

Стр. 1 из 25 29.07.2020, 21:10



Стр. 2 из 25 29.07.2020, 21:10

BIOS/UEFI

При загрузке системы нажимаем F2 или Del (зависит от производителя материнской платы). Во вкладке BIOS в приоритете загрузки делаем первым наше USB-устройство. Нажимаем F10 и сохраняем настройки.

Настройка сети

При подключении от кабеля ничего настраивать не надо. Настройка wifi требует ввода пары команд:

```
rfkill unblock wifi
```

Следует отметить, что не все usb wifi адаптеры гараниторованно поддерживаются. Например, у меня не захотел работать dexp wfa 301, а вот с tp-link все ок.

Разметка диска

Для начала посмотрим все доступные устройства:

fdisk -1

Создадим новый раздел:

fdisk /dev/nvme0n1

В меню fdisk вводим $_{\rm B}$ для создания нового раздела, порядковый номер раздела, потом начальное и конечные смещения. При задании конечного смещения можно отрицательное значение, например, $_{\rm 10G}$, так мы оставим свободными 10 Гб в конце диска. Для записи изменений на жесткий диск вводим $_{\rm W}$ и выходим - $_{\rm Q}$.

Теперь нужно разметить раздел по LVM. Касательно LVM: единственное удобство в нем для меня – это возможность динамически менять размер разделов. Самая частая проблема, которая у меня была раньше – это то, что я не угадывал размер для корня. Когда в нем кончалось место, я с установочной флешки через gparted пытался отщепнуть от виндового раздела кусок (а у меня разделы всегда шли так: recovery, efi, reserved, windows, root, home), из этого куска создавал временный раздел, копировал в него файлы с хомяка, удалял хомяка, изменял размер корневого, опять создавал хомяка и из временного раздела копировал в него файлы, потом удалял временный раздел, расширял виндовый, правил / etc/fstab ... A c LVM я могу просто выполнить пару команд: ужать хомяка, расширить корневой, причем, на лету. Но эксперты могут возразить: «Хомяк на отдельном разделе не нужен!» — Да, если хочешь потерять все данные пои переустановке.

Создадим группу:

vgcreate lvm /dev/nvme0n1p

Теперь создадим в ней логические разделы

```
lvcreate -L 30G arch -n root
lvcreate -L 20G arch -n home
mkfs.ext4 /dev/lvm/root
mkfs.ext4 /dev/lvm/home
```

Устанавливаем ядро

```
mount /dev/lvm/root /mnt
mkdir /mnt/home
mount /dev/lvm/home /mnt/home
mount /dev/lvm/home /mnt/home
mkdir -p /mnt/boot/efi
# Coздаем файл полкачки
fallocate -1 26 /mnt/swapfile
# Если хотим использовать гибериацию
# fallocate -1 'awk '/Mem:/ {print $2}' <(free -m)'M /mnt/swapfile
chmod 600 /mnt/swapfile
mkswap /mnt/swapfile
mkswap /mnt/swapfile
# Устанавливаем ядро системы
pacetrap /mnt base base-devel
```

Генерируем fstab

genfstab /mnt >> /mnt/etc/fstab

arch-chroot

```
# Предотвращаем ошибки lvm:
# WARNING: Failed to connect lvmetad...
# WARNING: Device /dev/nvme0n1 not initialized in udev database...
mkdir /mmt/hostlvm
mount --bind /run/lvm /mnt/hostlvm
arch-chroot /mnt
l --s /run/lvm /hostlvm
```

Настраиваем дату и локаль

```
ln -sf /usr/share/zoneinfo/Europe/Moscow /etc/localtime
hwclock --systohc
```

Далее:

nano /etc/locale.gen

Раскоментируем

en_US.UTF-8

Стр. 3 из 25 29.07.2020, 21:10

```
Генерируем локаль:
  locale-gen
echo "LANG-en_US.UTF-8" > /etc/locale.conf
Если пропустить этот шаг, то не будет запускаться терминал.
Прописываем хосты
  echo "sergey-pc" > /etc/hostname
 nano /etc/hosts
Добавляем в файл такие строки:
  127.0.0.1 localhost
  ::1 localhost
127.0.1.1 sergey-pc.localdomain sergey-pc
Initramfs
Если пропустить этот шаг, то система не станет грузиться с lvm.
Нам нужно отредактировать /etc/mkinitcpio.conf и модифицировать список HOOKS, добавив 1vm2 ДО ЗНАЧЕНИЯ
 HOOKS=(base udev autodetect modconf block lvm2 filesystems keyboard fsck)
Генерируем:
Ставим пакеты
Эти пакеты поналобятся далее:
  pacman -S sudo grub efibootmgr ntfs-3g os-prober alsa-utils xf86-video-ati gn
xf86-video-ati – свободный драйвер для видеокарт AMD. xorg и xorg-server отдельно ставить не нужно, так как эти пакеты
есть в зависимостях.
Пользователи
Задаем пароль для супер-пользователя:
Создаем пользователя:
  useradd -m -g users -G wheel -s /bin/bash sergey
Устанавливаем пароль для нового пользователя:
Пароль для пользователя можно ставить 1, так как при логине придется его сменит.
Теперь в файле /etc/sudoers нужно раскоментировать строку:
  %wheel ALL=(ALL:ALL) ALL
Установка grub
  grub-install --target=x86_64-efi --efi-directory=/boot/efi --bootloader-id="Arch Linux"
Чтобы отключить автоматическую загрузку Linux, редактируем дефолтный конфиг груба
  nano /etc/default/grub
Меняем GRUB_TIMEOUT:
 GRUB_TIMEOUT=-1
Затем генерируем grub:
  grub-mkconfig -o /boot/grub/grub.cfg
Завершение
Включаем gdm для экрана логина и MetworkManager для автоматического подключения к сети:
  systemctl enable gdm
systemctl enable NetworkManager
Надо еще настроить звук
  # Сохраним на всякий случай состояние
```

Стр. 4 из 25 29.07.2020, 21:10

```
alsactl store
# Для автоматического сохранения/восстановления значения громкости звука:
systemctl enable alsa-restore

Теперь можно выходить и перегружаться:
```

Пакетные менеджеры

Пакетным менеджером по-умолчанию для Arch Linux является растал. Для подсветки вывода растал в /etc/pacman.conf нужно раскомментировать color. Пользовательским репозиторием является АUR. Пакеты из него можно использовать только на свой страх и риск. Чтобы не собирать пакеты из него ручками можно поставить уау:

```
sudo pacman -S git
cd /tmp
git clone https://aur.archlinux.org/yay.git
cd yay
makepkg -si
```

Пакеты всегда нужно ставить из репозиториев. Смысла собирать их из исходников нет, так как в AUR и так самые последние версии. Так будет меньше хлама оставаться в системе после их удаления.

