

5

Рафи Кази ар

НКАбд-03-24

1032238132@pfur.ru

Кулябов Дмитрий Сергеевич

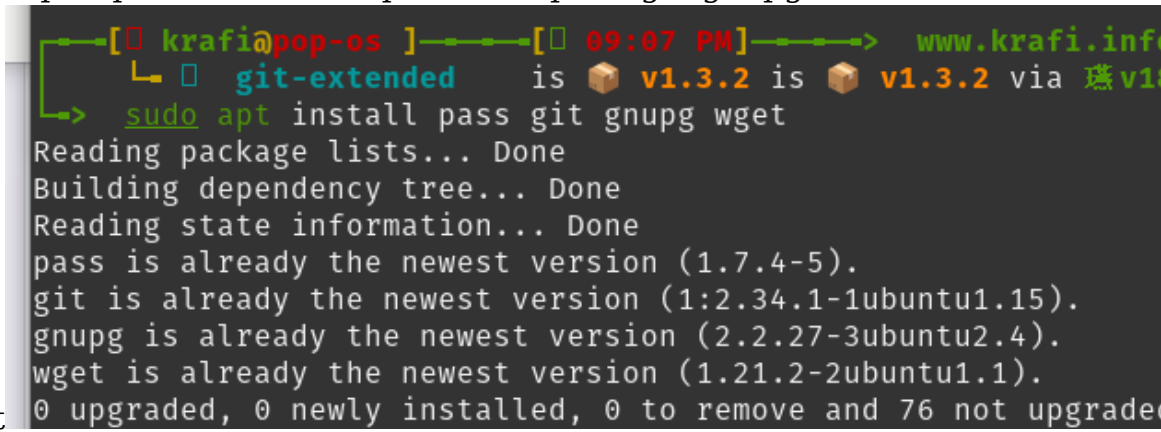
Moscow

Часть 1: Установка и настройка менеджера паролей Pass в Ubuntu

1.1. Установка В системе Ubuntu:

- **Обновление списка пакетов:** bash
- **Установка классического pass и необходимых зависимостей:** bash

`sudo apt update & sudo apt install pass git gnupg wget`



```
[krafi@pop-os] 09:07 PM www.krafi.info
└─ git-extended is v1.3.2 is v1.3.2 via 燕v1
└─ sudo apt install pass git gnupg wget
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
pass is already the newest version (1.7.4-5).
git is already the newest version (1:2.34.1-1ubuntu1.15).
gnupg is already the newest version (2.2.27-3ubuntu2.4).
wget is already the newest version (1.21.2-2ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 76 not upgraded
```

- git - для синхронизации хранилища.

- `gnupg` - для работы с GPG-ключами.
- `wget` - для загрузки скриптов.
- **Установка `gopass` (расширенная версия на Go):** `bash sudo apt install gopass` *Далее в инструкции предполагается использование `pass`, но команды для `gopass` часто идентичны.*

1.2. Настройка GPG-ключа `Pass` использует для шифрования GPG. Сначала нужен ключ.

- **Проверить существующие секретные ключи:** `bash`
`gpg --list-secret-keys`
- **Если ключей нет, создать новый (рекомендуется):** `bash`
`gpg --full-generate-key`

1. Выберите тип ключа RSA and RSA (по умолчанию).
2. Размер ключа: 4096.
3. Срок действия: 0 (ключ не истечёт).
4. Введите своё имя и email. Эти данные будут идентифицировать ключ.
5. **Внимание!** Придумайте и надёжно сохраните *парольную фразу (passphrase)* для этого ключа. Без него вы не расшифруете свои

```

[krafi@pop-os ] [ 09:07 PM] > www.krafi.info
└─ git-extended is v1.3.2 is v1.3.2 via v18.20
└─ gpg --full-generate-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 Free Software Foundation,
This is free software: you are free to change and redistribute it
There is NO WARRANTY, to the extent permitted by law.

Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (14) Existing key from card
Your selection?
RSA keys may be between 1024 and 4096 bits long.
What keysizes do you want? (3072)
Requested keysizes is 3072 bits
Please specify how long the key should be valid.
    0 = key does not expire
    <n> = key expires in n days
    <n>w = key expires in n weeks
    <n>m = key expires in n months
    <n>y = key expires in n years
Key is valid for? (0)
Key does not expire at all
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: krafi
Email address: krafi.info@gmail.com
Comment: krafi
You selected this USER-ID:
    "krafi (krafi) <krafi.info@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? 0
We need to generate a lot of random bytes. It is a good idea to p
some other action (type on the keyboard, move the mouse, utilize
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
We need to generate a lot of random bytes. It is a good idea to p
some other action (type on the keyboard, move the mouse, utilize
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key 8F36E89D0C050564 marked as ultimately trusted

```

пароли.

