

Отчет по лабораторной работе 4

Генералов Даниил, НПИбд-01-21, 1032202280

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	15
5	Контрольные вопросы	16

Список иллюстраций

3.1	dnf	7
3.2	httpd	8
3.3	httpd	9
3.4	httpd	10
3.5	httpd	11
3.6	dns	12
3.7	dns	12
3.8	httpd	13
3.9	httpd	13
3.10	httpd	14
3.11	vagrant	14

Список таблиц

1 Цель работы

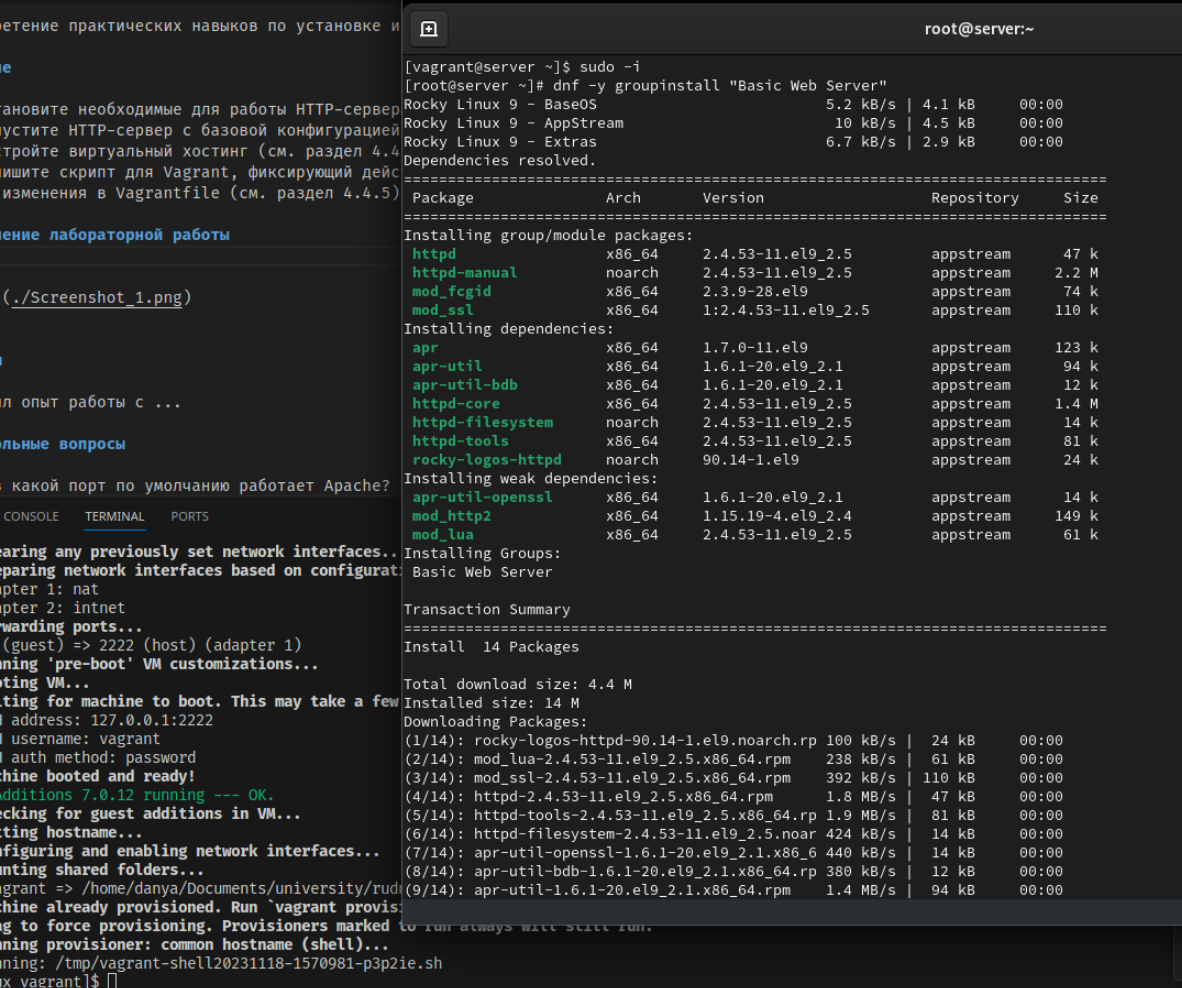
Приобретение практических навыков по установке и базовому конфигурированию HTTP-сервера Apache.

