Общее

- Автозапуск скрипта для Gnome
- Автоматическое обновление Flatpak пакетов
- Включение поддержки мультимедиа в Opera
- Включение поддержки rsa шифрования для ssh
- Включение поддержки Wayland для NoMachine
- Включение подсветки синтаксиса в nano
- Включение ускорения Firefox
- Генерация dbparam.pem
- Генерация QR кодов
- Добавить текст в начало или в конец каждой строки
- Конвертирование форматов изображений
- Копирование диска по сети
- Некорректные цвета в Firefox
- Обновление модулей Python
- Отключение защиты от уязвимостей
- Отключение защиты от Meltdown и Spectre
- Отключение проверки состояния сети
- Отключение экрана консоли
- Отключение IPv6
- Отключение Super+N в Gnome
- Разрешение запуска графических приложений под sudo
- Скрипт подключения к RDP
- Служба Tracker
- Создание файла подкачки
- Удаление агента Itarian
- Удаление загрузочных записей BIOS/UEFI
- Установка пакетного менеджера NIX

- Установка unbound
- cbonsai
- Cockpit SSL
- Gnome - увеличение громкости более 100%
- IPIP туннель
- LVM
- Nextcloud
- Nextcloud automount with rclone

Автозапуск скрипта для Gnome

Если есть необходимость запускатьскрипт при каждом запуске Gnome окружения, необходимо создать desktop файл автозапуска:

```
nano ~/.config/autostart/scriptname.desktop
```

Где **scriptname** - название файла, которое может быть любым, но логичнее всего сделать его аналогичным названию файла скрипта.

Вставить в него содержимое:

> [Desktop Entry]
> Name=my-script
> GenericName=my-script
> Comment=My Script
> Exec='/home/user/.config/autostart/scriptname.sh'
> Terminal=false
> Type=Application
> X-GNOME-Autostart-enabled=true

Где:

- **my-script** - название скрипта в графическом окружении
- **My Script** - описание скрипта
- **/home/user/.config/autostart/scriptname.sh** - пусть до скрипта, который необходимо запускать

После перезапуска графического окружения, скрипт автоматически будет запущен в фоне.

Автоматическое обновление Flatpak пакетов

Установка службы обновления для пользователей

```
sudo nano /etc/systemd/user/update-user-flatpaks.service
```

> [Unit]
> Description=Update user Flatpaks
>
> [Service]
> Type=oneshot
> ExecStart=/usr/bin/flatpak update --assumeyes --noninteractive --user
>
> [Install]
> WantedBy=default.target

Установка службы обновления для системы

```
sudo nano /etc/systemd/system/update-system-flatpaks.service
```

> [Unit]
> Description=Update system Flatpaks
> After=network-online.target
> Wants=network-online.target
>
> [Service]
> Type=oneshot
> ExecStart=/usr/bin/flatpak update --assumeyes --noninteractive --system
>
> [Install]
> WantedBy=multi-user.target

Установка таймера для пользователей

```
sudo nano /etc/systemd/user/update-user-flatpaks.timer
```

> [Unit]
> Description=Update user Flatpaks daily
>
> [Timer]
> OnCalendar=daily
> Persistent=true
>
> [Install]
> WantedBy=timers.target

Установка таймера для системы

```
sudo nano /etc/systemd/system/update-system-flatpaks.timer
```

> [Unit]
> Description=Update system Flatpaks daily
>
> [Timer]
> OnCalendar=daily
> Persistent=true
>
> [Install]
> WantedBy=timers.target

Включение и запуск служб

```
systemctl --user enable --now update-user-flatpaks.timer
sudo systemctl --system enable --now update-system-flatpaks.timer
```

Включение поддержки мультимедиа в Opera

Создать или скачать готовый файл fix_opera.sh и внести в него скрипт

```
nano fix-opera.sh
```

```
#!/usr/bin/env bash

TMP_DIR="/tmp"
APP_DIR="/usr/lib64/opera"

curl -s https://api.github.com/repos/nwjs-ffmpeg-prebuilt/nwjs-ffmpeg-prebuilt/releases/latest
| grep -wo "https.*-linux-x64.zip" | wget -qi - -P "$TMP_DIR"
mv "$TMP_DIR"/*-linux-x64.zip "$TMP_DIR"/libffmpeg.zip
unzip "$TMP_DIR"/libffmpeg.zip
rm -rf "$TMP_DIR"/libffmpeg.zip
sudo mv "$TMP_DIR"/libffmpeg.so "$APP_DIR"/.
```

