

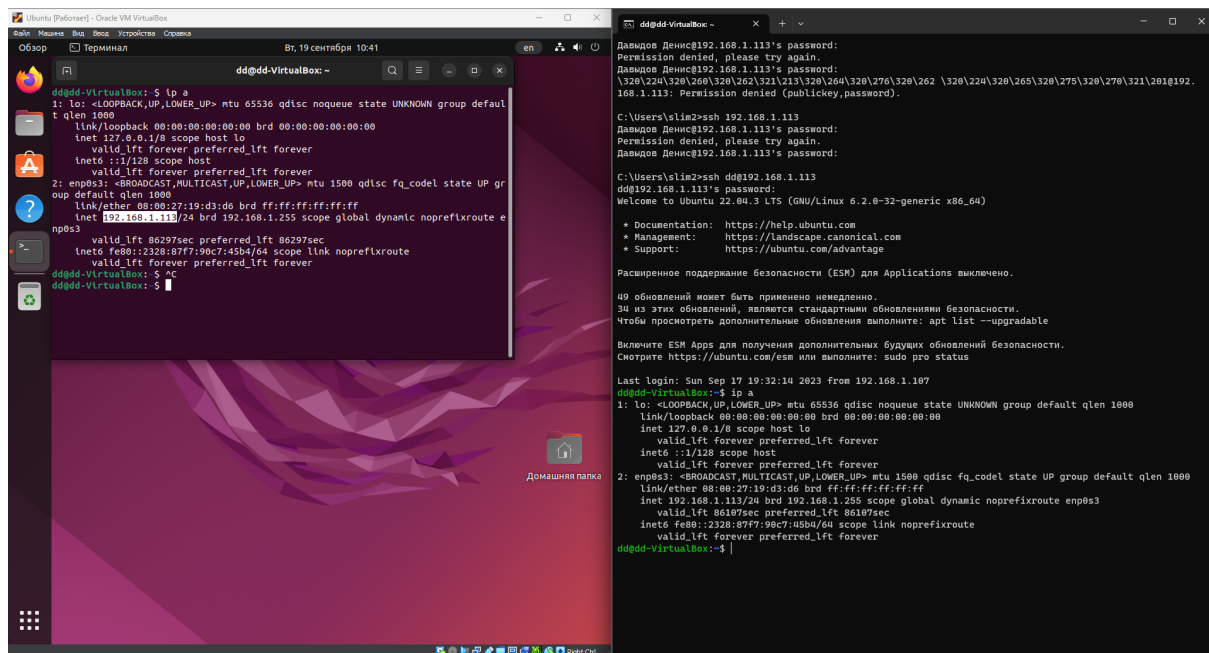
## Урок 1. Установка ОС Ubuntu в виртуальной машине. Работа в SSH-клиенте

### Условие

- Установить Ubuntu Desktop 22.04 в виртуальную машину.
- Установить гостевые дополнения ОС.
- Установить Midnight Commander.
- Установить SSH-соединение с виртуальной машиной из хостовой.

### Результат

Скриншот SSH-подключения к установленной системе с запущенным mc.



## Урок 2. Работа с файлами и ссылками

### Задание

- Используя команду `cat`, создать два файла с данными, а затем объединить их.
- Просмотреть содержимое созданного файла.
- Переименовать файл, дав ему новое имя.
- Создать несколько файлов.
- Создать директорию, переместить файл туда.
- Удалить все созданные в этом и предыдущем задании директории и файлы.
- Создать файл `file1` и наполнить его произвольным содержимым.
- Скопировать его в `file2`.
- Создать символическую ссылку `file3` на `file1`.
- Создать жёсткую ссылку `file4` на `file1`.
- Посмотреть, какие айтоды у файлов.
- Удалить `file1`.
- Что стало с остальными созданными файлами?
- Попробовать вывести их на экран.

- Дать созданным файлам другие, произвольные имена.
- Создать новую символическую ссылку.  
Переместить ссылки в другую директорию.

