

Лабораторная работа № 4

Базовая настройка HTTP-сервера Apache

Абд эль хай мохамад

Содержание

<i>Цель работы</i>	2
<i>Выполнение лабораторной работы</i>	2
1. Установка НТТР-сервера	2
2. Базовое конфигурирование НТТР-сервера.....	2
3. Анализ работы НТТР-сервера.....	3
4. Внесение изменений в настройки внутреннего окружения виртуальной машины.....	9
Вывод:	11
Ответы на контрольные вопросы:	11

Цель работы

Приобретение практических навыков по установке и базовому конфигурированию

HTTP-сервера Apache

Выполнение лабораторной работы

1. Установка HTTP-сервера

1. Загружаю операционную систему и перехожу в рабочий каталог с проектом:
2. Запускаю виртуальную машину server:
3. На виртуальной машине server вхожу под своим пользователем и открываю терминал. Перехожу в режим суперпользователя
4. Устанавливаю из репозитория стандартный веб-сервер (HTTP-сервер и утилиты httpd, крипто-утилиты и пр.):

LANG=C yum grouplist

dnf -y groupinstall "Basic Web Server"

2. Базовое конфигурирование HTTP-сервера

1. Просматриваю содержание конфигурационных файлов в каталогах /etc/httpd/conf и /etc/httpd/conf.d.

manual.conf - этот файл конфигурации позволяет получить доступ к руководству по <http://localhost/manual/>

fcgid.conf - это файл конфигурации сервера Apache для обеспечения поддержки FastCGI (клиент-серверного протокола взаимодействия веб-сервера и приложения) через modfcgid (бинарно-совместимая альтернатива Apache модуля mod_fastcgi. mod_fcgid имеет новую стратегию в области управления процессом, которая концентрируется на снижении количества FastCGI серверов).

welcome.conf - этот файл конфигурации включает тестовую страницу по умолчанию, если для корневого URL-адреса нет страницы индекса по умолчанию.

maabedelhaydir.conf - файл настройки для каталогов пользователей сервера Apache HTTP называется — httpd-maabedelhaydir.conf. Он содержит

директивы и параметры, управляющие работой виртуальных серверов Apache.

`autoindex.conf` - этот конфигурационный файл содержит директивы, управляющие отображением списков каталогов, созданных сервером.

`ssl.conf` – это файл конфигурации сервера Apache, предоставляющий поддержку SSL. Он содержит директивы конфигурации для указания серверу, как обслуживать страницы через `https`-соединение.

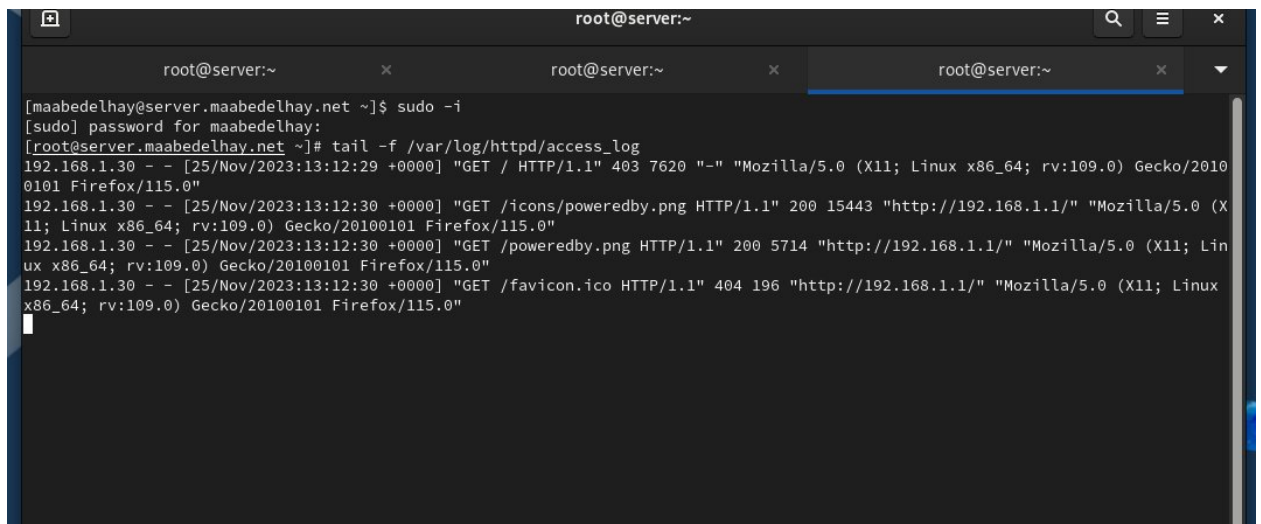
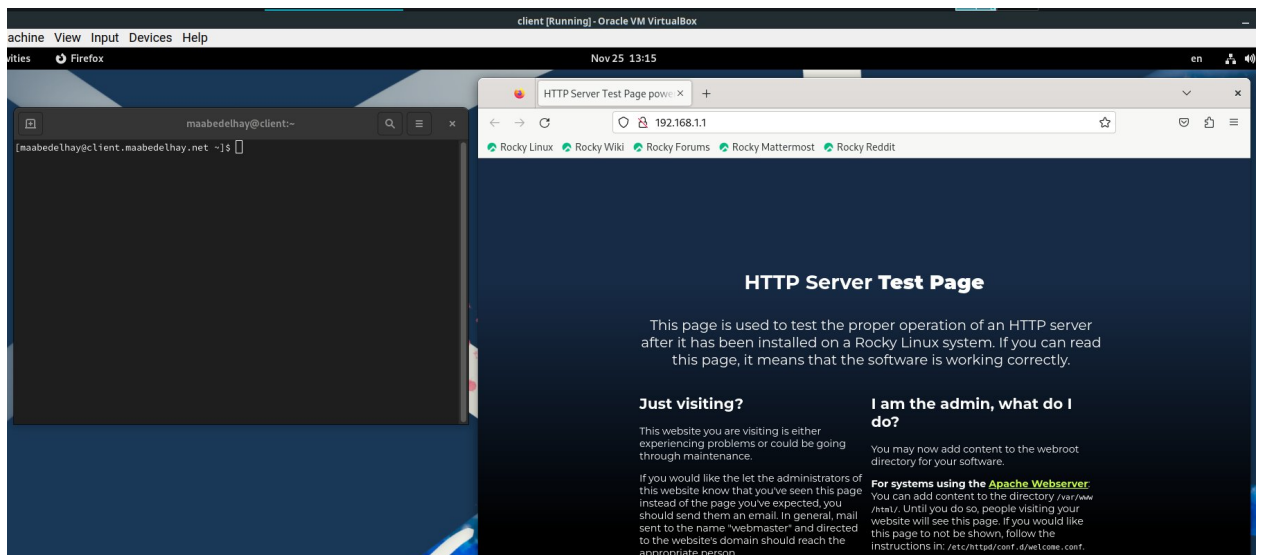
Файл `httpd.conf` содержит конфигурацию сервера - основное техническое описание работы демона.