В AUR много пакетов, оканчивающихся на —git. При их установке исходники копируются с github, а затем собираются, что занимает много времени и требуется дополнительное дисковое пространство для создания временных файлов и т.п., а еще для сборки могут потребоваться дополнительные пакеты. Поэтому лучше избегать установки таких пакетов, если есть альтернативы.

Синтаксис команды Yay аналогичен растап. Так что будет полезным почитать справку по растап.

Ссылки

• Сравнеие пакетных менеджеров

Нужные пакеты

Это список необходимых для меня пакетов:

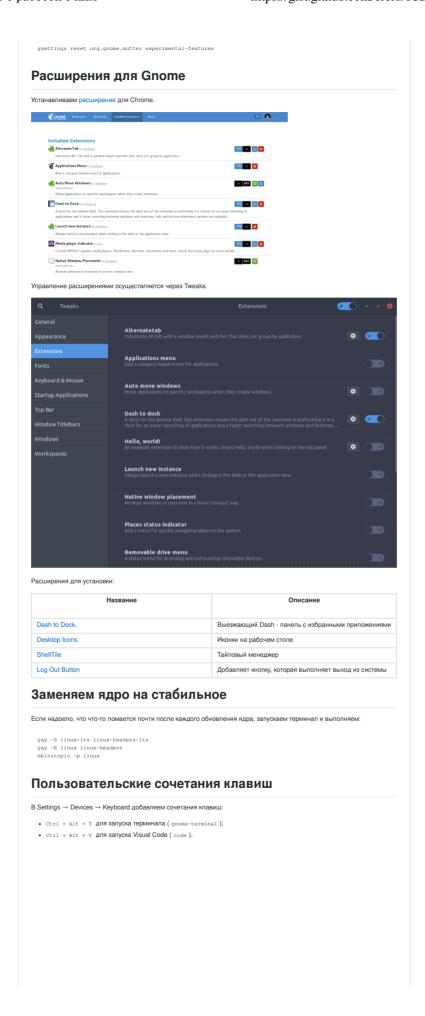
Масштабировавние 150% как в Windows

По-умолчанию в Gnome масштабирование кратно 100. Чтобы добавить варианты масштабирования 125% и 150% нужно выполнить в терминале:

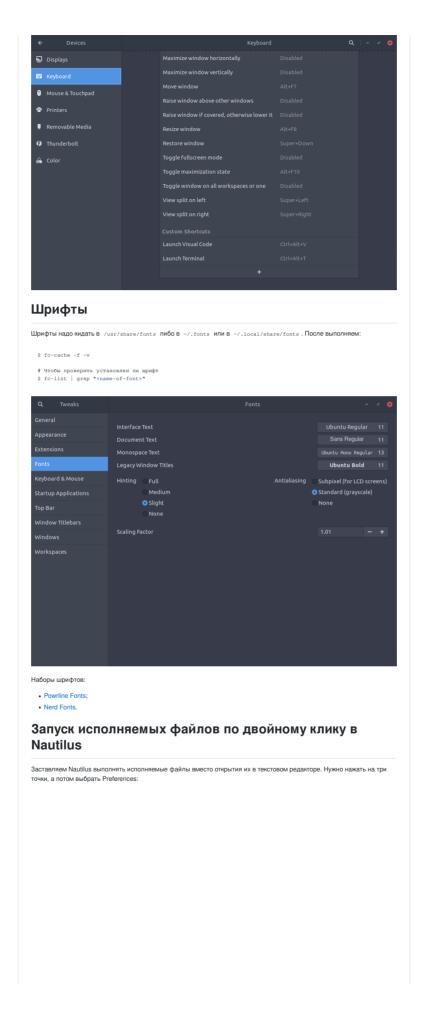
```
gsettings set org.gnome.mutter experimental-features "['scale-monitor-framebuffer']"
```