### **Результат**

Текст команд, которые применялись при выполнении задания.

Присылаем в формате текстового документа: задание и команды для решения (без вывода).

Формат — PDF (один файл на все задания).

1. Используя команду `cat`, создать два файла с данными, а затем объединить их.

Просмотреть содержимое созданного файла.

Переименовать файл, дав ему новое имя.

```
dd: /home/dd
> pwd
/home/dd
dd: /home/dd
> mkdir hw2
dd: /home/dd
> cd hw2
dd: /home/dd/hw2
> cat > file1
one file
dd: /home/dd/hw2
> cat > file2
two filedd: /home/dd/hw2
> cat file1 file2 > file3
dd: /home/dd/hw2
> cat file3
one file
two file
dd: /home/dd/hw2
> mv file3 resultFile
```

2. Создать несколько файлов.

Создать директорию, переместить файл туда.

Удалить все созданные в этом и предыдущем задании директории и файлы.

```
dd: /home/dd/hw2
> mkdir tz2
dd: /home/dd/hw2
> mv resultFile tz2
dd: /home/dd
> rm hw2 -r
dd: /home/dd
>
```

3. Создать файл `file1` и наполнить его произвольным содержимым.

Скопировать его в `file2`.

Создать символическую ссылку file3 на file1.  
Создать жёсткую ссылку file4 на file1.  
Посмотреть, какие айноды у файлов.  
Удалить file1.  
Что стало с остальными созданными файлами?  
Попробовать вывести их на экран.

```
dd: /home/dd
> mkdir hw2
dd: /home/dd
> cd hw2/
dd: /home/dd/hw2
> cat > file1
какой-то текст
dd: /home/dd/hw2
> cp file1 file2
dd: /home/dd/hw2
> ln -s file1 file3
dd: /home/dd/hw2
> ll
итого 12K
655999 -rw-rw-r-- 1 dd dd 26 сен 25 12:36 file1
655951 -rw-rw-r-- 1 dd dd 26 сен 25 12:36 file2
656014 lrwxrwxrwx 1 dd dd  5 сен 25 12:38 file3 -> file1
dd: /home/dd/hw2
> ln -P file1 file4
dd: /home/dd/hw2
> ll
итого 16K
655780 -rw-rw-r-- 3 dd dd 27 сен 25 11:05 file1
655925 -rw-rw-r-- 1 dd dd 27 сен 25 11:06 file2
655780 -rw-rw-r-- 3 dd dd 27 сен 25 11:05 file3
655780 -rw-rw-r-- 3 dd dd 27 сен 25 11:05 file4
dd: /home/dd/hw2
> rm file1
dd: /home/dd/hw2
> ll
итого 12K
655999 -rw-rw-r-- 2 dd dd 26 сен 25 12:36 file1
655951 -rw-rw-r-- 1 dd dd 26 сен 25 12:36 file2
656014 lrwxrwxrwx 1 dd dd  5 сен 25 12:38 file3 -> file1
655999 -rw-rw-r-- 2 dd dd 26 сен 25 12:36 file4
dd: /home/dd/hw2
> cat file1 file2 file3 file4
какой-то тексткакой-то тексткакой-то тексткакой-то текст
```

4. Дать созданным файлам другие, произвольные имена.  
Создать новую символическую ссылку.  
Переместить ссылки в другую директорию.

```
> mv file1 one_file
```

```

dd: /home/dd/hw2
> mv file2 two_file
dd: /home/dd/hw2
> ln -s file_one file_one_link
dd: /home/dd/hw2
> mkdir newFolder
dd: /home/dd
> mv hw2/* /home/dd/newFolder/
dd: /home/dd
> cd newFolder/
dd: /home/dd/newFolder
> ll
итого 12K
656014 lrwxrwxrwx 1 dd dd 5 сен 25 12:38 file3 -> file1
655999 -rw-rw-r-- 2 dd dd 26 сен 25 12:36 file4
655993 lrwxrwxrwx 1 dd dd 8 сен 25 12:42 file_one_link -> file_one
655999 -rw-rw-r-- 2 dd dd 26 сен 25 12:36 one_file
655951 -rw-rw-r-- 1 dd dd 26 сен 25 12:36 two_file
dd: /home/dd/newFolder
> tree
.
├── file3 -> file1
├── file4
├── file_one_link -> file_one
├── one_file
└── two_file

0 directories, 5 files