2. Вношу изменения в настройки межсетевого экрана узла `server`, разрешив работу с `http`:
3. Во дополнительном терминале запускаю в режиме реального времени расширенный лог системных сообщений, чтобы проверить корректность работы системы:
`journalctl -x -f`
4. В первом терминале активирую и запускаю HTTP-сервер:
Веб-сервер успешно запустился.

3. Анализ работы HTTP-сервера

1. Запускаю виртуальную машину `client`:
2. На виртуальной машине `server` просматриваю лог ошибок работы веб-сервера:
3. На виртуальной машине `server` запускаю мониторинг доступа к веб-серверу:
4. На виртуальной машине `client` запускаю браузер и в адресной строке ввожу `192.168.1.1`.

Настройка виртуального хостинга для HTTP-сервера



1. Приостанавливаю работу DNS-сервера для внесения изменений в файлы описания DNS-зон: `systemctl stop named`
2. Добавляю запись для HTTP-сервера в конце файла прямой DNS-зоны `/var/named/master/fz/maabeldelhay.net`:
 server A 192.168.1.1
 www A 192.168.1.1
 и в конце файла обратной зоны `/var/named/master/rz/192.168.1`:
 1 PTR server. maabeldelhay.net.
 1 PTR www.maabeldelhay.net.
 В обоих файлах изменяю серийный номер файла зоны, указав текущую дату в нотации ГГГГММДДВВ. Также из соответствующих каталогов удаляю файлы журналов DNS: `maabeldelhay.net.jnl` и `192.168.1.jnl`.

```
root@server:~  
GNU nano 5.6.1 /var/named/master/fz/maabedelhay.net Modified  
$ORIGIN .  
$TTL 86400 ; 1 day  
maabedelhay.net IN SOA maabedelhay.net. server.maabedelhay.net. (  
2023112404 ; serial  
86400 ; refresh (1 day)  
3600 ; retry (1 hour)  
604800 ; expire (1 week)  
10800 ; minimum (3 hours)  
)  
NS maabedelhay.net.  
A 192.168.1.1  
$ORIGIN maabedelhay.net.  
$TTL 300 ; 5 minutes  
client A 192.168.1.30  
TXT "315ccd8665df3e8985384dc37b06da1612"  
$TTL 86400 ; 1 day  
dhcp A 192.168.1.1  
ns A 192.168.1.1  
server A 192.168.1.1  
www A 192.168.1.1  
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo  
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo
```