Отключение

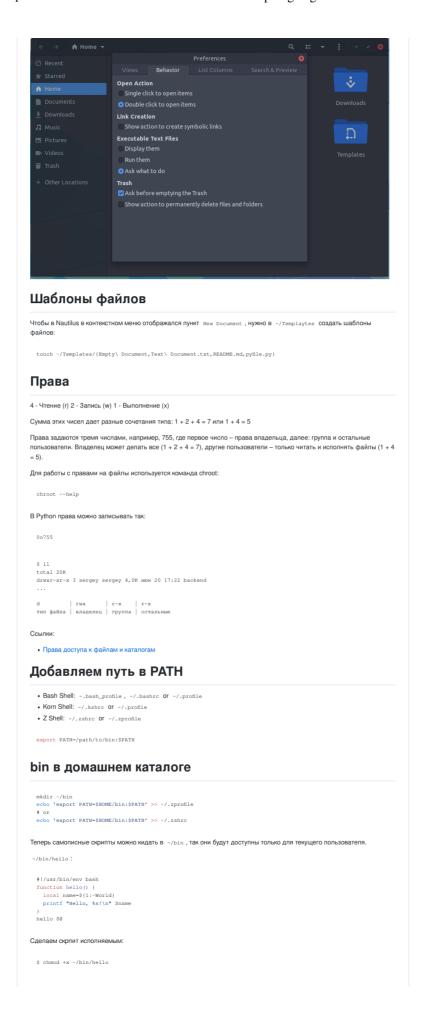
Стр. 5 из 25 29.07.2020, 21:10



Стр. 6 из 25 29.07.2020, 21:10



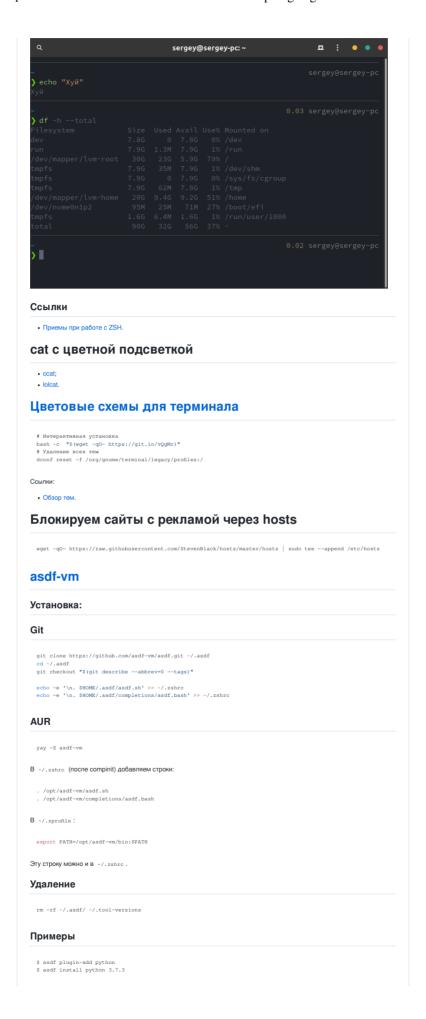
Стр. 7 из 25



Стр. 8 из 25 29.07.2020, 21:10

```
Проверка:
   # Если не перелогини
$ source ~/.zprofile
$ hello $USER
                         инивались после добавления пути в ~/.zprofile, то сначала выполняем
  Hello, sergey
ZSH
Установка
  $ yay -S zsh
Меняем shell на /bin/zsh:
  $ chsh -s $ (which zsh)
Чтобы изменения вступили в силу нужно залогиниться по-новой.
Oh My Zsh
Установка:
  sh -c "$(curl -fsSL https://raw.githubusercontent.com/robbyrussell/oh-my-zsh/master/tools/install.sh)"
Пакет из репозитория ставится вне домашнего каталога, а потому требует root права при установке плагинов, что не
удобно.
Так же для некоторых тем Oh My Zsh нужны шрифты наподобие Powerline:
  yay -S powerline-fonts
Ставим must-have плагины:
  git clone https://github.com/zsh-users/zsh-autosuggestions ${ZSH_CUSTOM:--/.oh-my-zsh/custom}/plugins/zsh-auto git clone https://github.com/zsh-users/zsh-completions ${ZSH_CUSTOM:--/.oh-my-zsh/custom}/plugins/zsh-completi git clone https://github.com/zsh-users/zsh-syntax-highlighting.git ${ZSH_CUSTOM:--/.oh-my-zsh/custom}/plugins/
Изменяем .zshrc:
  ZSH_THEME="agnoster"
  plugins=(
    command-not-found extract
 extract
git
zsh-autosuggestions
zsh-completions
zsh-syntax-highlighting
)
  autoload -Uz compinit && compinit
  source $ZSH/oh-my-zsh.sh
Для темы Agnoster настройках терминала выбираем шрифт Source Code Pro Regular , чтобы отображались стрелочки
  • Встроенные темы:
  • Сторонние темы
Темы
Jovial
  # Сначала сохраняем копию zhsrc, потому как jovial изменить оригинальеный файл
$ cp -/.zshrc -/.zshrc.bak
$ curl -sSL git.io/jovial | sudo bash -s $USER
У меня эта тема упорно устанавливается в /root/.oh-my-zsh
Powerlevel10k
Это красивая тема для ZSH.
  git clone https://github.com/romkatv/powerlevel10k.git $ZSH_CUSTOM/themes/powerlevel10k
 ~/.zshrc:
  ZSH_THEME=powerlevel10k/powerlevel10k
  cd && curl -fsSLO https://raw.githubusercontent.com/romkatv/dotfiles-public/master/.purepower echo 'source ~/.purepower' >>! ~/.zshrc
```

Стр. 9 из 25 29.07.2020, 21:10



Стр. 10 из 25 29.07.2020, 21:10

```
$ asdf install python 2.7.15
$ asdf list python
     2.7.15
   $ asdf uninstall python 2.7.15
  S asdf uninstall python 2.7.15
$ asdf global python 3.7.3
# Сделать системную версию Python глобальной
S asdf global python system
S which python
/home/sergey/.asdf/shims/python
  $ asdf plugin-add nodejs
# see: \https://github.com/asdf-vm/asdf-nodejs#install>
$ bash -/.asdf/plugins/nodejs/bin/import-release-team-keyring
$ asdf install nodejs 10.16.0
$ asdf global nodejs 10.16.0
   $ asdf global nodejs 10.16.0
$ which node
/home/sergey/.asdf/shims/node
$ which npm
/home/sergey/.asdf/shims/npm
$ asdf list
   golang
1.12
   10.16.0
postgres
  postgres
11.4
python
2.7.15
3.7.3
   sqlite
Все доступные плагины. При установке, использовании плагинов могут возникать проблемы. Например, плагин для
Python работает поверх руепv и при возникновении проблем, следует изучить страницу «Common build problem
  If you use pip to install a module like ipython that has a binaries. You will need to run asdf reshim python for the binary to be
  in your path.
После установки через рір пакетов, которые добавляют команды, чтобы те были доступны, нужно всегда выполнять
 asdf reshim nutho
Ссылки
   • Локументация
Использование NVM
▲ Использование asdf-vm предпочительнее
Устанавливаем последнюю версию Node.js:
   nvm install node
Настройка Docker
  sudo systemctl start docker
sudo systemctl enable docker
# sudo groupadd docker
# groupadd: group 'docker' already exists
sudo usermod -aG docker $USER
Нужно выйти и войти в систему, а потом проверить:
   docker run hello-world
Hастройка Visual Code
```

```
"editor.fontSize": 16,
"editor.rulers": {
    72,
    80,
    100,
    120
],
"editor.tabSize": 2,
"editor.wordWrap": "bounded",
"editor.wordWrapColumn": 120,
"files.insertFinalNewline": true,
"files.trimTraillingWhitespace": true,
"files.trimTraillingWhitespace": true,
"terminal.integrated.fontFamily": "Source Code Pro"
}
```