2 Задание

1. Установите необходимые для работы HTTP-сервера пакеты (см. раздел 4.4.1).
2. Запустите HTTP-сервер с базовой конфигурацией и проанализируйте его работу (см. разделы 4.4.2 и 4.4.3).
3. Настройте виртуальный хостинг (см. раздел 4.4.4).
4. Напишите скрипт для Vagrant, фиксирующий действия по установке и настройке HTTP-сервера во внутреннем окружении виртуальной машины `server`. Соответствующим образом внесите изменения в `Vagrantfile` (см. раздел 4.4.5).

3 Выполнение лабораторной работы

Сначала я запустил виртуальную машину сервера и установил группу пакетов “Basic Web Server”, включая httpd и утилиты.



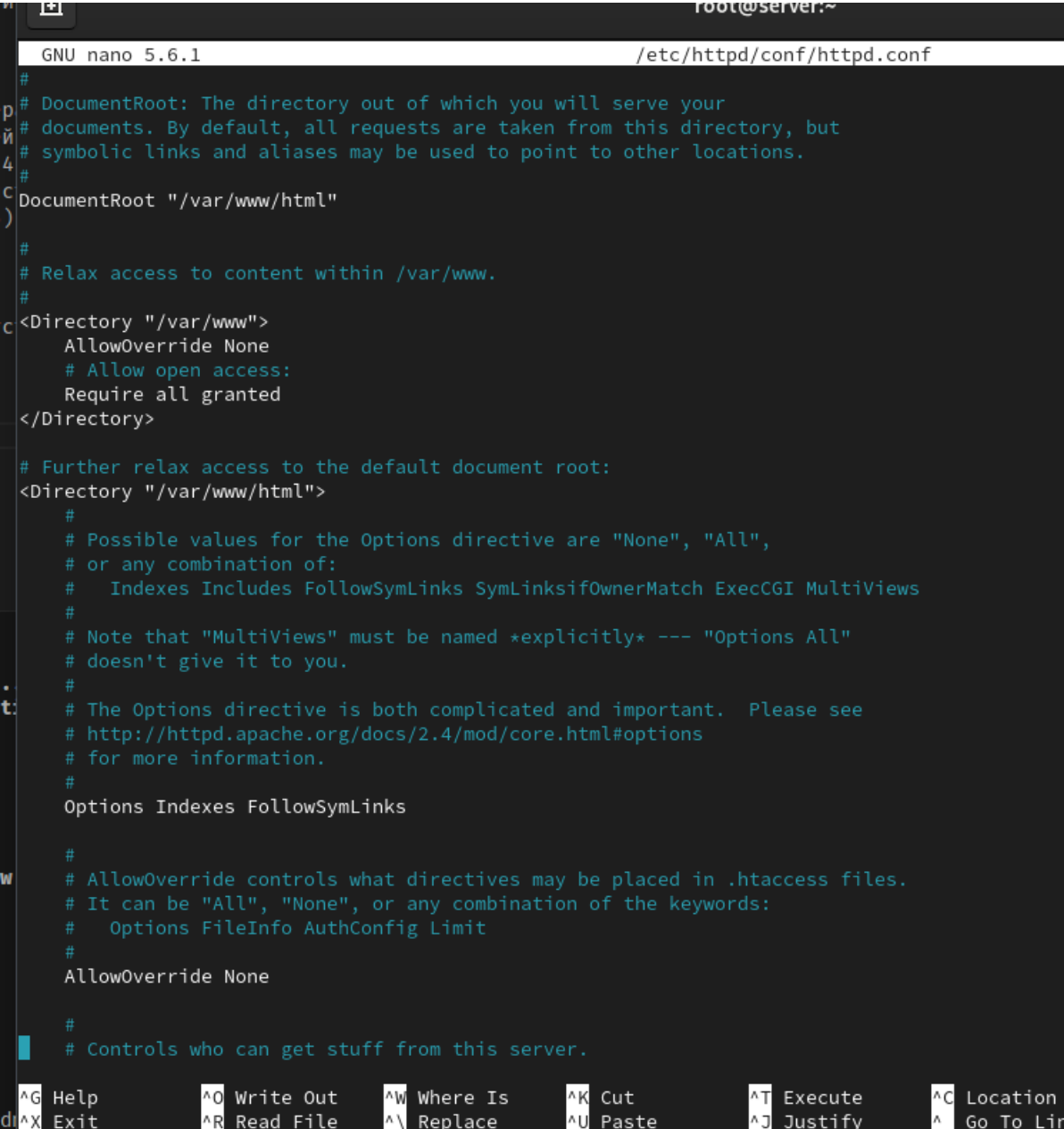
```
[vagrant@server ~]$ sudo -i
[root@server ~]# dnf -y groupinstall "Basic Web Server"
Rocky Linux 9 - BaseOS                    5.2 kB/s | 4.1 kB    00:00
Rocky Linux 9 - AppStream                  10 kB/s | 4.5 kB    00:00
Rocky Linux 9 - Extras                     6.7 kB/s | 2.9 kB    00:00
Dependencies resolved.
=====
Package                                Arch      Version              Repository           Size
=====
Installing group/module packages:
httpd                                   x86_64    2.4.53-11.el9_2.5    appstream            47 k
httpd-manual                           noarch    2.4.53-11.el9_2.5    appstream            2.2 M
mod_fcgid                               x86_64    2.3.9-28.el9         appstream            74 k
mod_ssl                                 x86_64    1:2.4.53-11.el9_2.5  appstream            110 k
Installing dependencies:
apr                                      x86_64    1.7.0-11.el9         appstream            123 k