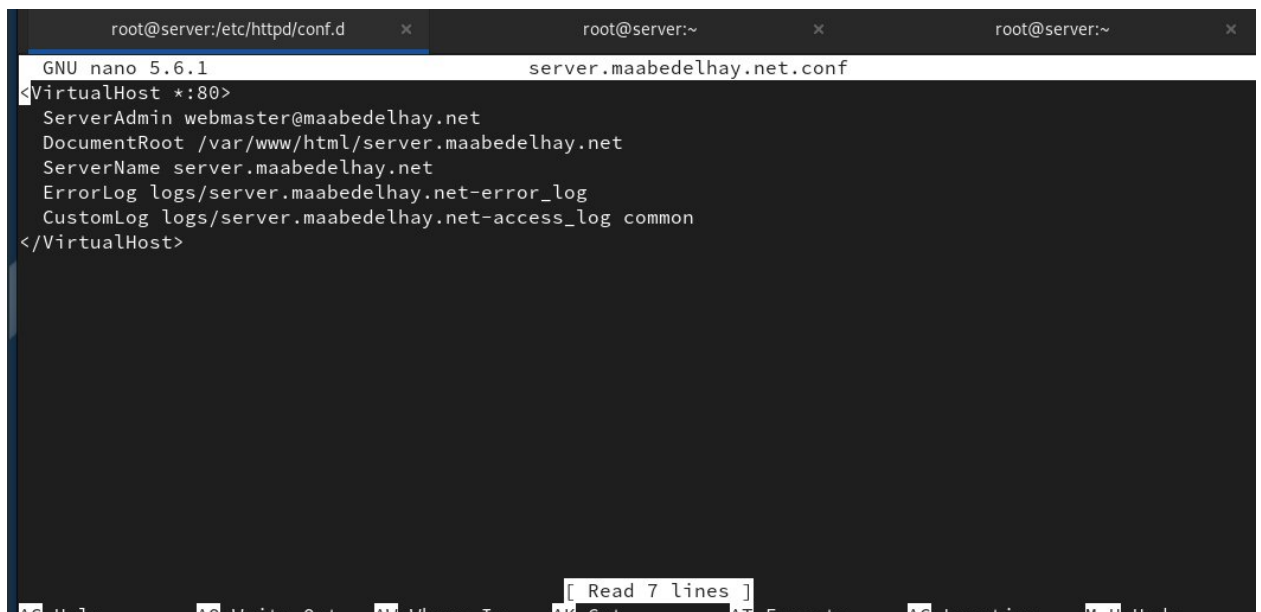
```
root@server:~  
GNU nano 5.6.1 /var/named/master/rz/192.168.1 Modified  
$ORIGIN .  
$TTL 86400 ; 1 day  
1.168.192.in-addr.arpa IN SOA 1.168.192.in-addr.arpa. server.maabedelhay.net. (  
2023112403 ; serial  
86400 ; refresh (1 day)  
3600 ; retry (1 hour)  
604800 ; expire (1 week)  
10800 ; minimum (3 hours)  
)  
NS 1.168.192.in-addr.arpa.  
A 192.168.1.1  
PTR server.maabedelhay.net.  
$ORIGIN 1.168.192.in-addr.arpa.  
1 PTR server.maabedelhay.net.  
1 PTR ns.maabedelhay.net.  
1 PTR dhcp.maabedelhay.net.  
1 PTR www.maabedelhay.net.  
$TTL 300 ; 5 minutes  
30 PTR client.maabedelhay.net.  
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo  
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo
```

3. Перезапускаю DNS-сервер:
4. В каталоге /etc/httpd/conf.d создаю файлы server.maabedelhay.net.conf и www.maabedelhay.net.conf:

```
[root@server.maabedelhay.net ~]# cd /etc/httpd/conf.d  
[root@server.maabedelhay.net conf.d]# touch server.maabedelhay.net.conf  
[root@server.maabedelhay.net conf.d]# touch www.maabedelhay.net.conf  
[root@server.maabedelhay.net conf.d]# ls  
autoindex.conf manual.conf server.maabedelhay.net.conf userdir.conf www.maabedelhay.net.conf  
fcgid.conf README ssl.conf welcome.conf  
[root@server.maabedelhay.net conf.d]#
```

5. Открываю на редактирование файл server.maabedelhay.net.conf и вношу следующее содержание:

```
<VirtualHost *:80>
ServerAdmin webmaster@maabedelhay.net
DocumentRoot /var/www/html/server.maabedelhay.net
ServerName server.maabedelhay.net
ErrorLog logs/server.maabedelhay.net-error_log
CustomLog logs/server.maabedelhay.net-access_log common
</VirtualHost>
```