Гибернация

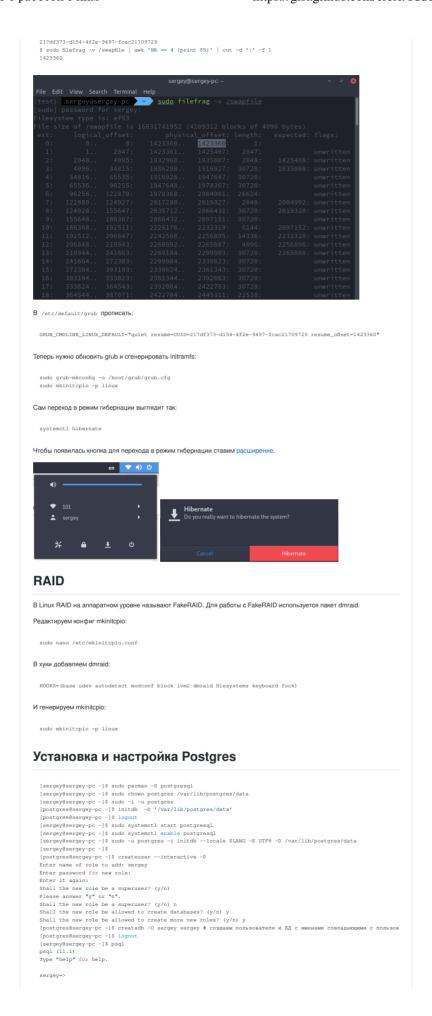
Режим гибернациии от режима сна отличается тем, что в первом случае содержимое оперативной памяти сохраняется на жесткий диск и питание полностью отключается, во втором - питание подается только на оперативку. Чем хороша гибернация? - Например, мы работаем в Linux, вошли в режим гибернации, а затем загрузились в Windows и играем. Когда мы в следующий раз загрузимся в Linux, то увидим все то, что было перед выключением. Прекрасно?! Но часто ли такое нужно?

При переходе в режим гибернации делается дамп памяти на диск, причем всей, а не только используемой, так что размер файла подкачки должен быть не меньше количества оперативки. Про гибернацию лучше почитать здесь.

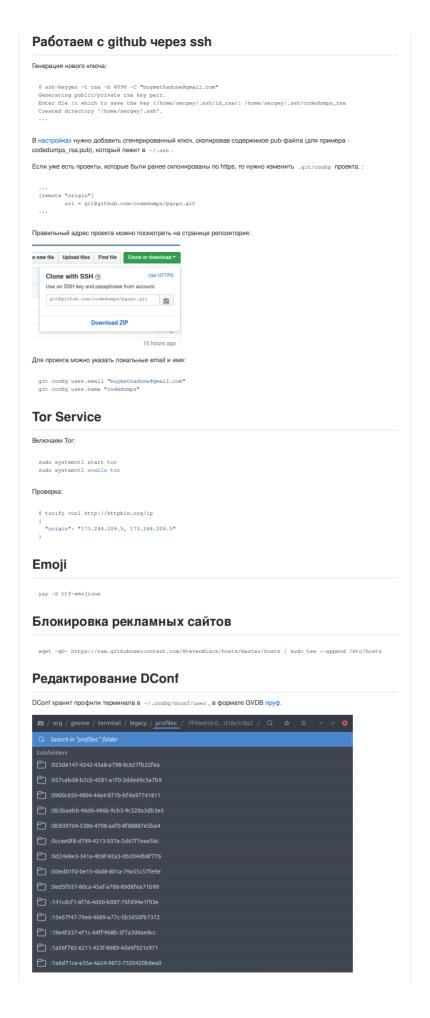
Режим гибернации по-умолчанию отключен. Чтобы его включить для начала нужно узнать UUID раздела, где расположен своп, а так же смещение своп-файла относительно начала раздела:

```
\ lsblk 'df /swapfile | awk '/^\/dev/ {print $1}'` -no UUID
```

Стр. 11 из 25 29.07.2020, 21:10



Стр. 12 из 25 29.07.2020, 21:10



Стр. 13 из 25 29.07.2020, 21:10

```
# Делаем дамп

$ dconf dump / > /tmp/dconf

# Редактируем и загружаем

$ dconf load / < /tmp/dconf
i3
 Описание
 i3 – это тайловый оконный менеджер для Linux. Тут настройки под меня.
Установка и настройка
    $ yay -5 awesome-terminal-fonts bumblebee-status compton fonts-powerline dmenu i3-gaps i3lock-fancy-git lxappe
$ sudo nano /usr/share/xsessions/13-custom.desktop
[Desktop Entry]
    Name=i3 custor
    Exec=/usr/local/bin/i3-custom
    Type=Application
$ sudo nano /usr/local/bin/i3-custom
#!/bin/bash
    #!/Din/Daan
Mcdir -p -/.config/i3/logs
export TERMINAL-termite
exec 13 -V >> -/.config/i3/logs/%(date +'%F-%T').log 2>61
% sudo chamod +x /usr/local/bin/i3-custom
    $ i3-config-wizard
    $ cp /etc/xdg/termite/config ~/.config/termite/config
    $ nano ~/.config/termite/config
[options]
     font pango:Inconsolata, Font Awesome 10
   # ...
# 20% background transparency (requires a compositor)
background = rgba(63, 63, 6.8)
$ cp /etc/xdg/compton.conf ~/.config
$ nano ~/.config/13/config
    font pango:Droid Sans 10
    # ... # Заменяем все Mod1 на m и создаем переменную выше вызовов bindsym set m Mod1
    # lockscreer
    bindsym Ctrl+Sm+l exec i3lock
    # Pulse Audio Controls
bindsym XF86AudioRaiseVolume exec --no-startup-id pactl set-sink-volume 0 +5% #increase sound volu
bindsym XF86AudioLowerVolume exec --no-startup-id pactl set-sink-volume 0 -5% #decrease sound volu
bindsym XF86AudioMute exec --no-startup-id pactl set-sink-mute 0 toggle # mute sound
    bindsym XF86MonBrightnessUp exec xbacklight -inc 20 # increase screen brightness
bindsym XF86MonBrightnessDown exec xbacklight -dec 20 # decrease screen brightness
    bindsym XF86TouchpadToggle exec /some/path/toggletouchpad.sh # toggle touchpad
   # Media player controls
bindsym XF86AudioFlay exec playerctl play
bindsym XF86AudioFlause exec playerctl pause
bindsym XF86AudioFlause exec playerctl next
bindsym XF86AudioFlause playerctl next
bindsym XF86AudioFlause playerctl previous
    # захватывает весь экран и копирует в буфер обмена
bindsym --release Print exec "scrot /tmp/8F_5T_5wx$h.png -e 'xclip -selection c -t image/png < $f &6 rm $f'"
# захватывает область экрана и копирует в буфер обмена
bindsym --release Shift+Print exec "Scrot -s /tmp/8F_5T_5wx$h.png -e 'xclip -selection c -t image/png < $f &6
   bar {
    set $disk_format "(path): {used}/{size}"
    status_command bumblebee-status -m nic disk:root disk:home cpu memory sensors pulseaudio datetime layout pac
    gaps outer -10
gaps inner 20
    floating_minimum_size 75 x 50
floating_maximum_size -1 x -1
# Убрать рамки у окон:
# 1)
   # new_window pixel 0
    # for_window [class="^.*"] border none
   # for_window [class="0.*"] border none
# force floating for all new windows
# for_window [class="1.*"] floating enable
for_window [class="Nautilus" instance="file_progress"] floating enable
for_window [class="Telegram"] floating enable, resize set 800 600
# Bcmmmanzume oxem opaysepa
for_window [window_role*"pop-up"] floating enable
# no_focus [window_role*"pop-up"]
# npopaywoors repumenas
# propaywoors repumenas
exec --no-startup-id compton --config -/.config/compton.conf
# cween aproxymany
    # смена расскладки
           c --no-startup-id setxkbmap -model pc105 -layout us,ru -option grp:ctrl_shift_toggle
               становление заставки рабочего стол
Нужно выйти из сессии и выбрать в Display Manager сессию 13 custom
Nitrogen позволяет менять обои.
Для изменения оформления i3 - служит i3-style:
```