apr-util                               x86_64    1.6.1-20.el9_2.1     appstream            94 k
apr-util-bdb                           x86_64    1.6.1-20.el9_2.1     appstream            12 k
httpd-core                             x86_64    2.4.53-11.el9_2.5    appstream            1.4 M
httpd-filesystem                       noarch    2.4.53-11.el9_2.5    appstream            14 k
httpd-tools                             x86_64    2.4.53-11.el9_2.5    appstream            81 k
rocky-logos-httpd                      noarch    90.14-1.el9          appstream            24 k
Installing weak dependencies:
apr-util-openssl                       x86_64    1.6.1-20.el9_2.1     appstream            14 k
mod_http2                              x86_64    1.15.19-4.el9_2.4    appstream            149 k
mod_lua                                 x86_64    2.4.53-11.el9_2.5    appstream            61 k
Installing Groups:
Basic Web Server

Transaction Summary
=====
Install 14 Packages

Total download size: 4.4 M
Installed size: 14 M
Downloading Packages:
(1/14): rocky-logos-httpd-90.14-1.el9.noarch.rpm 100 kB/s | 24 kB    00:00
(2/14): mod_lua-2.4.53-11.el9_2.5.x86_64.rpm    238 kB/s | 61 kB    00:00
(3/14): mod_ssl-2.4.53-11.el9_2.5.x86_64.rpm    392 kB/s | 110 kB   00:00
(4/14): httpd-2.4.53-11.el9_2.5.x86_64.rpm      1.8 MB/s | 47 kB    00:00
(5/14): httpd-tools-2.4.53-11.el9_2.5.x86_64.rpm 1.9 MB/s | 81 kB    00:00
(6/14): httpd-filesystem-2.4.53-11.el9_2.5.noar 424 kB/s | 14 kB    00:00
(7/14): apr-util-openssl-1.6.1-20.el9_2.1.x86_6 440 kB/s | 14 kB    00:00
(8/14): apr-util-bdb-1.6.1-20.el9_2.1.x86_64.rpm 380 kB/s | 12 kB    00:00
(9/14): apr-util-1.6.1-20.el9_2.1.x86_64.rpm    1.4 MB/s | 94 kB    00:00
```

