

Немного боли Windows или «Почему нам так хорошо в Linux».

Многие из нас пользуются ssh-ключами для доступа к тем или иным ресурсам, именно потому что это наиболее безопасный способ доступа. При правильной настройке ssh-сервера в сочетании со различными средствами защиты даёт значительное ускорение в любой работе.

Конечно же для доступа к своему Git-репозиторию, при имеющейся возможности, лучше использовать именно ssh-авторизацию.

Однако, немногие из нас пользуются защитой своего приватного ssh-ключа.

Такую защиту может обеспечить KeePassXC. В одной из прошлых статей я рассказывал как можно осуществлять правильное хранение приватного ключа «id_rsa», а также автоматически подключать его к системе, но только в Linux при установленном пакете «OpenSSH-server» (или, в некоторых системах, просто «OpenSSH»).

При таком использовании доступа вам даже пароль от вашего приватного ключа вводить не придётся - ввод осуществляется автоматически. Всё что необходимо сделать - это запустить KeePassXC (при подключенной базе) и авторизоваться.

Другое дело Windows.

Чтобы также автоматически авторизовываться в «Git-репозитории» по «ssh-ключу» придётся натворить немало велосипедов, иначе здесь это всё не прокатит.

Я пробовал несколько подходов: «OpenSSH-win32», «KeePassXC», «KeePass» с плагином «KeeAgent», «Cygwin OpenSSH», «Git Windows OpenSSH», «Putty Pageant». Но всё тщетно.

Все указанные программы, даже при условии дополнительных галочек, настроек, плагинов - для своей работы требуют включенный «ssh-agent».

Данный агент от «OpenSSH-win32» не работает. НО, запуск из под «Git-Windows»: `eval `ssh-agent.exe -s`` работать будет. При запуске последнего

ни одна программа данный агент попросту не видит. При этом Диспетчер задач говорит, что агент включен и работает.

При запуске утилиты «Pageant» от «Putty» - всё прекрасно заработает.

Не удивляйтесь, **НО НИЧЕГО РАБОТАТЬ НЕ БУДЕТ!**

Да-да, именно так. «ssh-ключ» из базы конечно будет автоматически добавляться в запущенный агент «Pageant», но к сожалению формат хранения ключей у этого агента не совместим с форматом «OpenSSH». При этом каким-либо образом сохранить ключ куда-либо не представляется возможным.

Сделать перенаправления или изменения конечной папки с ключами у обоих агентов, также не представляется возможным.

Могу вас обрадовать - все программы для «Windows» для работы с базами ключей «kdbx» работают только с агентом «Pageant».

Однако, не всё так плохо, как может показаться.

Есть один более-менее нормальный «Велосипед» для работы с ключами через «OpenSSH» агент непосредственно из под «Git-Windows».

Возьмём для примера «Portable-Git-Windows» и распакуем в папку temp. Туда же сохраним из базы ключей наш приватный «id_rsa».

Запускаем Git-Bash и выполняем следующие команды:

```
$ eval `ssh-agent -s`
```

```
$ ssh-add ./id_rsa
```

Мы включили встроенный агент самого «Git»-а и добавили наш приватный ключ.

В данном случае никакие переменные вроде «SSH_ENV» не нужны и скрипты тоже.

После работы с «Git»-репозиторием необходимо удалить данный ключ из базы «ssh-агента»:

```
$ ssh-add -D
```

-D удаляет все ключи, можно и только один указанный ключ «-d».

А затем выключаем запущенный агент:

```
$ ssh-agent -k
```

Команда сначала удаляет все созданные для работы папки по пути «/tmp/» а затем выключает сам агент.

Запомните следующее. Не пытайтесь запустить ещё один агент или проверить есть ли агент командой:

```
$ ssh-agent
```

Данная команда запустит в системе ещё один агент и вы уже не сможете выключить его одной командой. Вам скорее всего уже понадобится найти все запущенные агенты в процессах.

Хотите что-то узнать, лучше запросите простую помощь:

```
$ ssh-agent --help
```

Ну а на этом всё. Если вы вынуждены иногда пользоваться Windows - это не конец света. Даже здесь можно найти выход.

Ну а с вами как всегда был Shadow.

Подписывайтесь на канал. Ставьте лайки, комментируйте.

Всем Добра и Удачи!