



**Министерство науки и высшего образования Российской
Федерации
Федеральное государственное бюджетное образовательное
учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ

Информатика и системы управления

КАФЕДРА

Информационная безопасность (ИУ8)

Отчёт
по лабораторной работе №1
по дисциплине «Безопасность систем баз данных»

Выполнил: Песоцкий А. А.,
студент группы ИУ8-61

Проверил: Зенькович С. А.,
ассистент каф. ИУ8

г. Москва,
2020 г.

Содержание

<u>ВСТУПЛЕНИЕ</u>	<u>3</u>
<u>ОПИСАНИЕ ПРОЦЕССА РАБОТЫ</u>	<u>3</u>
1. ПОДГОТОВКА К УСТАНОВКЕ	3
2. УСТАНОВКА	5
3. НАСТРОЙКА СИСТЕМЫ	13
<u>ПЕРЕЧЕНЬ УСТАНОВЛЕННОГО ПО</u>	<u>17</u>

ВСТУПЛЕНИЕ

Цель работы: ознакомиться с процессом установки дистрибутива Linux на виртуальную машину, настройкой конечной системы и SSH-сервера.

ОПИСАНИЕ ПРОЦЕССА РАБОТЫ

Дистрибутив:	Arch Linux
Рабочее окружение:	GNOME

1. ПОДГОТОВКА К УСТАНОВКЕ

Создаём новую виртуальную машину в VirtualBox:

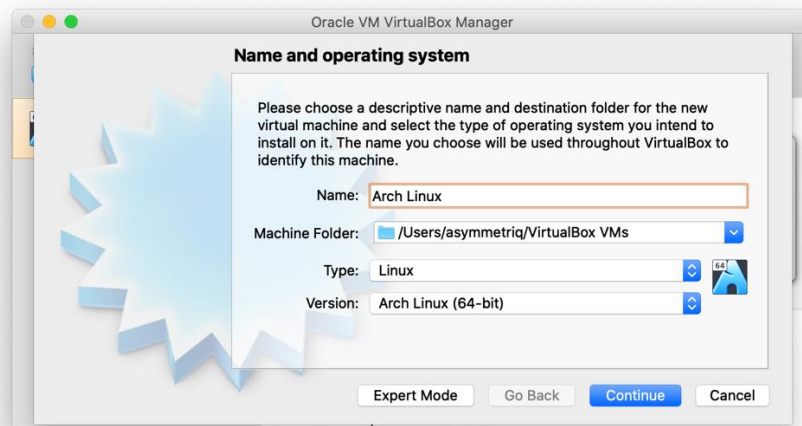


Рисунок 1. Указываем название

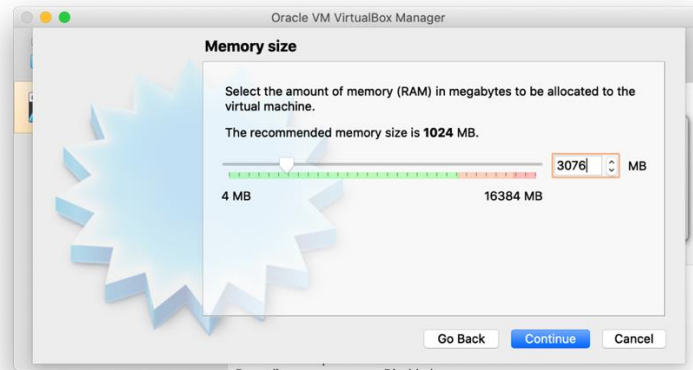


Рисунок 2. Настраиваем количество выделяемой ОЗУ

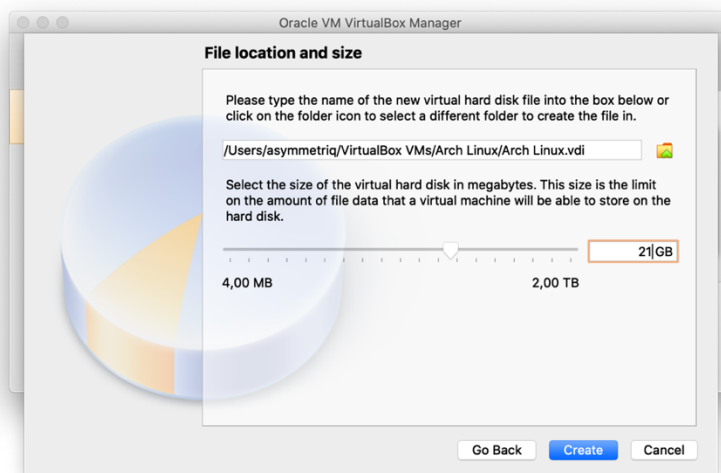


Рисунок 3. Задаём размер виртуального диска

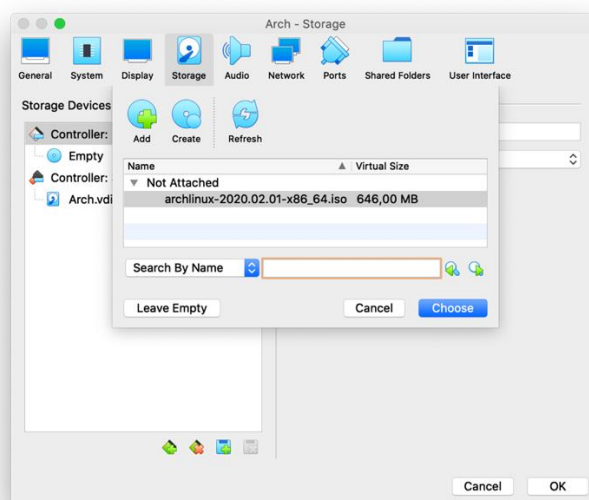
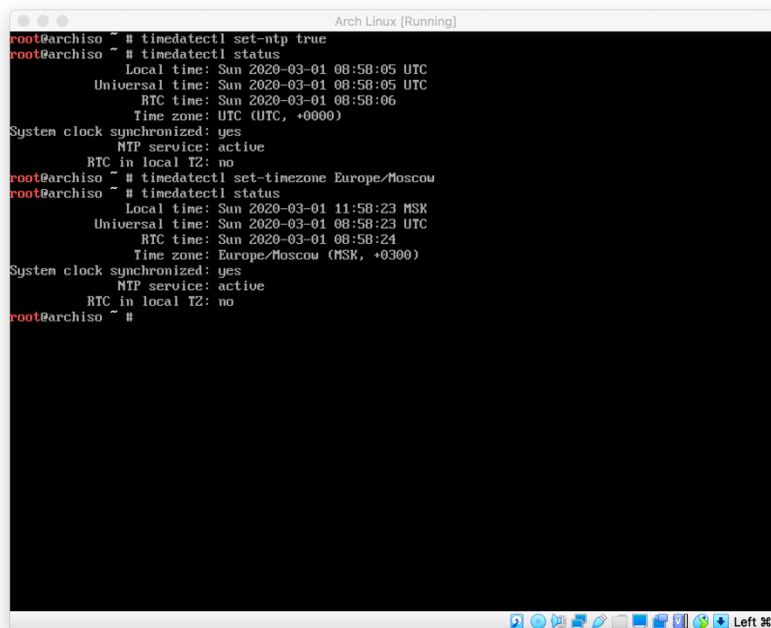