Рис. 3.1: dnf

Это создало несколько файлов настройки. Например, `/etc/httpd/conf/httpd.conf` указывает основные настройки сервера (вроде того, какие порты используются и под именем какого пользователя следует запускать сервер), а также разрешения на доступ к определенным папкам: `/var/www/html` разрешен, а остальные папки запрещены.



```
root@server:~  
GNU nano 5.6.1 /etc/httpd/conf/httpd.conf  
#  
# DocumentRoot: The directory out of which you will serve your  
# documents. By default, all requests are taken from this directory, but  
# symbolic links and aliases may be used to point to other locations.  
#  
DocumentRoot "/var/www/html"  
#  
# Relax access to content within /var/www.  
#  
<Directory "/var/www">  
    AllowOverride None  
    # Allow open access:  
    Require all granted  
</Directory>  
# Further relax access to the default document root:  
<Directory "/var/www/html">  
    #  
    # Possible values for the Options directive are "None", "All",  
    # or any combination of:  
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews  
    #  
    # Note that "MultiViews" must be named *explicitly* --- "Options All"  
    # doesn't give it to you.  
    #  
    # The Options directive is both complicated and important. Please see  
    # http://httpd.apache.org/docs/2.4/mod/core.html#options  
    # for more information.  
    #  
    Options Indexes FollowSymLinks  
    #  
    # AllowOverride controls what directives may be placed in .htaccess files.  
    # It can be "All", "None", or any combination of the keywords:  
    #   Options FileInfo AuthConfig Limit  
    #  
    AllowOverride None  
    #  
    # Controls who can get stuff from this server.
```

Рис. 3.2: httpd

В папке `/etc/httpd/conf.d` находятся модули конфигурации, которые отвечают за отдельные части поведения сервера: например, тут есть настройки для обслуживания `https`-запросов, настройки для авто-индексирования, которое генерирует вывод информации о файлах в папках, которые не содержат `index.html`-файлов, а также инструкции отображать пример HTML-страницы, если в папку с данными сервера еще ничего не было помещено.

```
[root@server ~]# nano /etc/httpd/conf/
httpd.conf magic
[root@server ~]# nano /etc/httpd/conf/
httpd.conf magic
[root@server ~]# nano /etc/httpd/conf/httpd.conf
[root@server ~]# nano /etc/httpd/conf.d/
autoindex.conf  fcgid.conf      manual.conf     README          ssl.conf        userdir.conf    welcome.conf
[root@server ~]# nano /etc/httpd/conf.d/welcome.conf
[root@server ~]# nano /etc/httpd/conf.d/ssl.conf
[root@server ~]# nano /etc/httpd/conf.d/README
[root@server ~]# cat /etc/httpd/conf.d/autoindex.conf
#
# Directives controlling the display of server-generated directory listings.
#
# Required modules: mod_authz_core, mod_authz_host,
#                  mod_autoindex, mod_alias
#
# To see the listing of a directory, the Options directive for the
# directory must include "Indexes", and the directory must not contain
# a file matching those listed in the DirectoryIndex directive.
#
#
# IndexOptions: Controls the appearance of server-generated directory
# listings.
#
IndexOptions FancyIndexing HTMLTable VersionSort

# We include the /icons/ alias for FancyIndexed directory listings.  If
# you do not use FancyIndexing, you may comment this out.
#
Alias /icons/ "/usr/share/httpd/icons/"

<Directory "/usr/share/httpd/icons">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

#
# AddIcon* directives tell the server which icon to show for different
# files or filename extensions.  These are only displayed for
# FancyIndexed directories.
#
```

Рис. 3.3: httpd

Теперь можно запустить HTTP-сервер и увидеть в логах, что он успешно запустился.

```

The job identifier is 520.
^Z
[1]+  Stopped                  journalctl -x -f
[root@server ~]# bg
[1]+  journalctl -x -f &
[root@server ~]# firewall-cmd --add-service=http
Warning: ALREADY_ENABLED: 'http' already in 'public'
success
[root@server ~]# Nov 18 13:38:31 server.user.net firewalld[625]: WARNING: ALREADY_ENABLED: 'http' already in 'public'

[root@server ~]# firewall-cmd --add-service=http --permanent
Warning: ALREADY_ENABLED: http
success
[root@server ~]# Nov 18 13:38:35 server.user.net firewalld[625]: WARNING: ALREADY_ENABLED: http

[root@server ~]# systemctl enable --now httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
Nov 18 13:38:49 server.user.net systemd[1]: Reloading.
Nov 18 13:38:49 server.user.net systemd-rc-local-generator[6639]: /etc/rc.d/rc.local is not marked executable, skipping.
Nov 18 13:38:50 server.user.net systemd[1]: Starting One-time temporary TLS key generation for httpd.service...
Subject: A start job for unit httpd-init.service has begun execution
Defined-By: systemd
Support: https://access.redhat.com/support

A start job for unit httpd-init.service has begun execution.

The job identifier is 2736.
Nov 18 13:38:50 server.user.net systemd[1]: Starting dnf makecache...