Стр. 14 из 25 29.07.2020, 21:10

```
$ yay -S i3-style
$ i3-style archlinux -o ~/.config/i3/config --reload
```

XTerm

BMecto ctrl+shift+V нужно использовать shift+Ins, а вместо ctrl+shift+C - ctrl+c.Права кнопка мыши

Termite: горячие клавиши

Сочетание	Значение
ctrl-shift-x	activate url hints mode
ctrl-shift-r	reload configuration file
ctrl-shift-c	copy to CLIPBOARD
ctrl-shift-v	paste from CLIPBOARD
ctrl-shift-u	unicode input (standard GTK binding)
ctrl-shift-e	emoji (standard GTK binding)
ctrl-tab	start scrollback completion
ctrl-shift-space	start selection mode
ctrl-shift-t	open terminal in the current directory [1]_
ctrl-shift-up	scroll up a line
ctrl-shift-down	scroll down a line
shift-pageup	scroll up a page
shift-pagedown	scroll down a page
ctrl-shift-1	reset and clear
ctrl-+	increase font size
ctrl	decrease font size
ctrl	reset font size to default

Отсюла

Цветовые схемы Termite

```
$ curl https://raw.githubusercontent.com/khamer/basel6-termite/master/themes/basel6-nord.config >> ~/.config/ter
$ nano ~/.config/termite/config
# 4-oe sнaчение отвечает за проэрачиость (1 - непрозрачию, 0 - абсолютная проэрачиость)
background - rgba(40, 44, 52, 0.8)
```

Заставка lockscreen

```
$ yay -S i3lock-fancy-git
$ nano -/.confg/13/config
# параметр = В.денает фоном lockscreen скриншот экрана с разматием
bindsym Ctrl+$m+l exec i3lock-fancy -gpf Ubuntu -- scrot -z
```

Репозиторий.

Сохранение/восстановление рабочего пространства

```
# Сохранение
i3-save-tree --workspace 1 > -/.i3/workspace-1.json
# Восстановление
i3-msg "workspace 1; append_layout -/.i3/workspace-1.json"
```

Требует установки зависимостей.

Документация.

Частые проблемы

Что делать, если каталоги открываются в VSCode?

Существует файл /usr/share/applications/mimeinfo.cache. В нем хранятся ассоциации между mime-типами и приложениями. Его редактирование исправляет проблему, НО такое решение является временным, так как этот файл генерируется при каждом обновлении системы из *.desktop файлов.

Hужно отредактировать MimeTуре (я ero просто закомментировал) в /usr/share/applications/visual-studio-code.desktop и обновить mimeinfo.cache:

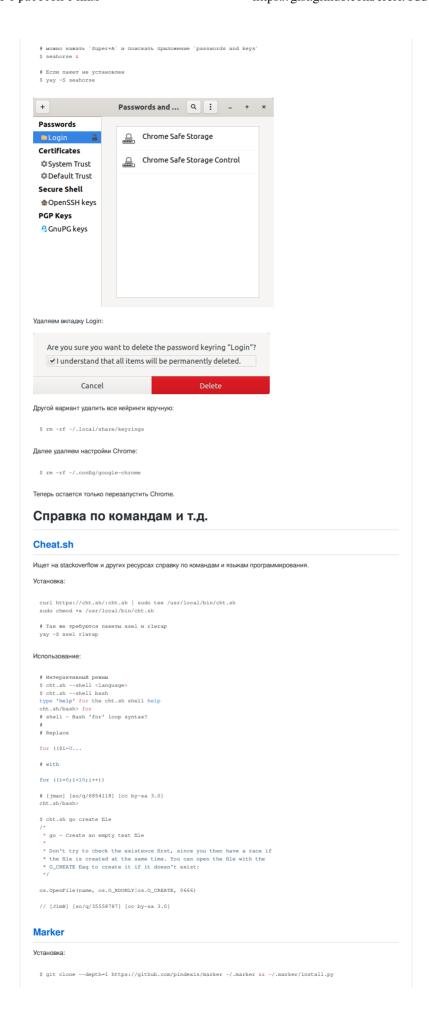
```
$ sudo nano /usr/share/applications/visual-studio-code.desktop
...
# MimeType-text/plain;inode/directory;
...
$ sudo update-desktop-database /usr/share/applications
```

Enter password to unlock your login keyring

В один прекрасный день Chrome выдаст такое предупреждение, после чего перестанут сохраняться пароли и не будет работать автозаполнение форм для логина.