Сделать его исполняемым, запустить и ввести пароль

```
chmod +x fix-opera.sh
./fix-opera.sh
```

Перезапустить браузер

Включение поддержки rsa шифрования для ssh

Включение поддержки RSA

```
echo "HostKeyAlgorithms +ssh-rsa" >> /etc/ssh/sshd_config
```

Перезапуск сервиса

```
systemctl restart sshd
```

Включение поддержки Wayland для NoMachine

Вариант 1

```
sudo nano /usr/NX/etc/node.cfg
```

> WaylandModes "compositor,drm"

```
sudo systemctl restart display-manager
sudo /etc/NX/nxserver --restart
```

Вариант 2

```
sudo /etc/NX/nxserver --egl-capture yes
sudo nano /usr/NX/etc/node.cfg
```

> WaylandModes "egl,compositor,drm"

```
sudo systemctl restart display-manager
sudo /etc/NX/nxserver --restart
```

Включение подсветки синтаксиса в nano

Если файл конфига есть, то:

```
nano ~/.config/nano/nanorc
```

Если файла конфига нет, то:

```
mkdir -p ~/.config/nano; cd !$; nano nanorc
```

> include "/usr/share/nano/*.nanorc"

Включение ускорения Firefox

В **about:config** изменить

Включение аппаратного ускорения:

```
layers.acceleration.force-enabled = true
webgl.force-enabled = true
gfx.webrender.enabled = true
gfx.webrender.all = true
dom.webgpu.enabled = true
```

Включение аппаратного декодирования:

```
widget.wayland-dmabuf-vaapi.enabled = true
media.ffmpeg.vaapi.enabled = true
media.ffmpeg.low-latency.enabled = true
media.navigator.mediadatadecoder_vpx_enabled = true
media.ffvpx.enabled = false
media.rdd-ffvpx.enabled = false
media.rdd-vpx.enabled = false
```

Перезапустить браузер

Генерация dbparam.pem

```
sudo openssl dhparam -out /etc/ssl/certs/dhparam.pem 4096
```

Генерация QR кодов

Установить qrencode, после чего выполнить необходимую команду по генерации.

Генерация ссылки

```
qrencode -m 0 -l H -s 6 -o qrcode.png "https://www.tiukov.com"
```

Генерация геопозиции

```
qrencode -m 0 -l H -s 6 -o "grcode.png" "geo:12.459444,99.978611"
```

Генерация номера телефона

```
qrencode -m 0 -l H -s 6 -o "grcode.png" "tel: +79876543210"
```

Генерация Wi-Fi подключения

```
qrencode -m 0 -l H -s 6 -o "grcode.png" "WIFI:T:WPA2;S:SSID_WIFI_NAME;P:P@$$W0RD;;"
```

Генерация создания письма

```
qrencode -m 0 -l H -s 6 -o "grcode.png" "mailto:mail@example.com?subject=Test&body=Test mail message"
```

Генерация SMS

```
qrencode -m 0 -l H -s 6 -o "grcode.png" "smsto:+79876543210,Test message"
```

Добавить текст в начало или в конец каждой строки

С помощью sed (3 варианта):

```
sed 's/^/ТЕКСТ ДО/' file.txt
sed -e 's/^\(.*\)$/ТЕКСТ ДО\1/' file.txt
sed 's/.*/ТЕКСТ ДО&/' file.txt

sed 's/$/ТЕКСТ ПОСЛЕ/' file.txt
sed -e 's/^\(.*\)$/\1ТЕКСТ ПОСЛЕ/' file.txt
sed 's/.*/&ТЕКСТ ПОСЛЕ/' file.txt

sed 's/^/ТЕКСТ ДО/; s/$/ТЕКСТ ПОСЛЕ/' file.txt
sed -e 's/^\(.*\)$/ТЕКСТ ДО\1ТЕКСТ ПОСЛЕ/' file.txt
sed 's/.*/ТЕКСТ ДО&ТЕКСТ ПОСЛЕ/' file.txt
```