```

Рис. 3.4: httpd

После этого можно запустить клиент и попробовать подключиться к серверу, и увидеть страницу по умолчанию. Из логов сервера можно увидеть: - что сервер хотел отобразить файл index.html в папке /var/www/html, но его там не было обнаружено, и авто-индексирование выключено; - что клиент 192.168.1.30 сделал несколько запросов по ссылкам /, /icons/poweredby.png, /poweredby.png, /favicon.ico, и получил коды ответа 403, 200, 200, 404; - что клиент использовал браузер Mozilla/5.0 на Linux x86_64 на X11.

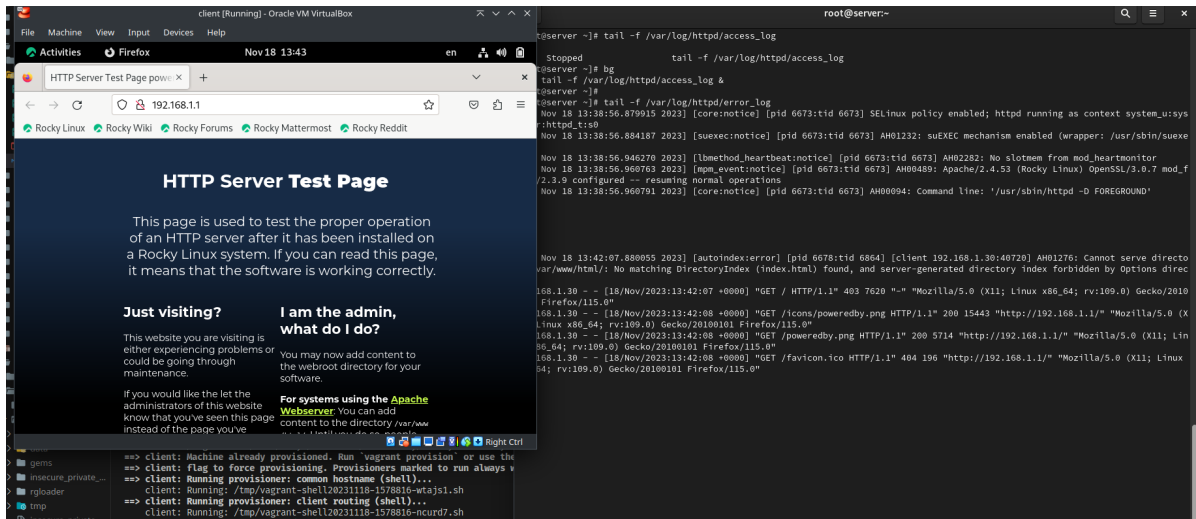


Рис. 3.5: httpd

Теперь можно настроить виртуальный хостинг. Для этого нужно иметь несколько DNS-записей, которые указывают на сервер. Мы добавляем запись `www.dmgeneralov.net`.

```
root@server:~
GNU nano 5.6.1 /var/named/master/fz/dmgeneralov.net
$ORIGIN .
$TTL 86400 ; 1 day
dmgeneralov.net IN SOA dmgeneralov.net. server.dmgeneralov.net. (
                        2023118603 ; serial
                        86400 ; refresh (1 day)
                        3600 ; retry (1 hour)
                        604800 ; expire (1 week)
                        10800 ; minimum (3 hours)
                        )
                        NS dmgeneralov.net.
                        A 192.168.1.1
$ORIGIN dmgeneralov.net.
dhcp A 192.168.1.1
ns A 192.168.1.1
server A 192.168.1.1
www A 192.168.1.1
```