```

## Урок 3. Права доступа и безопасность

### Задание

- Создать два произвольных файла.

Первому присвоить права на чтение и запись для владельца и группы, только на чтение — для всех.

Второму присвоить права на чтение и запись только для владельца. Сделать это в численном и символьном виде.

Назначить новых владельца и группу для директории целиком.

### Управление пользователями:

- \* создать пользователя, используя утилиту `useradd` и `adduser`;
- \* удалить пользователя, используя утилиту `userdel`.

### Управление группами:

- создать группу с использованием утилит `groupadd` и `addgroup`;
- попрактиковаться в смене групп у пользователей;
- добавить пользователя в группу, не меняя основной;
- Создать пользователя с правами суперпользователя. Сделать так, чтобы `sudo` не требовал пароль для выполнения команд.

### Дополнительные (необязательные) задания

- Используя дополнительные материалы, выдать одному из созданных пользователей право на выполнение ряда команд, требующих прав суперпользователя (команды выбираем на своё усмотрение).
- Создать группу developer и нескольких пользователей, входящих в неё.  
Создать директорию для совместной работы.  
Сделать так, чтобы созданные одними пользователями файлы могли изменять другие пользователи этой группы.
- Создать в директории для совместной работы поддиректорию для обмена файлами, но чтобы удалять файлы могли только их создатели.
- Создать директорию, в которой есть несколько файлов.  
Сделать так, чтобы открыть файлы можно было, только зная имя файла, а через ls список файлов посмотреть было нельзя.

```
dd: /home/dd
> mkdir hw3
dd: /home/dd
> cd hw3
dd: /home/dd/hw3
> cat > file1
one file
dd: /home/dd/hw3
> cat > file2
two file
dd: /home/dd/hw3
> chmod 664 file1
dd: /home/dd/hw3
> chmod 600 file2
dd: /home/dd/hw3
> chmod ug=rw,o=r file1
dd: /home/dd/hw3
> chmod u+rw,go-rwx file2
dd: /home/dd/hw3
> ll
итого 8,0K
655962 -rw-rw-r-- 1 dd dd 9 сен 28 16:10 file1
655964 -rw----- 1 dd dd 9 сен 28 16:11 file2
dd: /home/dd/hw3
>
```

```
sudo visudo
test localhost=ALL
dd: /home/dd/hw3
> sudo groupadd developer
dd: /home/dd/hw3
> sudo useradd -m -G developer -s /bin/bash dev_1
dd: /home/dd/hw3
> sudo useradd -m -G developer -s /bin/bash dev_2
```

```
dd: /home/dd/hw3
> mkdir developer
dd: /home/dd/hw3
> sudo chgrp developer ~/hw3/developer/
dd: /home/dd/hw3
> chmod 775 ~/hw3/developer/
dd: /home/dd/hw3
> sudo chmod g+s ~/hw3/developer/
dd: /home/dd/hw3
> ll
итого 12K
656032 drwxrwsr-x 2 dd developer 4,0K сен 28 17:03 developer/
655962 -rw-rw-r-- 1 dd dd      9 сен 28 16:10 file1
655964 -rw----- 1 dd dd      9 сен 28 16:11 file2
dd: /home/dd/hw3
>
```

## Урок 4. Подключение сторонних репозиториев, ручная установка пакетов

### Задание

- Подключить дополнительный репозиторий на выбор: Docker, Nginx, Oracle MySQL. Установить любой пакет из этого репозитория.
- Установить и удалить deb-пакет с помощью dpkg.
- Установить и удалить snap-пакет.
- Добавить задачу для выполнения каждые 3 минуты (создание директории, запись в файл).
- \* Подключить PPA-репозиторий на выбор. Установить из него пакет. Удалить PPA из системы.