С помощью awk:

```
awk '{ print "ТЕКСТ ДО" $0 }' file.txt

awk '{ print $0 "ТЕКСТ ПОСЛЕ" }' file.txt

awk '{ print "ТЕКСТ ДО" $0 "ТЕКСТ ПОСЛЕ" }' file.txt
```

С помощью perl (3 варианта):

```
perl -lne 'print "ТЕКСТ ДО$_"' file.txt
perl -pe 's/^/ТЕКСТ ДО/' file.txt
perl -lpe '$_="ТЕКСТ ДО$_"' file.txt

perl -lne 'print "$_ТЕКСТ ПОСЛЕ"' file.txt
perl -pe 's/$/ТЕКСТ ПОСЛЕ/' file.txt
```

```
perl -lpe '$_="$_ТЕКСТ ПОСЛЕ"' file.txt

perl -lne 'print "ТЕКСТ ДО$_ТЕКСТ ПОСЛЕ"' file.txt
perl -pe 's/^/ТЕКСТ ДО/; s/$/ТЕКСТ ПОСЛЕ/' file.txt
perl -lpe '$_="ТЕКСТ ДО$_ТЕКСТ ПОСЛЕ"' file.txt
```

Если нужно вывести не на экран, а в файл, то в конце команды добавить:

> “ > new_file.txt

Конвертирование форматов изображений

PNG->JPG

```
for file in *.png; do convert $file "$file.jpg"; done
```

JPG->PNG

```
for file in *.jpg; do convert $file "$file.png"; done
```

Копирование диска по сети

Получатель

```
nc -l $PORT | pbzip2 -d | dd of=/hdd.img obs=1M
```

Отправитель

```
pv /dev/$DISK | pbzip2 -9 | nc $IP $PORT
```

$DISK - Идентификатор диска или раздела (например **sda**)

$IP - IP адрес устройства, куда нужно отправить данные

$PORT - Порт для подключения

Вмнесто **/hdd.img** можно указать другое расположение или диск, для переноса данных 1к1.

Для работы нужны: **netcat**, **dd**, **pv** и **pbzip2**

Некорректные цвета в Firefox

В некоторых случаях цвета отдельных элементов сайта могут отображаться некорректно при использовании темной системной темы. Например, это наиболее часто проявляется на формах ввода. Исправляется сменой настроек браузера.

- Открыть расширенные настройки браузера через **about:config**
- Элементу **browser.display.use_system_colors** выставить значение **False**
- Элементу **widget.content.allow-gtk-dark-theme** выставить значение **False**

Обновление модулей Python

Проверить список устаревших модулей

```
pip3 list --outdated
```

Вариант 1

```
pip3 list --outdated --format=freeze | grep -v '^\-e' | cut -d = -f 1 | xargs -n1 pip3 install -U
```

Вариант 2

```
pip3 list -o | cut -f1 -d' ' | tr " " "\n" | awk '{if(NR>=3)print)' | cut -d' ' -f1 | xargs -n1 pip3 install -U
```

Отключение защиты от уязвимостей

Отредактировать параметр **GRUB_CMDLINE_LINUX** загрузчика

```
sudo nano /etc/default/grub
```

Если ядро **старее** 5.1.13

> GRUB_CMDLINE_LINUX="noibrs noibpb nopti nospectre_v2 nospectre_v1 l1tf=off nospec_store_bypass_disable no_stf_barrier mds=off tsx=on tsx_async_abort=off mitigations=off"

Если ядро **новее** 5.1.13

> GRUB_CMDLINE_LINUX="mitigations=off"

Обновить загрузчик

```
sudo update-grub
```

Перезапустить систему

```
sudo reboot
```

Проверить актуальное состояние защиты

```
sudo lscpu
```

Отключение защиты от Meltdown и Spectre

Для отключения защиты необходимо отредактировать строку загрузки, добавив необходимые параметры:

```
sudo nano /etc/default/grub
```

> GRUB_CMDLINE_LINUX_DEFAULT="... noibrs noibpb nopti nospectre_v2 nospectre_v1 l1tf=off nospec_store_bypass_disable no_stf_barrier mds=off tsx=on tsx_async_abort=off mitigations=off"

```
sudo update-grub
sudo reboot
```