Рисунок 4. Подключаем iso-образ

2. УСТАНОВКА

Запускаем виртуальную машину и начинаем установку:



```
root@archiso ~ # timedatectl set-ntp true
root@archiso ~ # timedatectl status
          Local time: Sun 2020-03-01 08:58:05 UTC
          Universal time: Sun 2020-03-01 08:58:05 UTC
            RTC time: Sun 2020-03-01 08:58:06
            Time zone: UTC (UTC, +0000)
System clock synchronized: yes
            NTP service: active
          RTC in local TZ: no
root@archiso ~ # timedatectl set-timezone Europe/Moscow
root@archiso ~ # timedatectl status
          Local time: Sun 2020-03-01 11:58:23 MSK
          Universal time: Sun 2020-03-01 08:58:23 UTC
            RTC time: Sun 2020-03-01 08:58:24
            Time zone: Europe/Moscow (MSK, +0300)
System clock synchronized: yes
            NTP service: active
          RTC in local TZ: no
root@archiso ~ #
```

Рисунок 5. Синхронизация системных часов

```
# timedatectl set-ntp true - запуск службы синхронизации
# timedatectl status - проверка текущего состояния
# timedatectl set-timezone Europe/Moscow - выбор часового пояса
```



Рисунок 6. Разметка диска

```
# cfdisk - запуск утилиты разметки
```

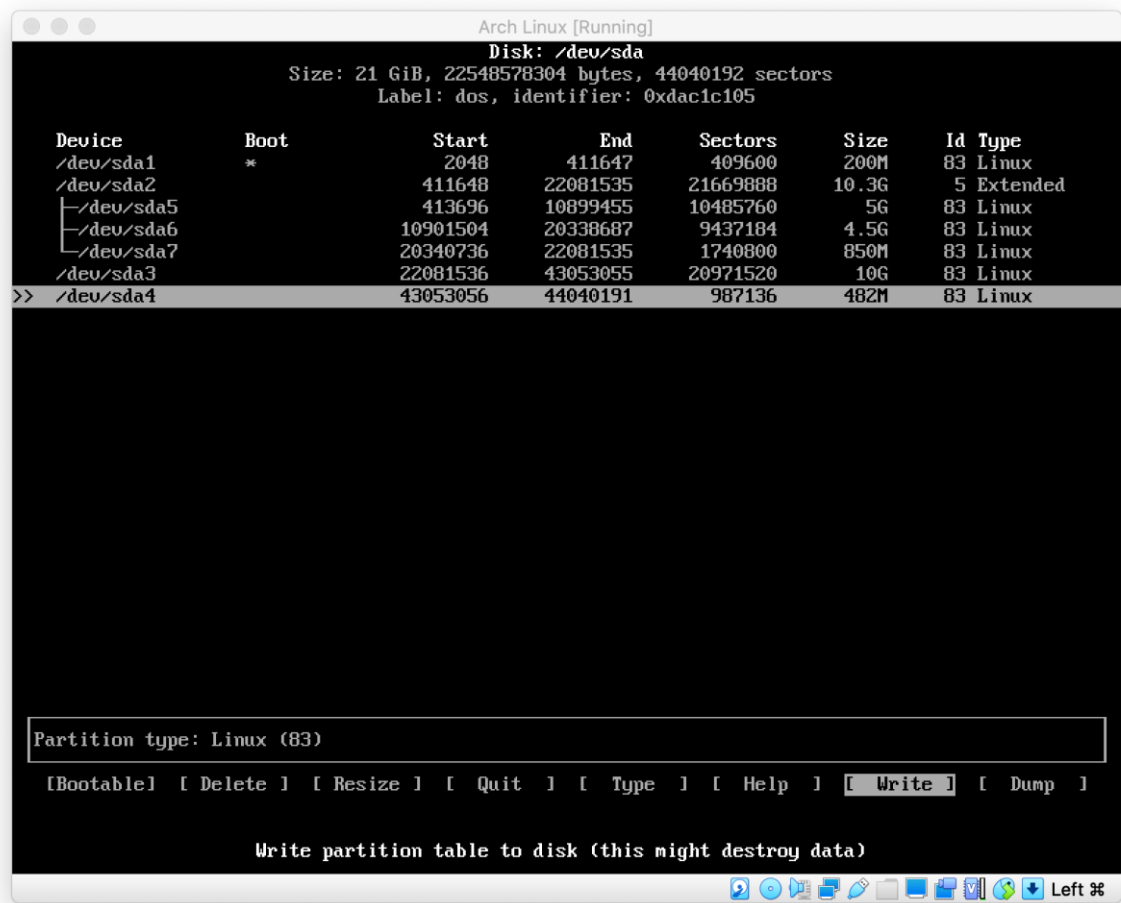


Рисунок 7. Результат разметки

sda1 - раздел под boot
sda2 - extended-раздел
sda3 - раздел под root
sda4 - раздел под tmp
sda5 - раздел под opt
sda6 - раздел под var
sda7 - раздел под home

```
Arch Linux [Running]
root@archiso ~ # mkfs.ext4 /dev/sda6
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 1179648 4k blocks and 294912 inodes
Filesystem UUID: 8fef6dca-2f1b-4904-823c-0308dfe83696
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done

root@archiso ~ # mkfs.ext4 /dev/sda7
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 217600 4k blocks and 54432 inodes
Filesystem UUID: 190fc331-9bd3-43b4-b8b4-3ebb842e3ec4
Superblock backups stored on blocks:
    32768, 98304, 163840

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done

root@archiso ~ # mkfs.ext4 /dev/sda4
mke2fs 1.45.5 (07-Jan-2020)
Creating filesystem with 493568 1k blocks and 123464 inodes
Filesystem UUID: 9c787226-d4f2-4c29-ab0f-50d0e62b499e
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729, 204801, 221185, 401409

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done

root@archiso ~ # _
```