- \* Создать задачу резервного копирования (tar) домашнего каталога пользователя. Реализовать с использованием пользовательских crontab-файлов
1. Подключить дополнительный репозиторий на выбор: Docker, Nginx, Oracle MySQL. Установить любой пакет из этого репозитория.

```
sudo apt install nginx
sudo apt remove nginx
sudo apt purge nginx
```

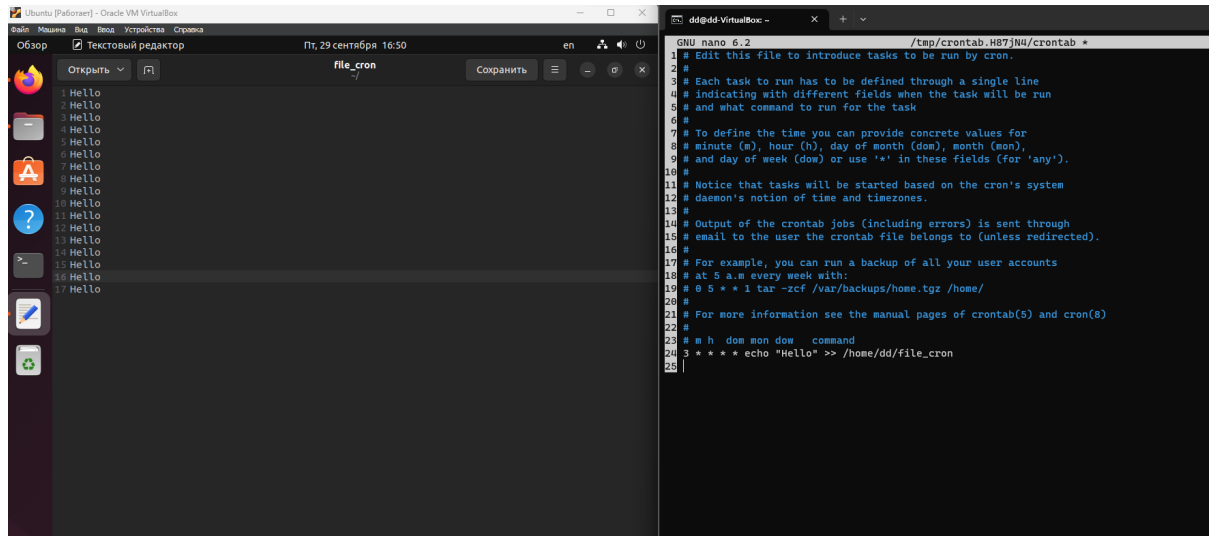
2. Установить и удалить deb-пакет с помощью dpkg.

```
sudo apt install ~/Загрузки/google-chrome-stable_current_amd64.deb -y
sudo dpkg -r google-chrome-stable
sudo apt autoremove
```
3. Установить и удалить snap-пакет.

```
sudo snap install telegram-desktop
sudo snap remove telegram-desktop
```

4. Добавить задачу для выполнения каждые 3 минуты (создание директории, запись в файл).

```
* * * * * echo "Hello" >> /home/dd/file_cron
```



5. `sudo add-apt-repository ppa:ondrej/php`  
`sudo add-apt-repository --remove ppa:ondrej/php`

## Урок 5. Настройка сети в Linux. Работа с IPtables

### Задание

- Настроить статическую конфигурацию (без DHCP) в Ubuntu через ip и netplan. Настроить IP, маршрут по умолчанию и DNS-сервера (1.1.1.1 и 8.8.8.8).