Рис. 3.6: dns

Мы также добавляем ее в обратную зону.

```
GNU nano 5.6.1 /var/named/master/rz/192.168.1
$ORIGIN .
$TTL 86400 ; 1 day
1.168.192.in-addr.arpa IN SOA 1.168.192.in-addr.arpa. server.dmgeneralov.net. (
                            2023111800 ; serial
                            86400 ; refresh (1 day)
                            3600 ; retry (1 hour)
                            604800 ; expire (1 week)
                            10800 ; minimum (3 hours)
                            )
                            NS 1.168.192.in-addr.arpa.
                            A 192.168.1.1
                            PTR server.dmgeneralov.net.
$ORIGIN 1.168.192.in-addr.arpa.
1 PTR server.dmgeneralov.net.
PTR ns.dmgeneralov.net.
PTR dhcp.dmgeneralov.net.
PTR www.dmgeneralov.net.
```

Рис. 3.7: dns

Затем создаем два файла настройки HTTP-сервера.

```
root@server ~]# systemctl restart named
root@server ~]# nano /etc/httpd/conf.d/server.dmgeneralov.net.conf
root@server ~]# nano /etc/httpd/conf.d/^C .dmgeneralov.net.conf
root@server ~]# nano /etc/httpd/conf.d/server.dmgeneralov.net.conf /etc/httpd/conf.d/www.dmgeneralov.net.conf
root@server ~]# cat /etc/httpd/conf.d/server.dmgeneralov.net.conf
VirtualHost *:80>
    ServerAdmin webmaster@dmgeneralov.net
    DocumentRoot /var/www/html/server.dmgeneralov.net
    ServerName server.dmgeneralov.net
    ErrorLog logs/server.dmgeneralov.net-error_log
    CustomLog logs/server.dmgeneralov.net-access_log common
</VirtualHost>
root@server ~]# cat /etc/httpd/conf.d/www.dmgeneralov.net.conf
VirtualHost *:80>
    ServerAdmin webmaster@dmgeneralov.net
    DocumentRoot /var/www/html/www.dmgeneralov.net
    ServerName www.dmgeneralov.net
    ErrorLog logs/www.dmgeneralov.net-error_log
    CustomLog logs/www.dmgeneralov.net-access_log common
</VirtualHost>
root@server ~]#
```