Рисунок 8. Форматируем разделы

```
# mkfs.ext4 /dev/sdX1 - форматируем раздел в файловую систему etx4
```

```
root@archiso ~ # mount /dev/sda3 /mnt
root@archiso ~ # mkdir /mnt/boot
root@archiso ~ # mkdir /mnt/home
root@archiso ~ # mkdir /mnt/var
root@archiso ~ # mkdir /mnt/opt
root@archiso ~ # mkdir /mnt/tmp
root@archiso ~ # mount /dev/sda1 /mnt/boot
root@archiso ~ # mount /dev/sda5 /mnt/opt
root@archiso ~ # mount /dev/sda6 /mnt/var
root@archiso ~ # mount /dev/sda7 /mnt/home
root@archiso ~ # mount /dev/sda4 /mnt/tmp
root@archiso ~ #
```

Рисунок 9. Монтируем файловую систему

```
# mount /dev/sda3 /mnt - монтируем корень (так же монтируем другие разделы)
# mkdir /mnt/directory - создаём директорию
```

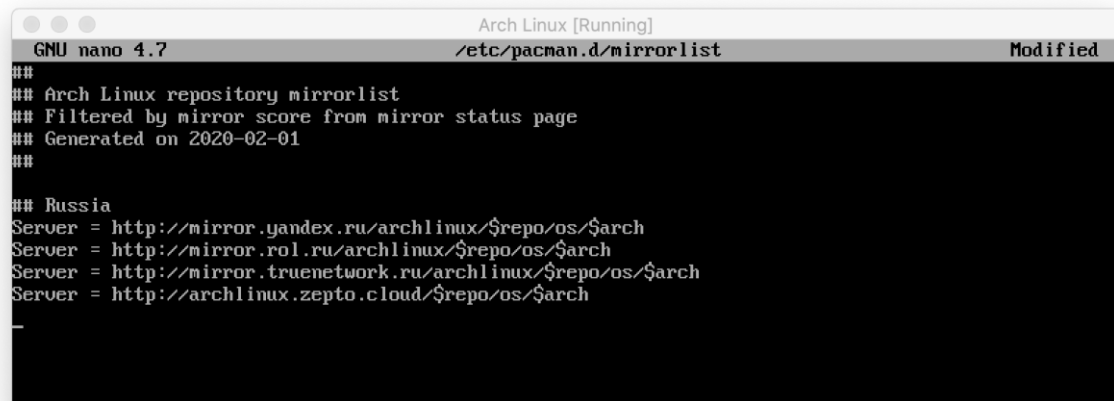


Рисунок 10. Выбираем зеркала загрузки, лишние удаляем

```
# nano /etc/pacman.d/mirrorlist - читаем и редактируем список зеркал
```

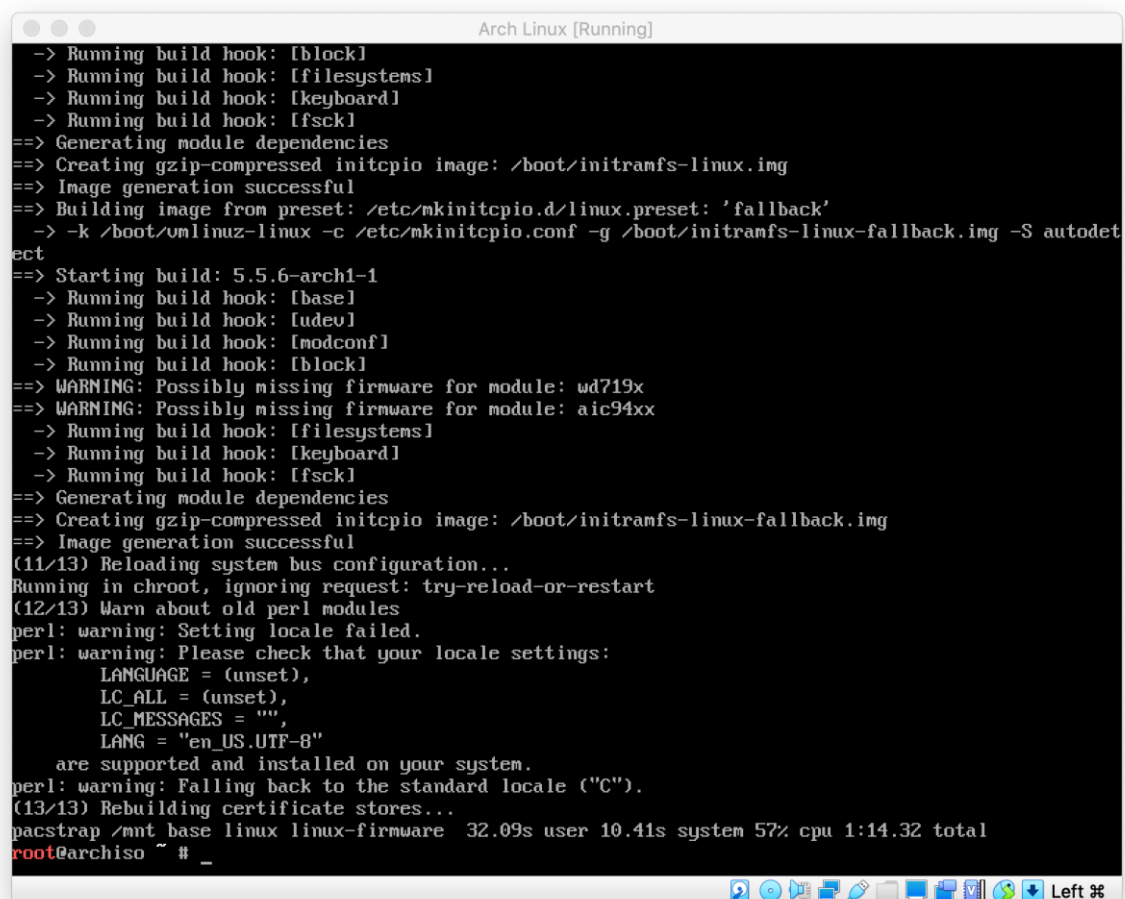


Рисунок 11. Установка основных пакетов

```
#pacstrap /mnt base linux linux-firmware - скрипт установки ядра и прошивки
```



```
root@archiso ~ # genfstab -U /mnt >> /mnt/etc/fstab
root@archiso ~ #
```