Решение

Стр. 15 из 25 29.07.2020, 21:10



Стр. 16 из 25 29.07.2020, 21:10

```
· Ctrl-space: search for commands that match the current written string in the command-line

    Ctrl-k (or marker mark): Bookmark a command.

    Ctrl-t: place the cursor at the next placeholder, identified by '{{anything}}'

    marker remove: remove a bookmark

TI DR
Этим я пользовался до cht.sh.
Получаем краткую справку по команде:
  $ tldr nvm

✓ Page not found. Updating cache...

✓ Creating index...
     Install, uninstall or switch between Node.js versions.

Supports version numbers like *0.12* or "v4.2*, and labels like *stable*, *system*, etc.

Homepage: https://github.com/creationix/nvm.
      - Install a specific version of Node.js:
nvm install node_version
      - Use a specific version of Node.js in the current shell: nvm use node_version
     - Set the default Node.js version:
nvm alias default node_version
      - List all available Node.js versions and highlight the default one:
      - Uninstall a given Node.js version:
      - Launch the REPL of a specific version of Node.js:
nvm run node_version --version

    Execute a script in a specific version of Node.js:
nvm exec node_version node app.js

Шпаргалка по командам Shell
    # Основы синтаксиса
  # $1, $2, $3, ... are the positional parameters.
# "$8" is an array-like construct of all positional parameters, {$1, $2, $3 ...}.
# "$*" is the ITS expansion of all positional parameters, $1 $2 $3 ....
# $# is the number of positional parameters.
# $- current options set for the shell.
# $$ pid of the current shell (not subshell).
# $_ most recent parameter (or the abe path of the command to start the current shell immediately after startu
# $ITS is the (input) field separator.
# $? is the most recent foreground pipeline exit status.
# $! is the PID of the most recent background command.
# $0 is the name of the shell or shell script.
   # присвоить значение пере
x=42
   # Для вывода используем echo и printf
    echo "x=$x"
echo "x=${x}"
   printf "x=%s\n" >
   # $ { varname :- word }
# If varname exists and isn't null, return its value; otherwise return word.
   # Purpose:
# Returning a default value if the variable is undefined.
   # Example:
# ${count:-0} evaluates to 0 if count is undefined.
   # $ { varname :- word} # If varname exists and isn't null, return its value; otherwise set it to word and then return its value. Posi
   # Purpose:
# Setting a variable to a default value if it is undefined.
   # $ { varname :? message } # If varname exists and isn't null, return its value; otherwise print varname : followed by message, and abort
   \ensuremath{\sharp} Purpose: \ensuremath{\sharp} Catching errors that result from variables being undefined.
   # Example:
# (count :?" undefined! " ) prints "count: undefined!" and exits if count is undefined.
   # $ { varname :+word }
# If varname exists and isn't null, return word; otherwise return null.
   # Testing for the existence of a variable.
   \mbox{\$ Example:} \\ \mbox{\$ $ \{count :+ 1\} returns 1 (which could mean "true") if count is defined.} \\
```

Стр. 17 из 25 29.07.2020, 21:10

```
# $ { varname : offset }
# $ { varname : offset:length }
# Purpose:
# Returning parts of a string (substrings or slices).
# Example:
# If count is set to frogfootman, $ {count :4} returns footman. $ {count :4:4} returns foot.
# Экспорт глобальной переменной export VAR-42
# Генерация строк с помощью Brace expansion 
 $ echo a{d,c,b}e
echo ${arr[0]} ${arr[1]}
${arr[*]} # Все записи в массиве
${|arr[*]} # Все видексы в массиве
${|arr[*]} * Количество записей в массиве
${arr[0]} # Длина первой записи (нумерация с нуля)
echo "Array size: ${#array[*]}" # Выводим размер массива
echo "Array items:" # Выводим записи массива for item in ${array[*]}
     printf " %s\n" $item
echo "Array indexes:" # Выводим индексы массива
for index in ${!array[*]}
printf " %d\n" $index
done
echo "Array items and indexes:" # Выводим записи массива с их индексами for index in \{\{array[*]\}\}
     printf "%4d: %s\n" $index ${array[$index]}
# Следующий пример покажет, как кавычки и конструкции без кавычек возвращают строки (особенно важно, когда в э
array-("first item" "second item" "third" "item")
echo "Number of items in original array: {\{\{array[*]\}}" for ix in {\{\{array[*]\}}
     printf " %s\n" "${array[$ix]}"
arr*(${array[*]})
echo "After unquoted expansion: ${\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\fir}\fir\f{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\frac{\f{
     printf " %s\n" "${arr[$ix]}"
arr=("${array[*]}")
echo "After * quoted expansion: ${\frac{1}{4}}"
for ix in ${!arr[*]}
     printf " %s\n" "${arr[$ix]}"
arr=("${array[@]}")
echo "After @ quoted
for ix in ${!arr[*]}
                                                  ted expansion: ${#arr[*]}"
go
   printf " %s\n" "${arr[$ix]}"
done
for i in $(seq 1 10);
for ((i = 0 ; i < max ; i++ ))
     echo $i
for i in {0..10}
for w in word1 word2 word3
doSomething($w)
while [ "$i" -le "$num" ]; do
printf 'counter is at %d\n' "$i"
     i=$(( i + 1 ))
```

Стр. 18 из 25 29.07.2020, 21:10

```
if [ "$seconds" -eq 0 ); then
    timezone_string="z"
elif [ "$seconds" -gt 0 ); then
    timezone_string=$[printf "%02d:%02d" %((seconds/3600)) %(((seconds / 60) % 60)))
timezone_string=$(printf
else
echo "Unknown parameter"
fi
# В условиях нужно использовать двойные скобки
# [ is just a regular command with a weird name.
# ] is just an argument of [ that prevents further arguments from being used.
 [[a = a 66 b = b]]: true, logical and  [[a = a 66 b = b]: syntax error, 66 parsed as an AND command separator cmd1 66 cmd2]ust an argument of [th
# x='a b'; [[ $x = 'a b' ]]: true, quotes not needed
# x='a b'; [ $x = 'a b' ]: syntax error, expands to [ a b = 'a b' ]
# Подробнее тут:
# <https://stackoverflow.com/a/47576482>
 # -eq
# is equal to
[ "$a" -eq "$b" ]
 # -ne
# is not equal to
[ "$a" -ne "$b" ]
# -gt
# is greater than
[ "$a" -gt "$b" ]
# -ge
# is greater than or equal to
[ "$a" -ge "$b" ]
# -lt
# is less than
[ "$a" -lt "$b" ]
# -le
# is less than or equal to
[ "$a" -le "$b" ]
\# < \# is less than (within double parentheses)
 (("$a" < "$b"))
\ensuremath{\sharp} <= \ensuremath{\sharp} is less than or equal to (within double parentheses)
(("$a" <- "$b"))
\ensuremath{\text{\#}} > \ensuremath{\text{\#}} is greater than (within double parentheses)
# >=
# is greater than or equal to (within double parentheses)
(("$a" >= "$b"))
# <http://tldp.org/LDP/abs/html/comparison-ops.html>
 # [ - это всего лишь команда, последним аргументом которой всегда должна быть "]"! Все операторы описаны в man
# Это вполне себе валидное выражение "[" 1 -eq 0 "]" || echo fail
# Строка слева всегда имя переменной x-42; [[ "x" -eq "42" ]] || echo fail
 # выражение слева всегда переменная
case $VAR in
foo) ...;
bar) ...;
# Bce осталь
*) ...;
esac
 # Объявление функции
foo() {
 # Аргументы функции
$1..$N
   # Локальная переменная local x=42
   # Теперь в $1 будет $2, в $2 - $3 и т.д.
die() { echo "$*" 1>&2 ; exit 1; }
[ "$#" -eq 2] || die "Needs 2 arguments, input and output"
# The syntax is token-level, so the meaning of the dollar sign depends on the token it's in. The expression $(
echo "Today is $(date). A fine day."
# Управление выводом
```