1.3. Инициализация хранилища паролей

- Проинициализируйте хранилище, указав email или ID вашего GPG-ключа: `bash pass init your-email@example.com` Хранилище создастся в папке `~/.password-store/`.

1.4. Синхронизация с Git

- **Настройка Git (если ещё не сделано):** `bash`
`git config --global user.name "krafi"`
`git config --global user.email "krafi.info@gmail.com"`
- **Инициализировать Git-репозиторий внутри хранилища:** `bash`
`pass git init`
- **Добавить удалённый репозиторий (предварительно создайте приватный репозиторий на GitHub/GitLab):** `bash`
`pass git remote add origin git@github.com:<ВАШ_USERNAME>/<ВАШ_RE`
Для использования HTTPS вместо SSH: `bash`
`pass git remote add origin https://github.com/<ВАШ_USERNAME>/<ВАШ_Р`
- **Для синхронизации изменений:** `bash`
`pass git pull` # Получить изменения с удалённого репозитория
`pass git push` # Отправить локальные изменения на

```

krafi@pop-os ] 09:17 PM www.k
└─ git-extended is v1.3.2 is v1.3.2
└─
ls -la ~/.password-store/

pass git init
total 4
drwx----- 1 krafi krafi 14 Aug 25 18:17 .
drwxr-x--- 1 krafi krafi 1024 Aug 25 18:17 ..
-rw----- 1 krafi krafi 23 Aug 25 18:17 .gpg-id
Initialized empty Git repository in /home/krafi/.password-store/
[master (root-commit) c2424df] Add current contents of password-store
1 file changed, 1 insertion(+)
create mode 100644 .gpg-id
[master ec7678c] Configure git repository for gpg files
1 file changed, 1 insertion(+)
create mode 100644 .gitattributes

krafi@pop-os ] 09:17 PM www.k
└─ git-extended is v1.3.2 is v1.3.2
└─
git config --global user.name "krafi"
git config --global user.email "krafi.info@gmail.com"

krafi@pop-os ] 09:18 PM www.k
└─ git-extended is v1.3.2 is v1.3.2
└─
pass git init
Reinitialized existing Git repository in /home/krafi/.password-store/

krafi@pop-os ] 09:18 PM www.k
└─ git-extended is v1.3.2 is v1.3.2
└─
git push
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Everything up-to-date

```

удалённый репозиторий

- **Важное замечание:** pass отслеживает только те изменения, которые сделаны через его команды. Если вы редактируете файлы напрямую в ~/.password-store/, вам нужно вручную сделать коммит и пуш: bash

cd ~/.password-store/

git add .

git commit -m 'Описание правок'