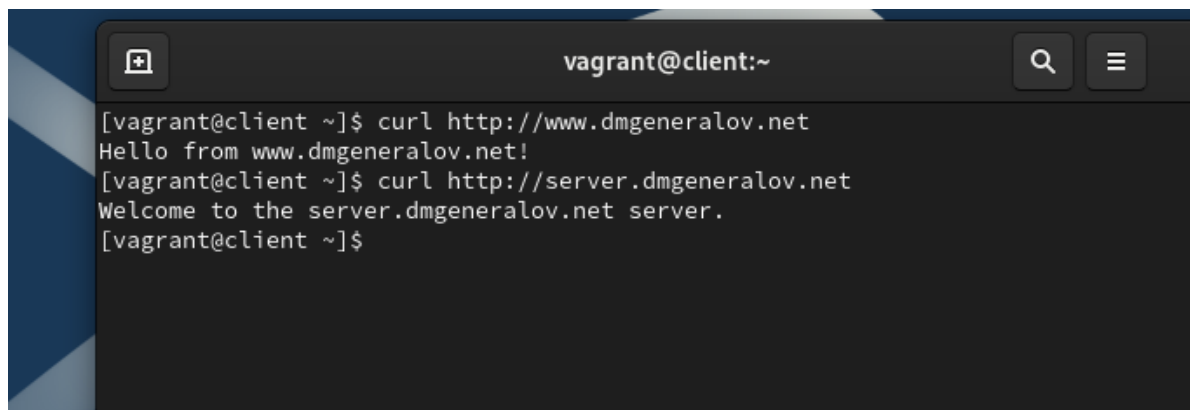
Рис. 3.8: httpd

После этого создаем папки с содержимым веб-страницы, затем перезапускаем сервер.

```
root@server:~
[root@server ~]# mkdir -p /var/www/html/server.dmgeneralov.net
[root@server ~]# cat > /var/www/html/server.dmgeneralov.net/index.html
Welcome to the server.dmgeneralov.net server.
[root@server ~]# mkdir -p /var/www/html/www.dmgeneralov.net
[root@server ~]# cat > /var/www/html/www.dmgeneralov.net/index.html
Hello from www.dmgeneralov.net!
[root@server ~]# chown -R apache:apache /var/www
[root@server ~]# restorecon -vR /etc/
Relabeled /etc/sysconfig/network-scripts/ifcfg-eth1 from unconfined_u:object_r:user_tmp_t:s0 to unconfined_u:object_r:net_conf_t:
s0
Relabeled /etc/systemd/system/dhcpd.service from unconfined_u:object_r:systemd_unit_file_t:s0 to unconfined_u:object_r:dhcpd_unit
_file_t:s0
[root@server ~]# restorecon -vR /var/named
[root@server ~]# restorecon -vR /var/www
[root@server ~]# systemctl restart httpd
[root@server ~]#
```

Рис. 3.9: httpd

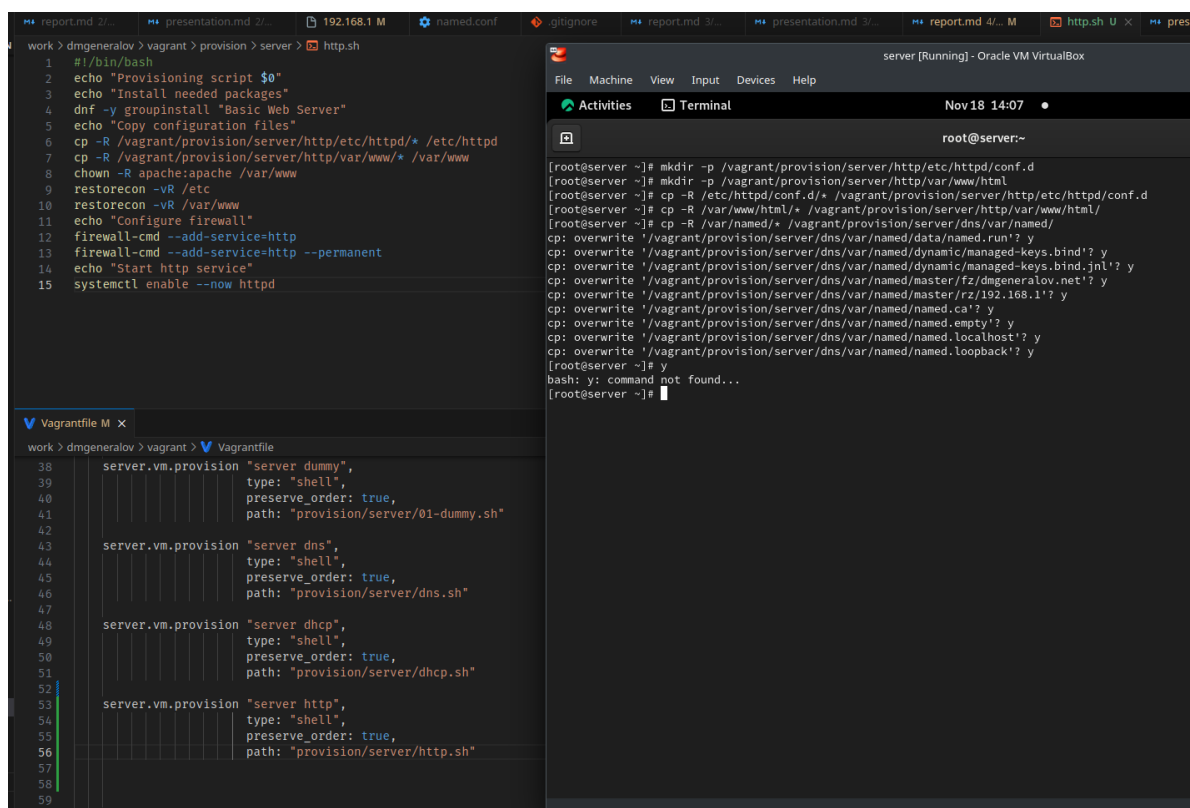
На клиенте можно получить обе страницы, хотя они находятся на одном и том же сервере.



```
vagrant@client:~  
[vagrant@client ~]$ curl http://www.dmgeneralov.net  
Hello from www.dmgeneralov.net!  
[vagrant@client ~]$ curl http://server.dmgeneralov.net  
Welcome to the server.dmgeneralov.net server.  
[vagrant@client ~]$
```