Стр. 19 из 25 29.07.2020, 21:10

```
# Направить stdout одной программы в stdin другой command1 | command2
# Перенаправление stdout и stderr
# Создать либо перезаписать файл, добавив строку
# Создать файл, если его не существует и дописать строку в конец
# Направить stderr команды в файл
# stdout 2-ой команды, является stdin для первой command1 <<< command2
джигурда
ЕОF
. -----
# Первой строкой скрипта идет Shebang, торый указывает какой интерпретатор использовать

    Выполнение скриптов

# Выполнит скрипт в текущем процессе (переменные и функции, объявленные внутри скрипта станут доступны в терми
$ source /path/to/file
# Запуск команд
# Запустить процесс и вернуть его дескриптор (при закрытии терминала будет остановлена)
$ соптапа 6
# То же самое за исключением того, что процесс не будет остановлен при закрытии терминала \$ nohup command \&
# Обновить все установленные пакеты
# Обновить в т.ч. с пакетами для разработчика
$ yay -Syu --devel --timeupdate
# Статистика по пакетами
$ yay -Ps
\mbox{\#} Generates development package DB used for devel updates \mbox{\$} yay -Y --gendb
# Ошибки с удалением зависимостей
$ yay -Rns gnome-extra
checking dependencies...
error: failed to prepare transaction (could not satisfy dependencies)
:: nautilus: removing nautilus-sendto breaks dependency 'nautilus-sendto'
$ yay -Rdd nautilus-sendto
# Теперь можно снести gnome-extra
# Все сервисы
# Покажут только включенные
$ systemctl
$ systemctl list-units --type service
# + выключенные
$ systemctl list-unit-files --type service
$ sudo systemctl enable docker
$ sudo systemctl disable docker
$ sudo systemctl restart nginx.service
```

Стр. 20 из 25 29.07.2020, 21:10

```
$ sudo systemctl start application.service
$ sudo systemctl start application.service
$ systemctl status nginx.service
# Замена в тексте
$ echo "This is a test" | sed 's/test/another test/'
# Ключ -е позволяет выполнить несколько команд:
# sed -e 's/This/That/; s/test/another test/'
# Перевод регистра
$ echo lowercase | tr '[:lower:]' '[:upper:]'
LOWERCASE
•
# Файловая система
# В Linux все файлы. Регулярные файлы - обычные файлы, каталоги - это файлы содержащие список файлов и т.д.
# Копировать каталог и все вложенные файлы
$ cp -r <src> <dst>
# Полный путь до файла
$ realpath example.txt
/home/username/example.txt
$ ls
$ tldr ls
# Вывести информацию о владельце и группе файла
$ ls -ld /path/to/file
$ stat /path/to/file
# Вывести все вложенные файлы
$ ls -R <path>
$ find <path> -print
# Покажет имя и размер
$ du -a <path>
$ mkdir <target>
# Создание каталога вместе с родительскими каталогами, если тех не существует \ mkdir -p <target>
# Создать каталог с определенными правами
$ mkdir -m 0750 <directory>
# Создание множества каталогов
$ mkdir foo bar baz
# или
$ mkdir prefix-{foo,bar,baz}
# Такой же трюк работает при создании файлов $ touch {foo,bar,baz}.txt
# Создать мягкую ссылку на файл либо заменить ее новой
$ ln -sf path/to/new_file path/to/symlink
# Мягжая ссылка содержит путь до файла. Жесткая ссылается на inode, искомый 
# файл при перемещении остается доступен по ссылке и невозможно ссылаться на 
# файл на другом устройстве
# Заархивировать каталог
$ tar -czvf filename.tar.gz directory
# Для извлечения файлов проще всего пользоваться плагином Oh My ZSH extract
$ paste file1.txt file2.txt > fileresults.txt
# Удалить файлы старше 5 дней
$ find /path/to/files* -mtime +5 -exec rm {} \;
# Поиск фала по имени в специальной базе

$ locate -e login.keyring

/home/sergey/.local/share/keyrings/login.keyring
$ yay -S mlocate
# Перед первым запуском следует выполнить
$ sudo updatedb
# После установки будет доступен сервис updatedb.timer, который будет ежедневно обновлять базу
1 384 522 files
119 423 666 bytes in file names
38 722 751 bytes used to store database
```

