файле chatId.json идентификатор супер-админа, человека имеющего доступ ко всем скриптам, главного админа. Самого себя, короче. ChatID в телеграмм – это типа номера телефона, но в системе телеграмм. У каждого пользователя, группы, канала этот номер имеется и он уникален. Чтобы узнать свой chatId, можно воспользоваться ботом IDBot (@username_to_id_bot). Находим бота в поиске, запускаем его и пересылаем ему любое свое сообщение из любого чата. Бот покажет вам искомые данные:

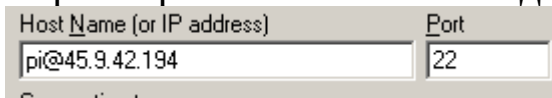


Копируем номер id из сообщения и вставляем его в запись "Supervisor" в файле chatId.json, вместо 12345..., строго между кавычек... ну вы знаете уже.

Ну что ж, вроде все готово к первому запуску нашего бота!

Если вы не устанавливали Nodejs и планируете запускать ботов в Доке-ре, то на этом этапе можно закончить и перейти к установке Докера и запуску контейнера с ботом (в файле Docker.txt).

Через терминал PuTTY заходим на наш сервер под именем pi:



и вводим команду:

```
node /home/pi/telegram/creator/Bot/creator_bot.js
```

```
pi@nasea:~$ node /home/pi/telegram/creator/Bot/creator_bot.js
```

Если мы нигде не накосячили (пока), то никаких сообщений в терминале не будет, и это значит, что бот уже работает. Идем в телеграмм, в поиске набираем имя нашего бота, стартуем его и наслаждаемся его работой! Но не увлекайтесь, не все еще настроено. Теперь нужно настроить службы сервера так, чтобы наш бот запускался автоматически в случае сбоя или перезагрузки системы.

Возвращаемся в терминал PuTTY и выключаем скрипт комбинацией Ctrl+c.

Сначала посмотрим путь к ноде:

`which node`

у меня это `/usr/bin/node`

```
pi@nasea:~$ which node
/usr/bin/node
pi@nasea:~$
```

Теперь посадим скрипт в службу, для это надо создать юнит с именем для лучшего запоминания:

`sudo nano /lib/systemd/system/creator_bot.service`

там написать:

```
[Unit]
Description=Start creator_bot.js on my Node
After=network.target
```

```
[Service]
Type=simple
User=pi
ExecStart=/usr/bin/node /home/pi/telegram/creator/Bot/creator_bot.js
Restart=on-failure
```

```
[Install]
WantedBy=multi-user.target
```

сохранить этот файл `Ctrl+o` (буква), и выйти из редактора `Ctrl+x`, перезапустить службу:

`sudo systemctl daemon-reload`

включить автозапуск при перезагрузке сервера:

`sudo systemctl enable creator_bot.service`

и запустить скрипт через службу:

`sudo systemctl restart creator_bot.service`

Идем в телеграмм и проверяем, что бот работает. Остановить бот можно командой:

`sudo systemctl stop creator_bot.service`

Если бот запустился, то самое время начать строить его структуру.

Пока бот пустой, вы будете видеть только текстовое сообщение о том, что в нем пока ничего нет. Пошлите боту команду `/help` (если вы админ 😊), и получите набор команд.

Успехов Вам!