Отключение проверки состояния сети

Внести изменения в файл конфигурации NetworkManager:

```
sudo cat >> /etc/NetworkManager/NetworkManager.conf <<EOF
[connectivity]
enabled=false
EOF
```

Перезапустить систему:

```
sudo reboot
```

Отключение экрана консоли

```
sudo nano /etc/default/grub
```

> GRUB_CMDLINE_LINUX="consoleblank=120"

```
sudo update-grub
```

Отключение IPv6

Применить настройки без перезапуска

```
sudo sysctl -w net.ipv6.conf.all.disable_ipv6=1
sudo sysctl -w net.ipv6.conf.default.disable_ipv6=1
sudo sysctl -w net.ipv6.conf.lo.disable_ipv6=1
```

Внести изменения в системную конфигурацию

```
echo "net.ipv6.conf.all.disable_ipv6=1" >> /etc/sysctl.conf
echo "net.ipv6.conf.default.disable_ipv6=1" >> /etc/sysctl.conf
echo "net.ipv6.conf.lo.disable_ipv6=1" >> /etc/sysctl.conf
```

Еси после перезагрузки сервера ipv6 не отключился, внести изменения в /etc/rc.local

```
/etc/sysctl.d
/etc/init.d/procps restart
```

Для отключения через загрузчик, добавить к параметру ядра

> GRUB_CMDLINE_LINUX_DEFAULT="ipv6.disable=1"
> GRUB_CMDLINE_LINUX="ipv6.disable=1"

Обновить загрузчик

```
update-grub
```

Отключение Super+N в Gnome

Создать скрипт

```
nano disable_super_n.sh
```

Вставить в него содержимое

```
#!/usr/bin/env bash

WSNUM=(1 2 3 4 5 6 7 8 9)
for WSN in ${WSNUM[@]}
do
  gsettings set org.gnome.shell.keybindings switch-to-application-$WSN []
done
```

Сделать скрипт запускаемым

```
chmod +x disable_super_n.sh
```

Выполнить скрипт

```
./disable_super_n.sh
```

Разрешение запуска графических приложений под sudo

Из под обычного пользователя выполнить команды

```
echo "xhost local:root" >> ~/.xinitrc
xhost local:root
```

Скрипт подключения к RDP

Предварительно установить пакет freerdp

```
sudo nano /opt/rdp.sh
```

> #!/bin/bash
> #GDK_BACKEND=x11
> RDPCLIP="-clipboard"
> RDPCOMP="-compression"
> RDPSHARE="-home-drive"
> RDPSERVER=`zenity --entry --title="RDP" --text="Server address"`
> RDPLOGIN=`zenity --entry --title="RDP" --text="Server login"`
> RDPPASS=`zenity --entry --hide-text --title="RDP" --text="Server password"`
>
> RDPCHOICE=`zenity --list --title "RDP" --text "Connection options" --checklist --column "" --column "Select" 1 "Clipboard" 2 "Compression" 3 "Share home folder"`
>
> case "${RDPCHOICE}" in
> *"Clipboard"*)
> RDPCLIP="+clipboard"
> ;;
> esac
>
> case "${RDPCHOICE}" in
> *"Compression"*)
> RDPCOMP="+compression"
> ;;
> esac
>
> case "${RDPCHOICE}" in
> *"Share home folder"*)
> RDPSHARE="+home-drive"
> ;;
> esac
>
> xfreerdp /u:$RDPLOGIN /p:$RDPPASS /v:$RDPSERVER /cert-ignore /network:auto /workarea /dynamic-resolution /disp /video /nsc /microphone /sound /smartcard /gfx /rfx +window-drag +offscreen-cache +menu-anims +gfx-progressive +gfx-

> small-cache +gfx-thin-client +glyph-cache +multitransport +gestures +bitmap-cache +aero +fonts $RDPSHARE $RDPCLIP $RDPCOMP &> /dev/null

При использовании Wayland, раскомментировать строку GDK_BACKEND=x11

Скачать иконку для приложения и положить в домашнюю директорию



```
sudo mv ~/rdp.png /opt/rdp.png
nano ~/.local/share/applications/rdp.desktop
```