Рис. 3.10: httpd

Наконец, мы сохраняем все изменения в Vagrantfile.



```
work > dmgeneralov > vagrant > provision > server > http.sh  
1 #!/bin/bash  
2 echo "Provisioning script $0"  
3 echo "Install needed packages"  
4 dnf -y groupinstall "Basic Web Server"  
5 echo "Copy configuration files"  
6 cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd  
7 cp -R /vagrant/provision/server/http/var/www/* /var/www  
8 chown -R apache:apache /var/www  
9 restorecon -VR /etc  
10 restorecon -VR /var/www  
11 echo "Configure firewall"  
12 firewall-cmd --add-service=http  
13 firewall-cmd --add-service=http --permanent  
14 echo "Start http service"  
15 systemctl enable --now httpd  
  
Vagrantfile M x  
work > dmgeneralov > vagrant > Vagrantfile  
38 server.vm.provision "server dummy",  
39 | type: "shell",  
40 | preserve_order: true,  
41 | path: "provision/server/01-dummy.sh"  
42  
43 server.vm.provision "server dns",  
44 | type: "shell",  
45 | preserve_order: true,  
46 | path: "provision/server/dns.sh"  
47  
48 server.vm.provision "server dhcp",  
49 | type: "shell",  
50 | preserve_order: true,  
51 | path: "provision/server/dhcp.sh"  
52  
53 server.vm.provision "server http",  
54 | type: "shell",  
55 | preserve_order: true,  
56 | path: "provision/server/http.sh"  
57  
58  
59  
  
server [Running] - Oracle VM VirtualBox  
File Machine View Input Devices Help  
Activities Terminal Nov 18 14:07  
root@server:~  
[root@server ~]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d  
[root@server ~]# mkdir -p /vagrant/provision/server/http/var/www/html  
[root@server ~]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d  
[root@server ~]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html/  
[root@server ~]# cp -R /vagrant/provision/server/dns/var/named/* /vagrant/provision/server/dns/var/named/  
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind.jnl'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/dmgeneralov.net'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/named.ca'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/named.empty'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/named.localhost'? y  
cp: overwrite '/vagrant/provision/server/dns/var/named/named.loopback'? y  
[root@server ~]# y  
bash: y: command not found...  
[root@server ~]#
```

Рис. 3.11: vagrant

4 Выводы

Я получил опыт настройки сервера Apache, в частности настройки виртуального хостинга.

5 Контрольные вопросы

1. Через какой порт по умолчанию работает Apache?

Через порт 80 – стандартный порт для незашифрованного HTTP, потому что клиентские программы будут пытаться подключиться именно к этому порту по умолчанию.

2. Под каким пользователем запускается Apache и к какой группе относится этот пользователь?

Это можно изменить в файле настройки сервера, но по умолчанию это пользователь apache и группа apache.

3. Где располагаются лог-файлы веб-сервера? Что можно по ним отслеживать?

В файле `/var/log/httpd/error_log` можно увидеть ошибки, которые получаются при обслуживании запросов, и по этому можно найти возможные проблемы настройки сервера или PHP-кода сайта; в файле `/var/log/httpd/access_log` можно увидеть каждый запрос к серверу, что можно использовать для сбора статистики посещений сайта.

4. Где по умолчанию содержится контент веб-серверов?

`/var/www/html`

5. Каким образом реализуется виртуальный хостинг? Что он даёт?

Когда сервер получает HTTP-запрос, он смотрит на значение Host-заголовка, и в зависимости от него определяет, какие файлы стоит отправлять в ответ. Это позволяет размещать несколько сайтов на разных доменах, но на одном и том же сервере, чтобы экономить сервера и IP-адреса.