Стр. 21 из 25 29.07.2020, 21:10

```
# Покажет что куда смонтировано (можно свободное место узнать) \$ df -h --total
# Узнать на каком разделе смонтирован каталог
# Просмотр числа inode
$ df -i
# Просмотр содержимого фйала с навигацией
$ less /path/to/file
# или более короткая версия в ZSH
$ < /path/to/file
# Просмотр логов в обратном порядке
$ tail -r /var/log/syslog | less
# Вывести строки не соответствующие шаблону $ grep -Pv <exclude_pattern> <filename>
# Создать файл, забитый null-байтами
$ dd if-/dev/zero of-/tmp/nullbytes bs-1M count-1
# Конфертировать .md в .rst
$ pip install m2r
$ m2r --help
# Конвертировать .webp в .png
$ yay -S libwebp
$ dwebp file.webp -o file.png
# Увеличиваем размер логического раздела
$ sudo lvresize -L +10GB /dev/mapper/lvm-root
# После lvresize нужно обязательно изменить размер файловой системы $ sudo resize2fs /dev/mapper/lvm-root
# Делает то же самое, что и две команды выше
$ sudo lvresize -r -L +10GB /dev/mapper/lvm-root
# Список логических разделов LVM
$ sudo lvscan
                          '/dev/lvm/root' [40.00 GiB] inherit
'/dev/lvm/home' [20.00 GiB] inherit
  ACTIVE
# Подробная информация о диске
$ sudo smartctl -a /dev/nvme0
# Cerb
# связанного процесса
$ netstat -plantu
# Все запущенные сервера на хосте
$ netstat -lnt
  .
≢ Шрифты
# Список установленных шрифтов
$ fc-list
# Буфер обмена
$ echo 123 | xclip -sel clip
# Копировать содержимое файла в буфер обмена
$ xclip -sel clip < ~/.ssh/github_rsa.pub
# Конвертировать файл в base64 и скопировать в буфер обмена

$ file="test.docx"

$ base64 -w 0 $file | xclip -selection clipboard
# Git
# Первоначальная настройка (без флага --global локального проекта)
$ git config --global user.name <yourname>
$ git config --global user.email <email>
# Сменить ветку
$ git checkout <branchname>
```

Стр. 22 из 25 29.07.2020, 21:10

```
# Создать ветку и переключится на нее $ git checkout -b <br/>branchname>
# Объединить текущую ветку с branchname
$ git merge <branchname>
\sharp Добавить изменения (все файлы в каталоге и вложенихы) \$ git add .
# Закоммитить изменения (сделать описание)
$ git commit -m "Тест"
# Если не были добавлены новые файлы, а лишь производились изменения в уже добавленных, то можно использовать 
$ git commit -am "Tece"
# Обновить репозиторий на сервере
$ git push
\$ Выгрузить изменения с сервера (тоже самое, что git fetch + git merge) \$ git pull
# Копировать репозиторий
$ git clone https://github.com/someuser/repo.git [<assigneddirectory>]
# Посмотреть историю
$ git log --graph --oneline --decorate --all
# Посмотреть отличия по сравнения с предыдущей версией 
$ git diff 871d36b [<filename>]
# Сохранить изменения
$ git stash [save "my stash message here"]
# Удалить stach и применить его
$ git stash pop [stash@{1}]
# Список stash
$ git stash list
# Посмотреть что там лежит
$ git stash show stash@{0}
$ git stash apply
$ git stash drop stash@{2}
$ git stash clear
# Search tags for a particular pattern:
$ git tag -1 <tag-pattern>
# Create a Lightweight Tag:
$ git tag <tag-name>
# Create an Annotated Tag:
$ git tag -a <tag-name> -m <tag-message>
# Create a tag for a specific commit:
$ git tag -a <tag-name> <commit-checksome>
# Push a specific tag to remote:
$ git push origin --tags
# Checkout a specific to local:
$ git checkout -b <brach-name> <tag-name>
# Языковые настройки
# Список влюченных локалей
$ locale -a
# В этом файле находится список всех поддерживаемых локалей
# Раскомментируем нужную
$ sudo nano /etc/locale.gen
# Можно так же локали так добавлять
$ sudo locale-gen de_DE.UTF-8
# Меняем язык системы (нужно перегрузиться)
echo "LANG=de_DE.UTF-8" > /etc/locale.conf
 "
# Генерация паролей
$ yay -S pwgen
$ pwgen -cnsy 10 1
1u_dr<ZLH;</pre>
$ pip install xkcdpass
$ xkcdpass -n 3 -d -
backdrop-unruly-yodel
```

Стр. 23 из 25 29.07.2020, 21:10

f!/usr/bin/env python

from pathili import Path

import re

base_path = Path(_file__).resolve().parent

with (base_path / 'Arch-Instruction.md').open() as fp:

contents = fp.read()

f convana nopsammyew Kowreur, yyansa sce scrasks Kona

contents = re.sub(r'''.*?''', '', contents, flags=re.S)

f Temps snew saronosks

beaders = re.findall(''*.'', contents, flags=re.S)

for header in headers:

i = len(header)

i = len(header)

for header in headers:

i = len(header)

for header in headers:

i = len(header)

g temps suppeases counse

name = name.strip()

f Temps suppeases counse

name = re.sub(r'('(.*?))\(',.?\)', r'\1', name)

uri = name.lower()

f Temps suppeases counse

name = re.sub(r'('\N)=','', '', uri)

f Temps suppeases counse counse

uri = re.sub(r'('\N)=', '', uri)

f Temps suppeases counse counse

uri = re.sub(r'('\N)=', '', uri)

f Temps suppeases counse counse

uri = re.sub(r'('\N)=', '', uri)

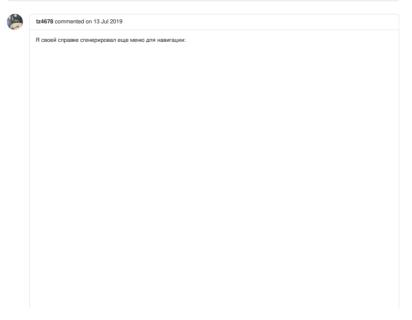
f Temps suppeases counse counse

uri = re.sub(r'('\N)=', '', uri)

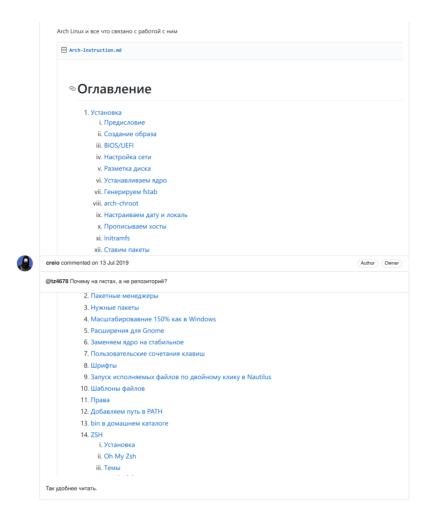
f Temps suppeases counse counse

uri = uri.replace(' ', '-')

print(' ' * (depth - 1) + f'l. [{name}](f{uri})')



Стр. 24 из 25 29.07.2020, 21:10



Стр. 25 из 25 29.07.2020, 21:10