> [Desktop Entry]
> Exec=/opt/rdp.sh %u
> Name=RDP
> GenericName=RDP
> X-GNOME-FullName=RDP
> Icon=/opt/rdp.png
> Type=Application
> Terminal=False

В списке приложений появится приложение с названием RDP

Выбор опций при подключении влияет на следующее:

- Clipboard - включение общего буфера обмена между компьютером и удаленным сервером
- Compression - включение сжатия (включать при работе через интернет или при медленной локальной сети)
- Share home folder - подключение домашней директории в виде сетевого диска на удаленном сервере

Служба Tracker

Отключение

```
systemctl --user mask tracker-extract-3.service tracker-miner-fs-3.service tracker-miner-rss-3.service tracker-writeback-3.service tracker-xdg-portal-3.service tracker-miner-fs-control-3.service
```

```
tracker3 reset -s -r
```

Перезапустить компьютер

Включение

```
systemctl --user unmask tracker-extract-3.service tracker-miner-fs-3.service tracker-miner-rss-3.service tracker-writeback-3.service tracker-xdg-portal-3.service tracker-miner-fs-control-3.service
```

Перезапустить компьютер

Создание файла подкачки

Создаем файл размером в 4GB:

```
sudo fallocate -l 4G /swapfile
```

Выставляем права:

```
sudo chmod 600 /swapfile
```

Превращаем его в swap раздел:

```
sudo mkswap /swapfile
```

Подключаем:

```
sudo swapon /swapfile
```

Добавляем авто подключение при старте системы:

```
echo '/swapfile none swap sw 0 0' |sudo tee -a /etc/fstab
```

Удаление агента Itarian

```
nano uninstall_ccc.sh
```

> #!/bin/bash
> if find /etc/systemd/system/itsm.service -name itsm.service -print -quit;
> then
> systemctl stop itsm && systemctl disable itsm && rm -f
> /etc/systemd/system/itsm.service && rm -rf /opt/COMODO && rm -rf
> /run/comodo && rm -rf /etc/xdg/menus/applications-merged/comodo-
> comodo.menu && rm -rf ~/Desktop/comodo-ccs.desktop
> read -p "REBOOT IS NEEDED, PERFORM NOW? Y/N " n
> echo $n
> if [$n = Y] || [$n = y];
> then
> /sbin/reboot
> fi
> elif find /etc/init/itsm.conf -name itsm.conf -print -quit;
> then
> stop itsm && rm /etc/init/itsm.conf && initctl reload-configuration && rm -rf
> /opt/COMODO && rm -rf /run/comodo && rm -rf /etc/xdg/menus/applications-
> merged/comodo-comodo.menu && rm -rf ~/Desktop/comodo-ccs.desktop
> echo ""
> echo "***COMODO CLIENT COMMUNICATION IS UNINSTALLED***"
> echo ""
> read -p "REBOOT IS NEEDED, PERFORM NOW? Y/N " n
> echo $n
> if [$n = Y] || [$n = y];
> then
> /sbin/reboot
> fi
> else
> echo ""
> echo "***DEVICE IS NOT ENROLLED IN ITSM***"
> echo ""
> fi

```
chmod +x uninstall_ccc.sh
sudo sh uninstall_ccc.sh
```

Удаление загрузочных записей BIOS/UEFI

Посмотреть список записей

```
efibootmgr -v
```

Удалить конкретную запись

```
sudo efibootmgr -b # -B
```

Где # - номер записи без нулей (например для записи Boot0003 необходимо указывать просто 3).

Установка пакетного менеджера NIX

Установка

```
curl -L https://nixos.org/nix/install | sh
```

Добавление списка установленных приложений nix в общий список

```
ln -s /home/$USER/.nix-profile/share/applications/* /home/$USER/.local/share/applications/
```

Установка unbound

```
sudo apt install unbound
wget https://www.internic.net/domain/named.root -qO- | sudo tee /var/lib/unbound/root.hints
sudo nano /etc/unbound/unbound.conf.d/main.conf
```