Рисунок 12. Генерируем таблицу файловой системы

```
Arch Linux [Running]
root@archiso ~ # arch-chroot /mnt
zsh: no such file or directory: arch-chroot /mnt
127 root@archiso ~ # arch-chroot /mnt
[root@archiso /]# ln -sf /usr/share/zoneinfo/Europe/Moscow /etc/localtime
[root@archiso /]# hwclock --systohc
[root@archiso /]# locale-gen
Generating locales...
Generation complete.
[root@archiso /]# _
```

Рисунок 13. Переходим к корневому каталогу новой системы

```
# arch-chroot /mnt - переходим к корневому каталогу новой системы
# ln -sf /usr/share/zoneinfo/Europe/Moscow /etc/localtime - задаём час.пояс
# hwclock -systohc - установка аппаратных часов в соответствии с сист. врем
# locale-gen - предварительно раскомментировав нужные локали, генерируем их
```

```
Arch Linux [Running]
GNU nano 4.8 /etc/locale.conf Modified
LANG=en_US.UTF-8
```

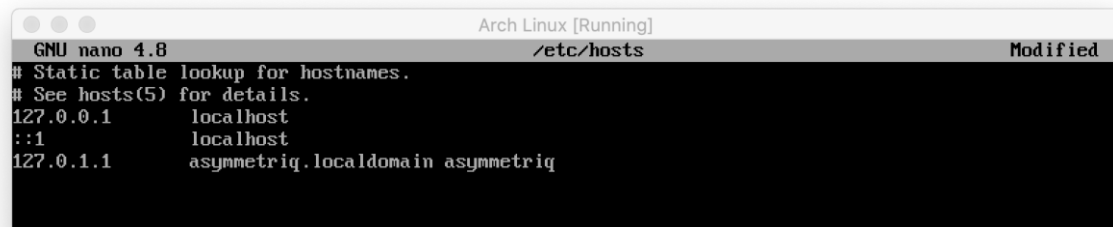
Рисунок 14. Создаём locale.conf с нужным языком

```
# nano /etc/locale.conf - создаём locale.conf
```

```
Arch Linux [Running]
GNU nano 4.8 /etc/hostname Modified
asymmetriq
```

Рисунок 15. Задаём имя хоста

```
# nano /etc/hostname - задаём имя хоста
```



```
Arch Linux [Running]
GNU nano 4.8 /etc/hosts Modified
# Static table lookup for hostnames.
# See hosts(5) for details.
127.0.0.1    localhost
::1         localhost
127.0.1.1    asymmetriq.localdomain asymmetriq
```

Рисунок 16. Задаём локальный хост

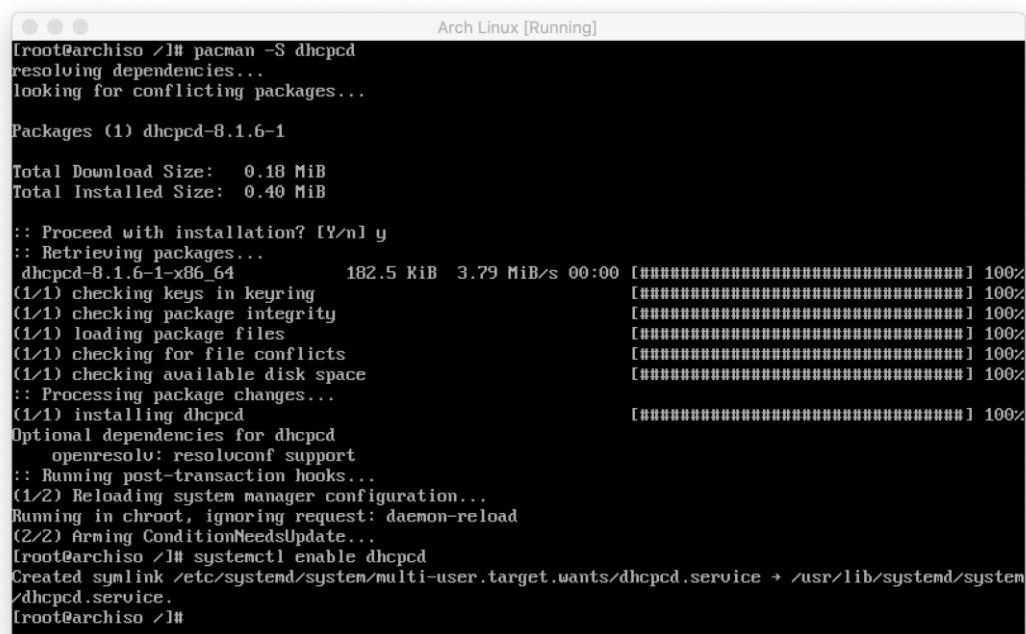
```
# nano /etc/hosts - вписываем локальный хост
```



```
Arch Linux [Running]
[root@archiso /]# passwd
New password:
Retype new password:
passwd: password updated successfully
[root@archiso /]#
```

Рисунок 17. Устанавливаем пароль для root

```
# passwd - устанавливаем пароль для root
```



```
Arch Linux [Running]
[root@archiso /]# pacman -S dhcpd
resolving dependencies...
looking for conflicting packages...

Packages (1) dhcpd-8.1.6-1

Total Download Size: 0.18 MiB
Total Installed Size: 0.40 MiB

:: Proceed with installation? [Y/n] y
:: Retrieving packages...
  dhcpd-8.1.6-1-x86_64           182.5 KiB   3.79 MiB/s   00:00 [#####] 100%
(1/1) checking keys in keyring [#####] 100%
(1/1) checking package integrity [#####] 100%
(1/1) loading package files [#####] 100%
(1/1) checking for file conflicts [#####] 100%
(1/1) checking available disk space [#####] 100%
:: Processing package changes...
(1/1) installing dhcpd [#####] 100%
Optional dependencies for dhcpd
  openresolv: resolvconf support
:: Running post-transaction hooks...
(1/2) Reloading system manager configuration...
Running in chroot, ignoring request: daemon-reload
(2/2) Arming ConditionNeedsUpdate...
[root@archiso /]# systemctl enable dhcpd
Created symlink /etc/systemd/system/multi-user.target.wants/dhcpd.service → /usr/lib/systemd/systemd/systemd/dhcpd.service.
[root@archiso /]#
```