Проверить работоспособность сети.

- Настроить правила iptables для доступности сервисов на TCP-портах 22, 80 и 443.

Также сервер должен иметь возможность устанавливать подключения к серверу обновлений.

Остальные подключения запретить.

- Запретить любой входящий трафик с IP 3.4.5.6.
- \* Запросы на порт 8090 перенаправлять на порт 80 (на этом же сервере).
- \* Разрешить подключение по SSH только из сети 192.168.0.0/24.

task1:

```
dd@dd-VirtualBox: ~  
GNU nano 6.2 /etc/netplan/01-network-manager-all.yaml  
# Let NetworkManager manage all devices on this system  
network:  
  version: 2  
  renderer: NetworkManager  
  ethernets:  
    enp0s3:  
      dhcp4: no  
      addresses: [192.168.1.222/24]  
      routes:  
        - to: default  
          via: 192.168.1.1  
      nameservers:  
        addresses:  
          - 8.8.8.8  
          - 1.1.1.1
```

```
Microsoft Windows [Version 10.0.22621.2361]  
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.  
  
C:\Users\slim2>ssh dd@192.168.1.113  
  
C:\Users\slim2>ssh dd@192.168.1.222  
The authenticity of host '192.168.1.222 (192.168.1.222)' can't be established.  
ED25519 key fingerprint is SHA256:LcEUKAA9f8MbBwSxktf1LVtE8x54XpLcJZhXeNPp1E.  
This host key is known by the following other names/addresses:  
  C:\Users\slim2/.ssh/known_hosts:1: 192.168.1.113  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.1.222' (ED25519) to the list of known hosts.  
dd@192.168.1.222's password:  
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 6.2.0-33-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
Расширенное поддержание безопасности (ESM) для Applications выключено.  
  
29 обновлений может быть применено немедленно.  
27 из этих обновлений, являются стандартными обновлениями безопасности.  
Чтобы просмотреть дополнительные обновления выполните: apt list --upgradable  
  
Включите ESM Apps для получения дополнительных будущих обновлений безопасности.  
Смотрите https://ubuntu.com/esm или выполните: sudo pro status  
  
Last login: Thu Oct  5 15:47:48 2023 from 192.168.1.107  
dd: /home/dd  
>
```



task2:

```
dd@dd-VirtualBox: ~  
dd: /home/dd  
> sudo iptables -A INPUT -p tcp --dport=22 -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -p tcp -m multiport --dport 80,443 -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -i lo -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -p icmp -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT  
dd: /home/dd  
> sudo iptables -P INPUT DROP  
dd: /home/dd  
> |
```

```
dd@dd-VirtualBox: ~  
dd: /home/dd  
> sudo iptables -A INPUT -p tcp --dport=22 -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -p tcp -m multiport --dport 80,443 -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -i lo -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -p icmp -j ACCEPT  
dd: /home/dd  
> sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT  
dd: /home/dd  
> sudo iptables -P INPUT DROP  
dd: /home/dd  
> sudo apt update  
Сущ:1 http://ru.archive.ubuntu.com/ubuntu jammy InRelease  
Сущ:2 http://ru.archive.ubuntu.com/ubuntu jammy-updates InRelease  
Сущ:3 http://ru.archive.ubuntu.com/ubuntu jammy-backports InRelease  
Сущ:4 https://download.docker.com/linux/ubuntu jammy InRelease  
Сущ:5 http://security.ubuntu.com/ubuntu jammy-security InRelease  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Может быть обновлено 27 пакетов. Запустите «apt list --upgradable» для их показа.  
dd: /home/dd  
> |
```