> server:
> logfile: "/var/log/unbound/unbound.log"
> verbosity: 1
>
> interface: 0.0.0.0
> port: 53
>
> do-ip4: yes
> do-udp: yes
> do-tcp: yes
> do-ip6: no
>
> prefer-ip6: no
>
> #root-hints: "/var/lib/unbound/root.hints"
>
> harden-glue: yes
> harden-dnssec-stripped: yes
>
> hide-identity: yes
> hide-version: yes
>
> use-caps-for-id: no
>
> edns-buffer-size: 1472
>
> prefetch: yes
>
> num-threads: 1
>
> so-rcvbuf: 1m
>
> private-address: 192.168.0.0/16
> private-address: 169.254.0.0/16

```
    private-address: 172.16.0.0/12
    private-address: 10.0.0.0/8
    private-address: fd00::/8
    private-address: fe80::/10

    access-control: 192.168.0.0/16 allow
    access-control: 169.254.0.0/16 allow
    access-control: 172.16.0.0/12 allow
    access-control: 10.0.0.0/8 allow
    access-control: 127.0.0.1/24 allow
```

```
sudo mkdir -p /var/log/unbound
sudo touch /var/log/unbound/unbound.log
sudo chown unbound /var/log/unbound/unbound.log
sudo systemctl disable systemd-resolved --now
sudo systemctl restart unbound
```

cbonsai

Установка из исходников

```
git clone https://gitlab.com/jallbrit/cbonsai
cd cbonsai
sudo make install PREFIX=/usr/ocal
cd ..
rm -rf cbonsai
```

Cockpit SSL

Залить на сервер сертификат (certificate.crt) и ключ (certificate.key)

```
sudo cat certificate.crt > /etc/cockpit/ws-certs.d/ssl.cert
sudo cat certificate.key >> /etc/cockpit/ws-certs.d/ssl.cert
sudo systemctl restart cockpit
```

Gnome - увеличение громкости более 100%

```
gsettings set org.gnome.desktop.sound allow-volume-above-100-percent 'true'
```

IPIP туннель

192.168.2.1 - внутренний адрес узла туннеля
255.255.255.0 - подсеть внутреннего туннеля

Временный туннель

```
sudo apt install ipip
sudo modprobe ipip
sudo modprobe tun
sudo ip tunnel add tun0 mode ipip remote <remote_ip> local <local_ip>
sudo ip link set tun0 up
sudo ip addr add 192.168.2.1/24 dev tun0
```

Постоянный туннель

```
sudo apt install ipip
sudo modprobe ipip
sudo modprobe tun
sudo nano /etc/network/interfaces
```

> auto tun0
> iface tun0 inet static
> address 192.168.2.1
> netmask 255.255.255.0
> network 192.168.2.0
> broadcast 192.168.2.255
> pre-up /sbin/ip tunnel add tun0 mode ipip remote <remote_ip> local <local_ip>
> post-down /sbin/ip tunnel del tun0

```
sudo ifup tun0
```

LVM

Расширение раздела

- Подключаем диск

В данном примере, группа lvm vgmain, подключенный диск sdb, а увеличиваемый раздел - корневой.

- Смотрим текущий список групп:

```
vgdisplay
```

- Добавляем диск в группу:

```
vgextend vgmain /dev/sdb
```

- Смотрим текущий список разделов:

```
lvdisplay
```

- Увеличиваем раздел на размер добавленного диска:

```
lvextend -r -l +100%FREE /dev/vgmain/root
```

- Меняем разметку файловой системы:

```
xfs_growfs / -d
```

Nextcloud

Включение HTTPS

```
sudo nano /var/www/html/config/config.php
```

> 'overwrite.cli.url' => 'https://my.nextcloud.com',
> 'overwritehost' => 'my.nextcloud.com',
> 'overwriteprotocol' => 'https',
> 'htaccess.RewriteBase' => '/',

```
sudo -u www-data php /var/www/html/occ maintenance:update:htaccess
```

Очистка приложений

```
sudo -u www-data php /var/www/html/occ files:scan-app-data
```

Переиндексация файлов

```
sudo -u www-data php /var/www/html/occ files:scan --all
```

Исправление кодировки

```
sudo convmv -f utf-8 -t utf-8 -r --notest --nfc <nextcloud-data-folder>
sudo -u www-data php /var/www/html/occ files:scan --all
```

Отключение индексирования

```
sudo touch <nextcloud-data-folder>/.noindex
```

Отключение сканирования фото

```
sudo touch <nextcloud-data-folder>/.noimage
```