Рисунок 18. Загрузка пакета dhcpd

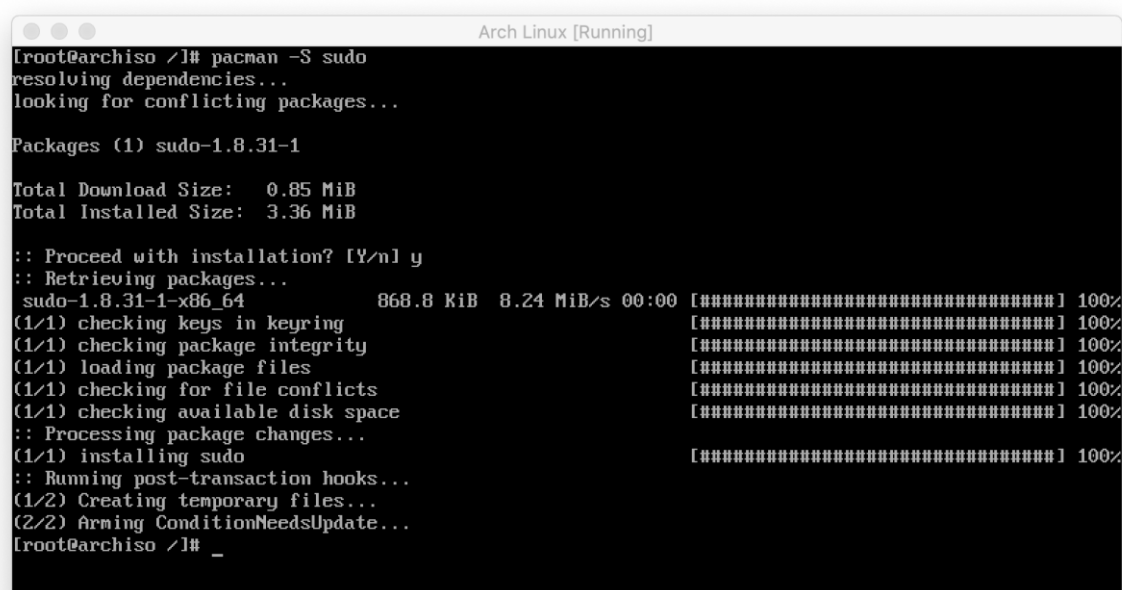
```
# pacman -S dhcpd - загрузка пакета dhcpd
```

A terminal window titled "Arch Linux [Running]" showing the process of creating a user named "artem". The commands executed are "useradd -m artem" and "passwd artem". The output shows the password being set and confirmed successfully.

```
[root@archiso /]# useradd -m artem
[root@archiso /]# passwd artem
New password:
Retype new password:
passwd: password updated successfully
[root@archiso /]# _
```

Рисунок 19. Создаём пользователя и ставим пароль

```
# useradd -m artem - создаём пользователя artem
# passwd artem - устанавливаем пароль для artem
```

A terminal window titled "Arch Linux [Running]" showing the installation of the "sudo" package using "pacman". The output displays the package details, download size, and progress bars for various steps like checking keys, integrity, and installing the package.

```
[root@archiso /]# pacman -S sudo
resolving dependencies...
looking for conflicting packages...

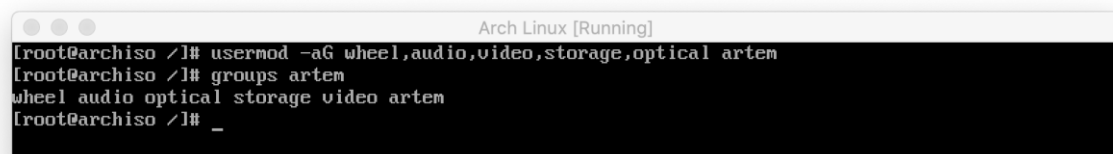
Packages (1) sudo-1.8.31-1

Total Download Size:  0.85 MiB
Total Installed Size: 3.36 MiB

:: Proceed with installation? [Y/n] y
:: Retrieving packages...
 sudo-1.8.31-1-x86_64           868.8 KiB  8.24 MiB/s 00:00 [#####] 100%
(1/1) checking keys in keyring [#####] 100%
(1/1) checking package integrity [#####] 100%
(1/1) loading package files     [#####] 100%
(1/1) checking for file conflicts [#####] 100%
(1/1) checking available disk space [#####] 100%
:: Processing package changes...
(1/1) installing sudo          [#####] 100%
:: Running post-transaction hooks...
(1/2) Creating temporary files...
(2/2) Arming ConditionNeedsUpdate...
[root@archiso /]# _
```

Рисунок 20. Загрузка пакета sudo

```
# pacman -S sudo - загрузка пакета sudo
```

A terminal window titled "Arch Linux [Running]" showing the assignment of groups to the user "artem". The command "usermod -aG wheel, audio, video, storage, optical artem" is executed, followed by "groups artem" which shows the resulting group memberships.

```
[root@archiso /]# usermod -aG wheel, audio, video, storage, optical artem
[root@archiso /]# groups artem
wheel audio optical storage video artem
[root@archiso /]# _
```

Рисунок 21. Настраиваем группы доступа

```
# usermod -aG wheel, audio, video, storage, optical artem - настраиваем доступ
```

```
Arch Linux [Running]
## this may allow users to subvert the command being run via sudo.
## Defaults env_keep += "XMODIFIERS GTK_IM_MODULE QT_IM_MODULE QT_IM_SWITCHER"
##
## Uncomment to use a hard-coded PATH instead of the user's to find commands
## Defaults secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin"
##
## Uncomment to send mail if the user does not enter the correct password.
## Defaults mail_badpass
##
## Uncomment to enable logging of a command's output, except for
## sudoreplay and reboot. Use sudoreplay to play back logged sessions.
## Defaults log_output
## Defaults!usr/bin/sudoreplay !log_output
## Defaults!usr/local/bin/sudoreplay !log_output
## Defaults!REBOOT !log_output
##
## Runas alias specification
##
##
## User privilege specification
##
root ALL=(ALL) ALL

## Uncomment to allow members of group wheel to execute any command
%wheel ALL=(ALL) ALL

## Same thing without a password
# %wheel ALL=(ALL) NOPASSWD: ALL

## Uncomment to allow members of group sudo to execute any command
# %sudo ALL=(ALL) ALL

## Uncomment to allow any user to run sudo if they know the password
## of the user they are running the command as (root by default).

82,1 90%
```