task3:

```
dd: /home/dd
> sudo iptables -t filter -A INPUT -s 3.4.5.6 -j DROP
dd: /home/dd
> sudo iptables -L -nv
Chain INPUT (policy DROP 119 packets, 23303 bytes)
  pkts bytes target     prot opt in     out     source           destination
    325 19504 ACCEPT     tcp  --  *      *        0.0.0.0/0        0.0.0.0/0        tcp dpt:22
    0    0 ACCEPT     tcp  --  *      *        0.0.0.0/0        0.0.0.0/0        multiport dpo
rts 80,443
    24  3077 ACCEPT     all  --  lo     *        0.0.0.0/0        0.0.0.0/0
    0    0 ACCEPT     icmp --  *      *        0.0.0.0/0        0.0.0.0/0
    62 32591 ACCEPT     all  --  *      *        0.0.0.0/0        0.0.0.0/0        state RELATED
,ESTABLISHED
    0    0 DROP       all  --  *      *        3.4.5.6          0.0.0.0/0

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination
dd: /home/dd
> |
```

task 4:

iptables -t nat -I PREROUTING -p tcp --dport 8090 -j REDIRECT --to-port 80

```
dd@dd-VirtualBox: ~
dd: /home/dd
> sudo iptables -L -nv -t nat
Chain PREROUTING (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination
    0    0 REDIRECT  tcp  --  *      *        0.0.0.0/0        0.0.0.0/0        tcp dpt:8090 redir ports 80

Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination

Chain POSTROUTING (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination
dd: /home/dd
> |
```

task 5:

sudo iptables -A INPUT -s 192.168.0.0/24 -p tcp --dport 22 -j ACCEPT

```
dd@dd-VirtualBox: ~
dd: /home/dd
> sudo iptables -L -nv
Chain INPUT (policy DROP 436 packets, 84493 bytes)
  pkts bytes target     prot opt in     out     source           destination
    531 34260 ACCEPT     tcp  --  *      *        0.0.0.0/0        0.0.0.0/0        tcp dpt:22
    0    0 ACCEPT     tcp  --  *      *        0.0.0.0/0        0.0.0.0/0        multiport dports 80,443
    25  3150 ACCEPT     all  --  lo     *        0.0.0.0/0        0.0.0.0/0
    0    0 ACCEPT     icmp --  *      *        0.0.0.0/0        0.0.0.0/0
    75 34380 ACCEPT     all  --  *      *        0.0.0.0/0        0.0.0.0/0        state RELATED,ESTABLISHED
    0    0 DROP       all  --  *      *        3.4.5.6          0.0.0.0/0
    0    0 ACCEPT     tcp  --  *      *        192.168.0.0/24   0.0.0.0/0        tcp dpt:22

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
  pkts bytes target     prot opt in     out     source           destination
dd: /home/dd
> |
```

# Урок 6. Запуск стека для веб-приложения

## Задание

- Установить Nginx и настроить его на работу с PHP-FPM.
- Установить Apache. Настроить обработку PHP. Добиться одновременной работы с Nginx.
- Настроить схему обратного прокси для Nginx (динамика - на Apache).
- Установить MySQL. Создать новую базу данных и таблицу в ней.
- \* Установить пакет phpmuadmin и запустить его веб-интерфейс для управления MySQL.
- \* Настроить схему балансировки трафика между несколькими серверами Apache на стороне Nginx с помощью модуля ngx\_http\_upstream\_module.

```
sudo apt install apache2
```

```
sudo apt install nginx
```

```
dd: /home/dd
> sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-10-08 18:23:20 MSK; 1h 14min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 789 (apache2)
    Tasks: 55 (limit: 4599)
   Memory: 7.6M
      CPU: 526ms
   CGroup: /system.slice/apache2.service
           └─789 /usr/sbin/apache2 -k start
             └─791 /usr/sbin/apache2 -k start
               └─792 /usr/sbin/apache2 -k start

окт 08 18:23:20 dd-VirtualBox systemd[1]: Starting The Apache HTTP Server...
окт 08 18:23:20 dd-VirtualBox apachectl[765]: AH00558: apache2: Could not reliably determine the
окт 08 18:23:20 dd-VirtualBox systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