Отключение сканирования медиа

```
sudo touch <nextcloud-data-folder>/.nomedia
```

Превью

```
sudo -u www-data php /var/www/html/occ config:app:set --value="32 64 1024" previewgenerator squareSizes
sudo -u www-data php /var/www/html/occ config:app:set --value="64 128 1024" previewgenerator widthSizes
sudo -u www-data php /var/www/html/occ config:app:set --value="64 256 1024" previewgenerator heightSizes
sudo -u www-data php /var/www/html/occ config:app:set preview jpeg_quality --value="60"
sudo -u www-data php /var/www/html/occ config:system:set preview_max_x --value 2048
sudo -u www-data php /var/www/html/occ config:system:set preview_max_y --value 2048
sudo -u www-data php /var/www/html/occ config:system:set jpeg_quality --value 60
sudo systemctl restart apache2
sudo -u www-data php /var/www/html/occ preview:delete_old
sudo -u www-data php /var/www/html/occ preview:generate-all -vvv
sudo -u www-data php /var/www/html/occ preview:pre-generate
sudo crontab -u www-data -e
```

> */10 * * * * php -f /var/www/html/occ preview:pre-generate

```
sudo nano /var/www/html/config/config.php
```

> 'preview_libreoffice_path' => '/usr/bin/libreoffice',
> 'enable_previews' => true,
> 'enabledPreviewProviders' =>
> array (
> 0 => 'OC\\Preview\\Image',
> 1 => 'OC\\Preview\\Movie',
>),

Миграция в PostgreSQL

```
sudo apt install php-pgsql
sudo -u postgres psql
CREATE USER nextcloud WITH PASSWORD 'p@$$w0rd';
CREATE DATABASE nextcloud TEMPLATE template0 ENCODING 'UNICODE';
ALTER DATABASE nextcloud OWNER TO nextcloud;
GRANT ALL PRIVILEGES ON DATABASE nextclouddb TO nextcloud;
```

```
\q
sudo -u www-data php /var/www/html/nextcloud/occ db:convert-type --all-apps --password
"p@$$w0rd" pgsql nextcloud localhost nextcloud
```

Сказать клиентам, что данные на сервере поменялись

```
sudo -u www-data php /var/www/html/occ maintenance:data-fingerprint
```

Исправление ошибок Redis

```
sudo redis-cli -s /var/run/redis/redis-server.sock flushall
```

Исправление ошибок с датой

```
sudo find /data/folder -mtime +18000 -exec echo {} ";"
sudo touch /data/folder/wrong/file
sudo -u www-data php /var/www/html/occ files:scan --all
```

Nextcloud automount with rclone

Добавить Nextcloud подключение в конфигурацию rclone.

Для примера будет использоваться название подключения - nextcloud.

Сменить в сервисе **{USER_NAME}** на имя пользователя.

Создать в домашней директории точку монтирвоания

```
mkdir ~/Nextcloud
```

Создать сервис

```
sudo nano /etc/systemd/system/rclone-nextcloud.service
```

```
[Unit]
Description=RClone mount for Nextcloud
Documentation=man:rclone(1)
After=network-online.target
Wants=network-online.target
AssertPathIsDirectory=/home/{USER_NAME}/Nextcloud

[Service]
Type=simple
User=tiukov
Group=tiukov
ExecStart=/usr/bin/rclone mount nextcloud: /home/{USER_NAME}/Nextcloud \
    --config=/home/{USER_NAME}/.config/rclone/rclone.conf \
    --vfs-cache-mode full \
    --vfs-cache-max-age 6h \
    --vfs-cache-max-size 50G \
    --buffer-size 16M \
    --dir-cache-time 72h \
    --umask 002 \
```

```
    --uid 1000 \
    --gid 1000 \
    --daemon-timeout 30s
ExecStop=/bin/fusermount -u /home/{USER_NAME}/Nextcloud
TimeoutStartSec=60
TimeoutStopSec=20
KillMode=process
Restart=on-failure
RestartSec=30
StartLimitInterval=300
StartLimitBurst=3

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

Обновить список

```
sudo systemctl daemon-reload
```

Включить и запустить

```
sudo systemctl enable --now rclone-nextcloud.service
```