Рисунок 22. Разрешаем юзерам в wheel пользоваться sudo

```
Arch Linux [Running]
[root@archiso /]# pacman -S grub
resolving dependencies...
looking for conflicting packages...

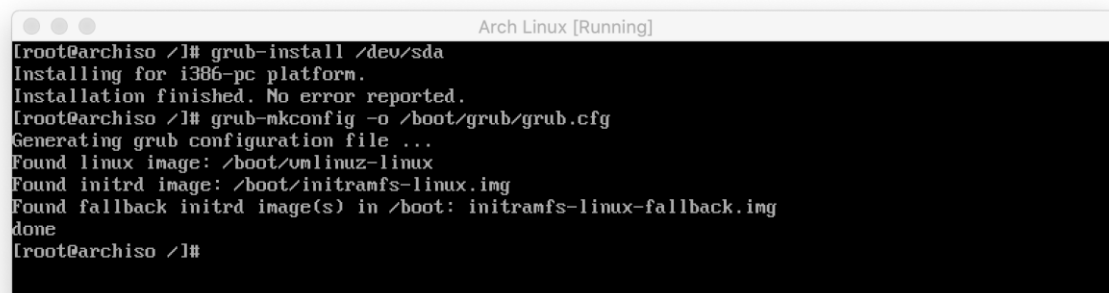
Packages (1) grub-2:2.04-5

Total Download Size:   6.68 MiB
Total Installed Size: 32.53 MiB

:: Proceed with installation? [Y/n] y
:: Retrieving packages...
grub-2:2.04-5-x86_64           6.7 MiB  8.39 MiB/s 00:01 [#####] 100%
(1/1) checking keys in keyring [#####] 100%
(1/1) checking package integrity [#####] 100%
(1/1) loading package files [#####] 100%
(1/1) checking for file conflicts [#####] 100%
(1/1) checking available disk space [#####] 100%
:: Processing package changes...
(1/1) installing grub [#####] 100%
Generate your bootloader configuration with:
grub-mkconfig -o /boot/grub/grub.cfg
Optional dependencies for grub
  freetype2: For grub-mkfont usage
  fuse2: For grub-mount usage
  dosfstools: For grub-mkrescue FAT FS and EFI support
  efibootmgr: For grub-install EFI support
  libisoburn: Provides xorriso for generating grub rescue iso using grub-mkrescue
  os-prober: To detect other OSes when generating grub.cfg in BIOS systems
  nttools: For grub-mkrescue FAT FS support
:: Running post-transaction hooks...
(1/1) Arming ConditionNeedsUpdate...
[root@archiso /]# _
```

Рисунок 23. Установка grub

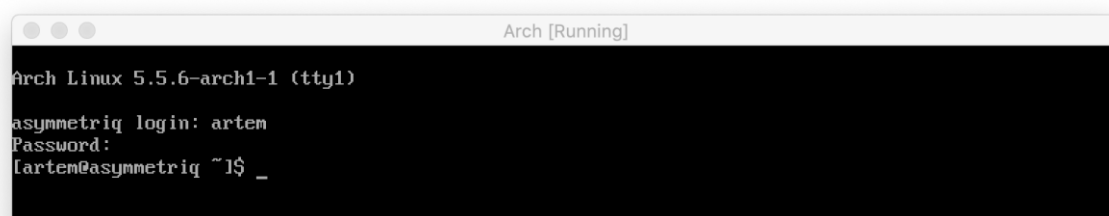
```
# pacman -S grub - загрузка бутлоадера grub
```

A terminal window titled "Arch Linux [Running]" showing the installation of GRUB. The user runs `grub-install /dev/sda`, which installs for the i386-pc platform. Then, `grub-mkconfig -o /boot/grub/grub.cfg` is run to generate the configuration file. The terminal shows it found the Linux image at `/boot/vmlinuz-linux` and the initrd image at `/boot/initramfs-linux.img`, with a fallback at `/boot: initramfs-linux-fallback.img`.

```
[root@archiso /]# grub-install /dev/sda
Installing for i386-pc platform.
Installation finished. No error reported.
[root@archiso /]# grub-mkconfig -o /boot/grub/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-linux
Found initrd image: /boot/initramfs-linux.img
Found fallback initrd image(s) in /boot: initramfs-linux-fallback.img
done
[root@archiso /]#
```

Рисунок 24. Установка *grub* для диска и создание кофига

```
# grub-install /dev/sda - установка grub для диска
# grub-mkconfig -o /boot/grub/grub.cfg
```

A terminal window titled "Arch [Running]" showing the login process. The prompt is `Arch Linux 5.5.6-arch1-1 (tty1)`. The user enters `asymmetriq` as the login name and `artem` as the password. The prompt changes to `[artem@asymmetriq ~]#`.

```
Arch Linux 5.5.6-arch1-1 (tty1)
asymmetriq login: artem
Password:
[artem@asymmetriq ~]#
```

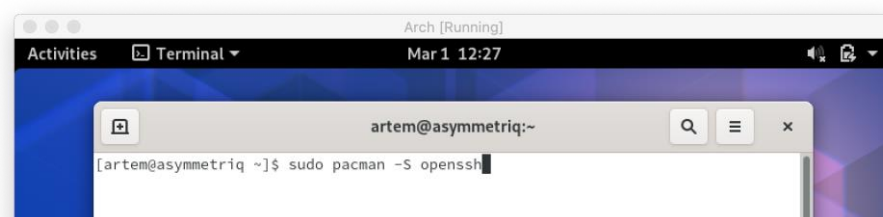
Рисунок 25. Перезагружаемся и входим в учётную запись

3. НАСТРОЙКА СИСТЕМЫ

Настроим графическую оболочку:

```
# pacman -S xorg xorg-server gnome gnome-extra gdm - установка GNOME
# systemctl enable gdm - включение Gnome Display Manager
```

Установим SSH-сервер:

A terminal window titled "Arch [Running]" showing the installation of OpenSSH. The user runs `sudo pacman -S openssh`. The terminal shows the command being executed. The window has a title bar with "Activities", "Terminal", and "Mar 1 12:27".

```
Arch [Running]
Activities Terminal Mar 1 12:27
[artem@asymmetriq ~]$ sudo pacman -S openssh
```