```
sudo nano /etc/apache2/ports.conf
```

Меняем порты





# Ubuntu

## Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

### Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```

/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf

```


- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts.

sudo apt install php8.1

PHP 8.1.2-1ubuntu2.14 - php

127.0.0.1/info.php

PHP Version 8.1.2-1ubuntu2.14



System	Linux dd-VirtualBox 6.2.0-34-generic #34~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Thu Sep 7 13:12:03 UTC 2 x86_64
Build Date	Aug 18 2023 11:41:11
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-ctype.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-finfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-phar.ini, /etc/php/8.1/apache2/conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /etc/php/8.1/apache2/conf.d/20-shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysmsg.ini, /etc/php/8.1/apache2/conf.d/20-syssem.ini, /etc/php/8.1/apache2/conf.d/20-sysshm.ini, /etc/php/8.1/apache2/conf.d/20-tokyo.ini

sudo apt install mysql-server -y

```
dd: /var/www/html
> sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.34-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/1
	/8.1/apache2/conf.d/20-calend
	/conf.d/20-exif.ini, /etc/php/8.1
	/etc/php/8.1/apache2/conf.d/2
	/8.1/apache2/conf.d/20-iconv.i
	/conf.d/20-posix.ini, /etc/php/8
	shmop.ini, /etc/php/8.1/apach
	/etc/php/8.1/apache2/conf.d/2
	/8.1/apache2/conf.d/20-tokeni
	20210902
PHP Extension	20210902
Zend Extension	20210902
Zend Extension Build	API420210902.NTS
PHP Extension Build	API20210902.NTS

```
mysql> SELECT * FROM users;
+-----+-----+-----+
| id    | name  | age  |
+-----+-----+-----+
| 1     | DENIS | 35   |
| 5     | Artem | 6    |
+-----+-----+-----+
2 rows in set (0,00 sec)

mysql>
```

PHP 8.1.2-1ubuntu2.14 - php X +

→ 127.0.0.1/info.php

This program makes use of the Zend Scripting Language Engine:  
Zend Engine v4.1.2. Copyright (c) Zend Technologies  
with Zend OPcache v8.1.2-1ubuntu2.14, Copyright (c), by Zend Technologies

zend engine

Configuration

calendar

Calendar support	enabled
------------------	---------

cgi-fcgi

php-fpm	active	
Directive	Local Value	Master Value
cgi.disable_path	Off	Off
cgi.fix_pathinfo	On	On
cgi.force_redirect	On	On
cgi.nph	Off	Off
cgi.redirect_status_env	no value	no value
cgi.rfc2616_headers	Off	Off
fastcgi.error_header	no value	no value
fastcgi.logging	On	On
fpm.config	no value	no value

Урок 7. Запуск веб-приложения из контейнеров

Задание

- Установить в виртуальную машину или VDS Docker, настроить набор контейнеров через docker compose по [инструкции](#).

Часть с настройкой certbot и HTTPS опустить, если у вас нет настоящего домена и белого IP.

- Запустить два контейнера, связанные одной сетью (используя документацию). Первый контейнер БД (например, образ mariadb:10.8), второй контейнер — phpmyadmin. Получить доступ к БД в первом контейнере через второй контейнер (веб-интерфейс phpmyadmin).

[https://www.digitalocean.com/community/tutorials/how-to-install-wordpress-with-docker-comp  
ose.ru](https://www.digitalocean.com/community/tutorials/how-to-install-wordpress-with-docker-compose)

```
mkdir wordpress && cd wordpress
```

```
mkdir nginx-conf
```

```
nano nginx-conf/nginx.conf
```

```
server {
```

```
    listen 80;
```

```
    listen [::]:80;
```

```
    server_name example.com www.example.com;
```

```
    index index.php index.html index.htm;
```

```
    root /var/www/html;
```

```
    location ~ /.well-known/acme-challenge {
```

```
        allow all;
```

```
        root /var/www/html;
```

```
    }
```

```
    location / {
```

```
        try_files $uri $uri/ /index.php$is_args$args;
```

```
    }
```

```
location ~ /\.php$ {  
  
    try_files $uri =404;  
  
    fastcgi_split_path_info ^(.+\.php)(/.+)$;  
  
    fastcgi_pass wordpress:9000;  
  
    fastcgi_index index.php;  
  
    include fastcgi_params;  
  