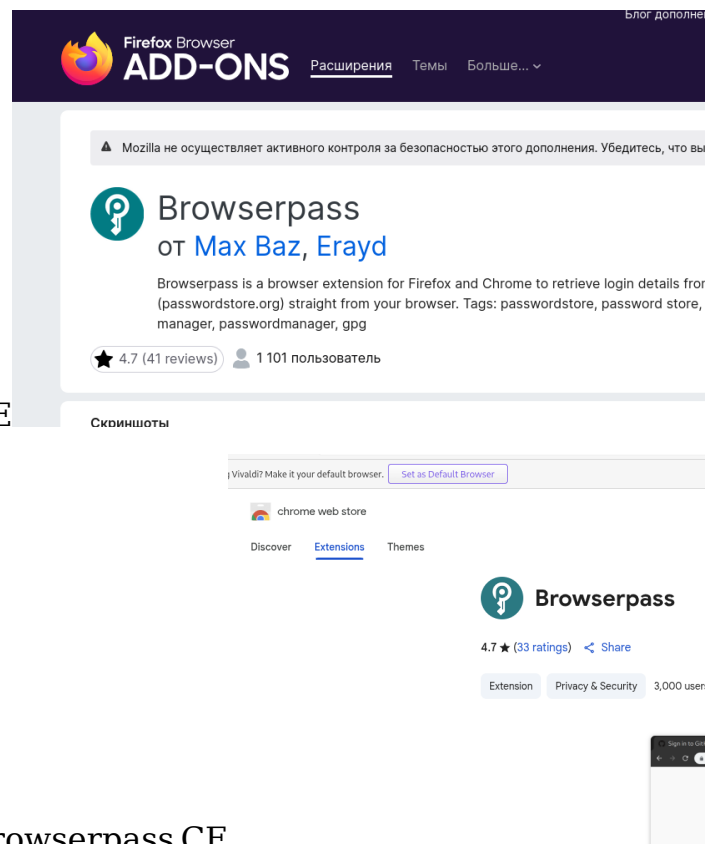
git push origin main

- **Проверить статус репозитория:** `bash`
`pass git status`

1.5. Интеграция с браузером (Browserpass) Для удобного автозаполнения паролей в браузере.

- **Установите расширение для вашего браузера:**

- **Firefox:** Browserpass CE



- **Chrome/Chromium:** Browserpass CE

После установки необходимо настроить браузер для работы с хостом. Часто это делается командой: `bash`

```
sudo mkdir -p /etc/opt/chrome/native-messaging-
hosts/ # Для Chrome sudo mkdir -p /etc/chromium/native-
messaging-hosts/ # Для Chromium sudo mkdir -p
/usr/lib/mozilla/native-messaging-hosts/ # Для Firefox
sudo ./configure
```

Запустить из каталога установки browserpass
*Актуальные инструкции всегда смотрите на
GitHub Browserpass.*

1.6. Базовое использование Pass

- **Добавить новый пароль (программа запросит его ввод):** bash
pass insert Соцсети/instagram.com
- **Показать пароль в терминале:** bash
pass Соцсети/instagram.com
- **Скопировать пароль в буфер обмена (без отображения на экране):** bash pass
-с Соцсети/instagram.com
- **Сгенерировать новый сложный пароль и сохранить его (заменит старый):** bash pass
generate -n Соцсети/instagram.com 20 # 20 -
длина пароля

Часть 2: Установка и настройка Chezmoi для управления конфигурацией в Ubuntu

2.1. Установка дополнительного ПО (опционально)

Установка некоторых полезных утилит, часто встречающихся в конфигах:

sudo apt install curl git ripgrep fzf dunst # Основные утилиты

```

[krafi@pop-os ] 09:22 PM www.krafi.info
└─ .password-store
└─ sudo apt install curl git ripgrep fzf dunst
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
dunst is already the newest version (1.5.0-1build1).
curl is already the newest version (7.81.0-1ubuntu1.20).
git is already the newest version (1:2.34.1-1ubuntu1.15).
The following NEW packages will be installed:
  fzf ripgrep
0 upgraded, 2 newly installed, 0 to remove and 76 not upgraded.
Need to get 2,252 kB of archives.
After this operation, 6,742 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://apt.pop-os.org/ubuntu jammy-security/universe amd64 fzf amd64 0.29.0-1ubuntu0.1 [2,252 kB]
Get:2 http://apt.pop-os.org/ubuntu jammy-security/universe amd64 ripgrep amd64 13.0.0-2ubuntu0.1 [1,000 kB]
Fetched 2,252 kB in 0s (5,487 kB/s)
Selecting previously unselected package fzf.
(Reading database ... 316402 files and directories currently installed.)
Preparing to unpack .../fzf_0.29.0-1ubuntu0.1_amd64.deb ...
Unpacking fzf (0.29.0-1ubuntu0.1) ...
Selecting previously unselected package ripgrep.
Preparing to unpack .../ripgrep_13.0.0-2ubuntu0.1_amd64.deb ...
Unpacking ripgrep (13.0.0-2ubuntu0.1) ...
Setting up fzf (0.29.0-1ubuntu0.1) ...
Setting up ripgrep (13.0.0-2ubuntu0.1) ...
Processing triggers for man-db (2.10.2-1) ...
[krafi@pop-os ] 09:22 PM www.krafi.info
└─ .password-store took 3s
└─