Рисунок 26. Устанавливаем *OpenSSH*

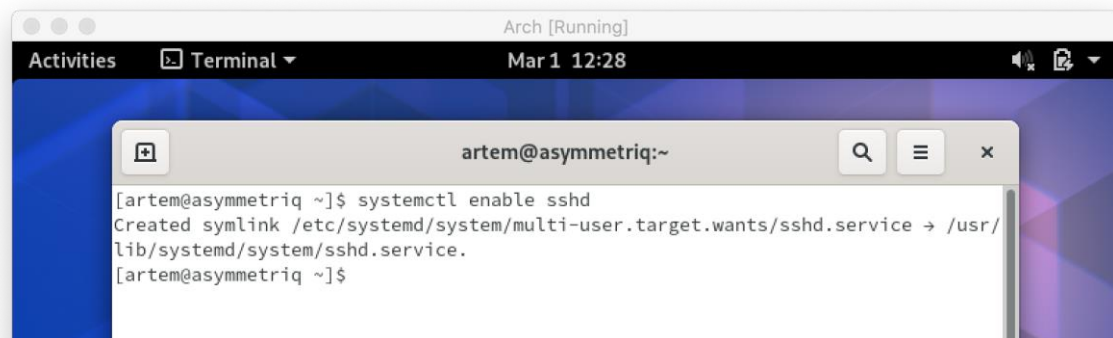


Рисунок 27. Включаем автозагрузку сервера при старте системы

```
# systemctl enable sshd - включаем автозагрузку
```

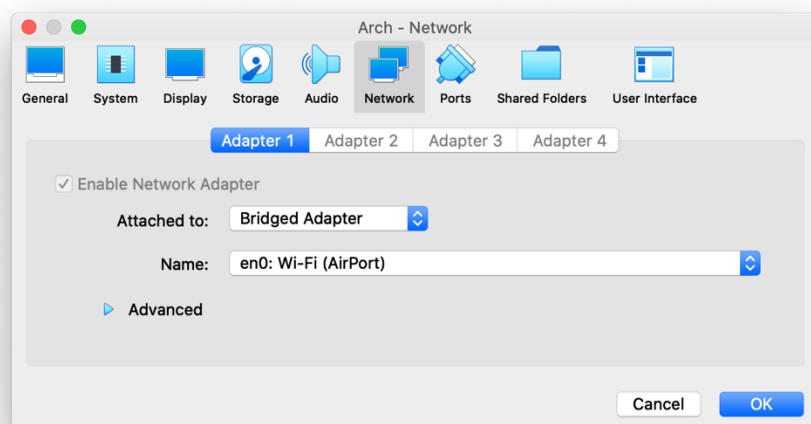


Рисунок 28. Настраиваем сетевой мост

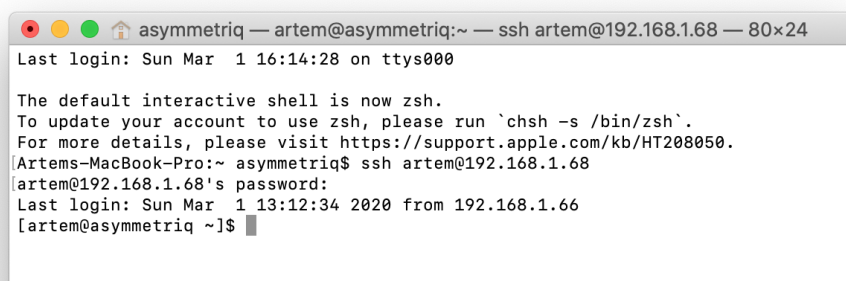
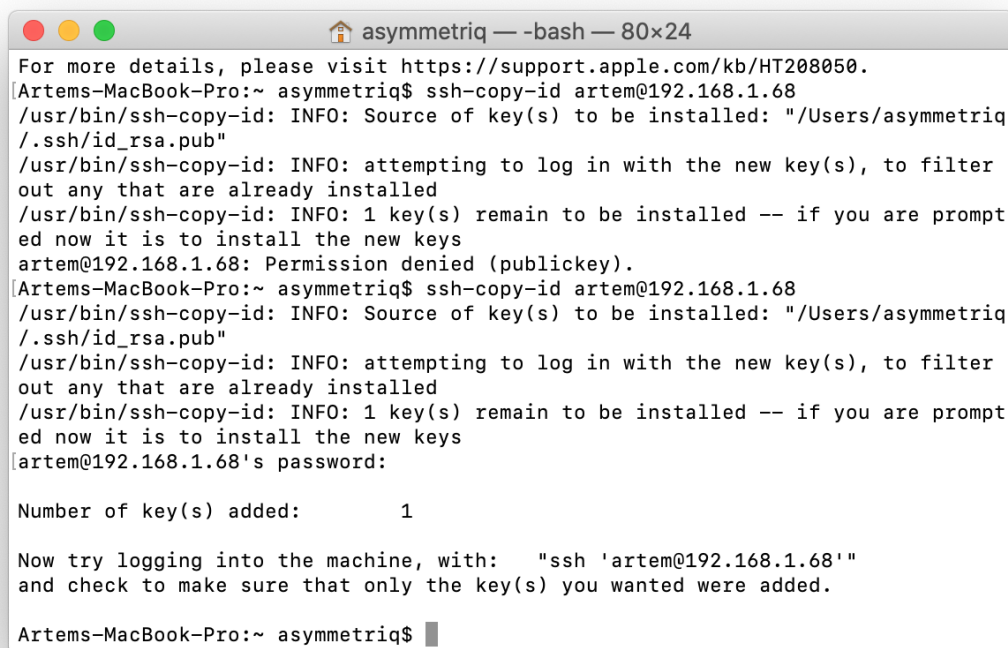