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;  
  
    fastcgi_param PATH_INFO $fastcgi_path_info;  
  
}
```

```
location ~ /\.ht {  
  
    deny all;  
  
}
```

```
location = /favicon.ico {  
  
    log_not_found off; access_log off;  
  
}
```

```
location = /robots.txt {  
  
    log_not_found off; access_log off; allow all;  
  
}
```

```
location ~* \.(css|gif|ico|jpeg|jpg|js|png)$ {  
  
    expires max;  
  
    log_not_found off;
```



```
}  
  
}
```

nano .env

```
MYSQL_ROOT_PASSWORD=2312  
MYSQL_USER=db_denis  
MYSQL_PASSWORD=2312
```

```
MYSQL_ROOT_PASSWORD=2312  
MYSQL_USER=db_denis  
MYSQL_PASSWORD=2312
```

nano .dockerignore

```
.env  
.git  
docker-compose.yml  
.dockerignore
```

nano docker-compose.yml

```
version: '3'  
  
services:  
  db:  
    image: mysql:8.0  
    container_name: db  
    restart: unless-stopped  
    env_file: .env  
    environment:  
      - MYSQL_DATABASE=wordpress  
    volumes:  
      - dbdata:/var/lib/mysql  
    command: '--default-authentication-plugin=mysql_native_password'  
    networks:  
      - app-network  
  
  wordpress:  
    depends_on:  
      - db  
    image: wordpress:5.1.1-fpm-alpine  
    container_name: wordpress  
    restart: unless-stopped
```

```

    env_file: .env
    environment:
      - WORDPRESS_DB_HOST=db:3306
      - WORDPRESS_DB_USER=$MYSQL_USER
      - WORDPRESS_DB_PASSWORD=$MYSQL_PASSWORD
      - WORDPRESS_DB_NAME=wordpress
    volumes:
      - wordpress:/var/www/html
    networks:
      - app-network

```

```

webserver:
  depends_on:
    - wordpress
  image: nginx:1.15.12-alpine
  container_name: webserver
  restart: unless-stopped
  ports:
    - "80:80"
  volumes:
    - wordpress:/var/www/html
    - ./nginx-conf:/etc/nginx/conf.d
  networks:
    - app-network

```

```
volumes:
```

```

wordpress:
  dbdata:

```

```

networks:
  app-network:
    driver: bridge

```

sudo docker-compose up -d

```

> sudo docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                               NAMES
5cc21b226908   nginx:1.15.12-alpine               "nginx -g 'daemon of..." 6 minutes ago  Up 28 seconds  0.0.0.0:80->80/tcp, :::80->80/tcp   webserver
a6527ed251ce   wordpress:5.1.1-fpm-alpine         "docker-entrypoint.s..." 6 minutes ago  Up 6 minutes   9000/tcp                           wordpress
87ced96e2f5f   mysql:8.0                          "docker-entrypoint.s..." 6 minutes ago  Up 6 minutes   3306/tcp, 33060/tcp                db
dd: /home/dd/wordpress

```



English (United States)

Afrikaans

العربية

العربية المغربية

অসমীয়া

گۆنئی آذربایجان

Azərbaycan dili

Беларуская мова

Български

বাংলা

བོད་སྐད་

Bosanski

Català

Cebuano

Čeština

Cymraeg

Dansk

Deutsch (Österreich)

Deutsch (Schweiz, Du)

Deutsch (Schweiz)

Deutsch (Sie)

Deutsch

தமிழ்

Ελληνικά

Continue