```

и *dunst* (уведомления)

2.2. Установка шрифтов (рекомендуется) Для правильного отображения значков и оформления в некоторых конфигах (например, для статус-баров). * **Скачать и установить шрифты Nerd Fonts (например, FiraCode) вручную:** 1. Скачайте архив с www.nerdfonts.com

2. Распакуйте и скопируйте .ttf файлы в ~/.local/share/fonts/

3. Обновите кэш шрифтов:

fc-cache -fv

2.3. Установка Chezmoi Установите последнюю версию с официального сайта (рекомендуемый способ):

```

[krafi@pop-os ] 09:22 PM
└─ .password-store took 3s
└─ sh -c "$(wget -qO- chezmoi.io/get)"
info found chezmoi version 2.64.0 for lat
info found glibc version 2.35
info installed bin/chezmoi

```

sh -c "\$(wget -qO- chezmoi.io/get)"

- *Альтернатива: установка из бинарного пакета (актуальную версию уточняйте на GitHub):* `bash # Пример для версии 2.47.1 и архитектуры AMD64`
`wget https://github.com/twpayne/chezmoi/releases`
`sudo dpkg -i chezmoi_2.47.1_linux_amd64.deb`

2.4. Создание своего репозитория dotfiles Используйте GitHub CLI (gh), чтобы создать **приватный** репозиторий.
*** Сначала установите gh:** `bash sudo apt install gh *`
Авторизуйтесь: `bash`

`gh auth login *` **Создайте репозиторий:** `bash`

`gh repo create dotfiles --private --clone` Если хотите использовать шаблон, добавьте флаг `--template="yamadharma/dotfiles-template"`, но убедитесь, что шаблон совместим с Ubuntu.

2.5. Первоначальная настройка Chezmoi на первой машине

1. **Инициализация:** Перейдите в клонированный репозиторий и инициализируйте Chezmoi. `bash`
`cd ~/dotfiles chezmoi init`
2. **Просмотр планируемых изменений (ВСЕГДА делайте это сначала!):** `bash chezmoi diff`
Эта команда покажет, какие файлы будут добавлены/изменены в вашем домашнем каталоге.
3. **Применение изменений:** Если вас всё устраивает, примените конфигурацию. `bash`
`chezmoi apply -v # Ключ -v для подробного вывода`
4. **Добавление существующих конфигов:** Чтобы добавить текущий файл конфигурации (например, `.bashrc`) под управление Chezmoi:
`bash chezmoi add ~/.bashrc`
5. **Сохранение изменений в ваш репозиторий:**
`bash chezmoi cd # Перейти в каталог хранилища`

```
Chezmoi git add . git commit -m "Добавил .bashrc"  
git push origin main
```

2.6. Настройка второй машины (или чистой системы)

1. **Инициализация и немедленное применение (рекомендуемый способ):** `bash chezmoi init --apply https://github.com/<BAIII_USERNAME>/dotfiles.git`
2. **Или стандартный способ:** `bash chezmoi init https://github.com/<BAIII_USERNAME>/dotfiles.git`
`chezmoi diff # Посмотреть, что поменяется chezmoi apply -v # Применить изменения`

2.7. Ежедневное использование Chezmoi

- **Обновление и применение изменений из репозитория:** `bash`
`chezmoi update # Эквивалент:`
`git pull + chezmoi apply`
- **Редактирование управляемого файла (правильный способ):** `bash`
`chezmoi edit ~/.bashrc`
`# Откроет файл в $EDITOR После сохранения Chezmoi автоматически применит изменения локально.`
- **Просмотр изменений перед применением:** `bash`
`chezmoi diff`
- **Если возник конфликт (файл изменён и локально, и в репозитории):** `bash chezmoi merge ~/.bashrc # Запустит инструмент для разрешения конфликтов`

2.8. Автоматизация (опционально, с осторожностью)

Чтобы Chezmoi автоматически коммитил и пушил изменения после команд `add` или `edit`, добавьте в `~/.config/chezmoi/chezmoi.toml`:

[git]

autoCommit = true

autoPush = true