Рисунок 29. Проверяем подключение



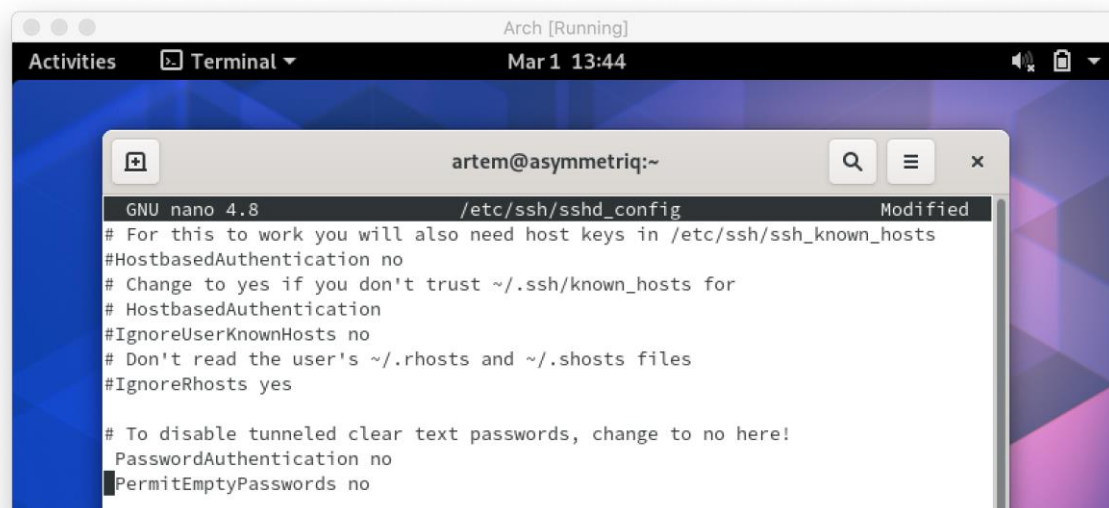
```
asymmetriq — -bash — 80x24
For more details, please visit https://support.apple.com/kb/HT208050.
[Artems-MacBook-Pro:~ asymmetriq$ ssh-copy-id artem@192.168.1.68 ]
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/Users/asymmetriq/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
artem@192.168.1.68: Permission denied (publickey).
[Artems-MacBook-Pro:~ asymmetriq$ ssh-copy-id artem@192.168.1.68 ]
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/Users/asymmetriq/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
[artem@192.168.1.68's password: ]

Number of key(s) added:      1

Now try logging into the machine, with:  "ssh 'artem@192.168.1.68'"
and check to make sure that only the key(s) you wanted were added.

Artems-MacBook-Pro:~ asymmetriq$
```

Рисунок 30. Генерируем и копируем публичный ключ



```
Arch [Running]
Activities Terminal Mar 1 13:44
artem@asymmetriq:~
GNU nano 4.8 /etc/ssh/sshd_config Modified
# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication no
PermitEmptyPasswords no
```

Рисунок 31. Запрещаем доступ по паролю

```
# sudo nano /etc/ssh/sshd_config - открываем конфиг
```


Настроим приложения:

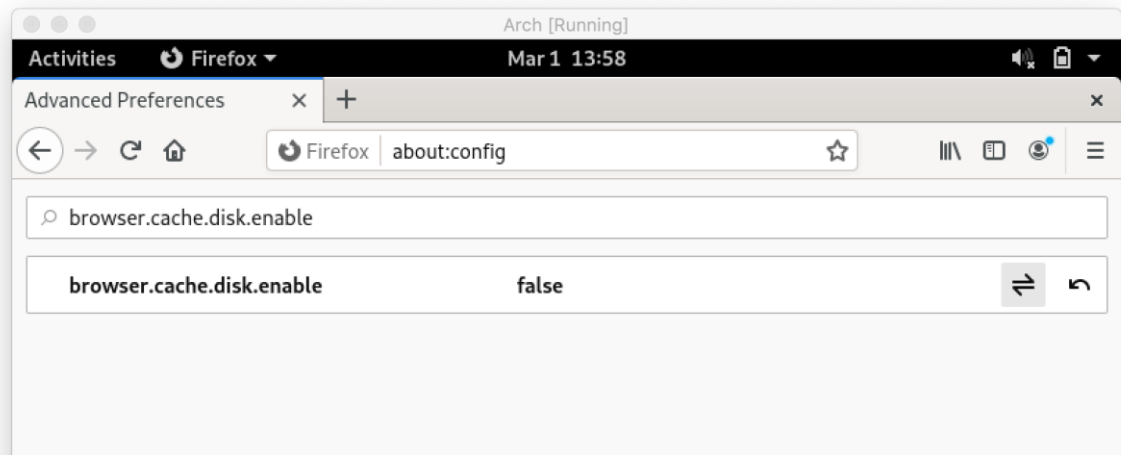


Рисунок 32. Отключаем кэш на диске браузера Firefox (только O3Y)

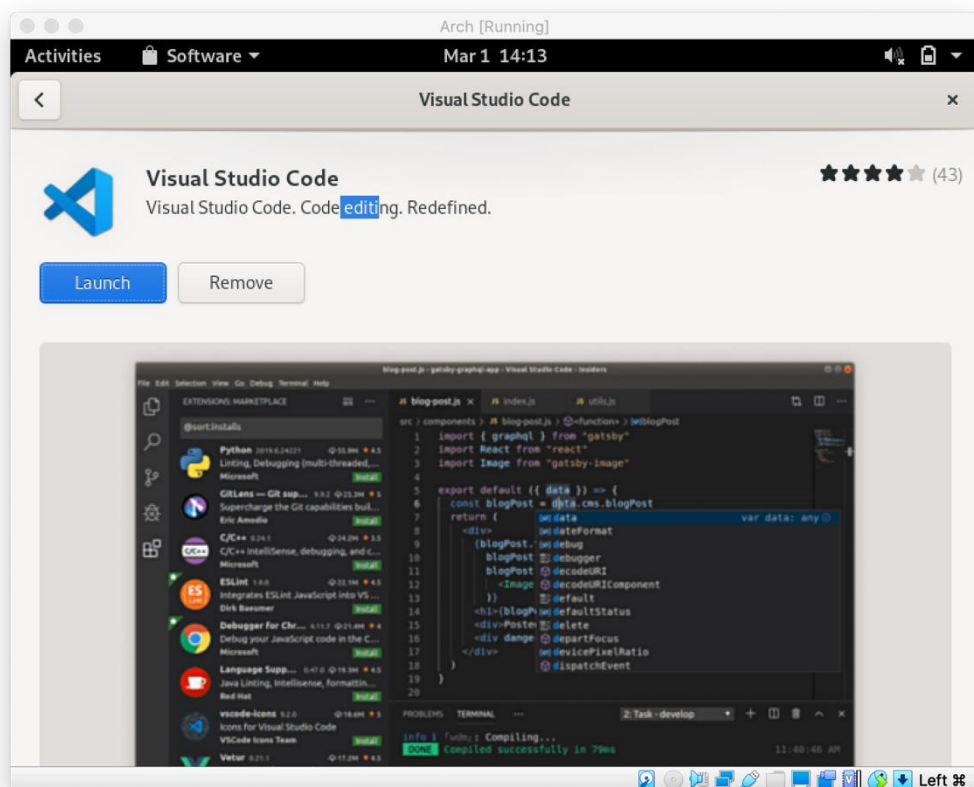


Рисунок 33. Устанавливаем VS Code

ПЕРЕЧЕНЬ УСТАНОВЛЕННОГО ПО

1. Arch Linux 2020.02.01
2. GNOME Desktop Environment
3. Visual Studio Code
4. OpenSSH
5. Mozilla Firefox