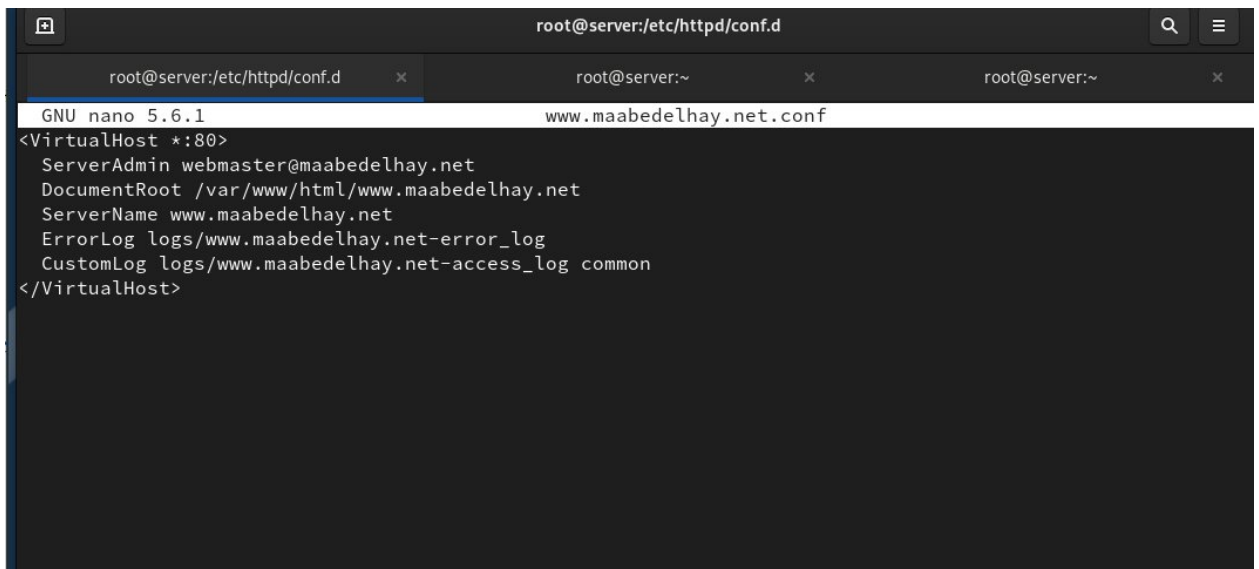
The screenshot shows a terminal window with three tabs: 'root@server:/etc/httpd/conf.d', 'root@server:~', and 'root@server:~'. The active tab is the first one, showing the nano 5.6.1 text editor editing the file 'server.maabedelhay.net.conf'. The content of the file is:

```
<VirtualHost *:80>
ServerAdmin webmaster@maabedelhay.net
DocumentRoot /var/www/html/server.maabedelhay.net
ServerName server.maabedelhay.net
ErrorLog logs/server.maabedelhay.net-error_log
CustomLog logs/server.maabedelhay.net-access_log common
</VirtualHost>
```

At the bottom of the terminal, there is a status bar with various icons and a button that says '[Read 7 lines]'.

6. Открываю на редактирование файл `www.maabedelhay.net.conf` и вношу следующее содержание:

```
<VirtualHost *:80>
ServerAdmin webmaster@maabedelhay.net
DocumentRoot /var/www/html/www.maabedelhay.net
ServerName www.maabedelhay.net
ErrorLog logs/www.maabedelhay.net-error_log
CustomLog logs/www.maabedelhay.net-access_log common
</VirtualHost>
```



```
root@server:/etc/httpd/conf.d
GNU nano 5.6.1 www.maabedelhay.net.conf
<VirtualHost *:80>
  ServerAdmin webmaster@maabedelhay.net
  DocumentRoot /var/www/html/www.maabedelhay.net
  ServerName www.maabedelhay.net
  ErrorLog logs/www.maabedelhay.net-error_log
  CustomLog logs/www.maabedelhay.net-access_log common
</VirtualHost>
```

7. Перехожу в каталог `/var/www/html`, в котором должны находиться файлы с содержимым (контентом) веб-серверов и создаю тестовые страницы для виртуальных веб-серверов `server.maabedelhay.net` и `www.maabedelhay.net`.

Для виртуального веб-сервера `server.maabedelhay.net`:

```
cd /var/www/html
mkdir server.maabedelhay.net
cd /var/www/html/server.maabedelhay.net
touch index.html
```

Открываю на редактирование файл `index.html` и вношу следующее содержание:

Welcome to the `server.maabedelhay.net` server.

Для виртуального веб-сервера `www.maabedelhay.net`:

```
cd /var/www/html
mkdir www.maabedelhay.net
cd /var/www/html/www.maabedelhay.net
```

Открываю на редактирование файл `index.html` и вношу следующее содержание:

Welcome to the `www.maabedelhay.net` server.

8. Корректирую права доступа в каталог с веб-контентом:

```
chown -R apache:apache /var/www
```

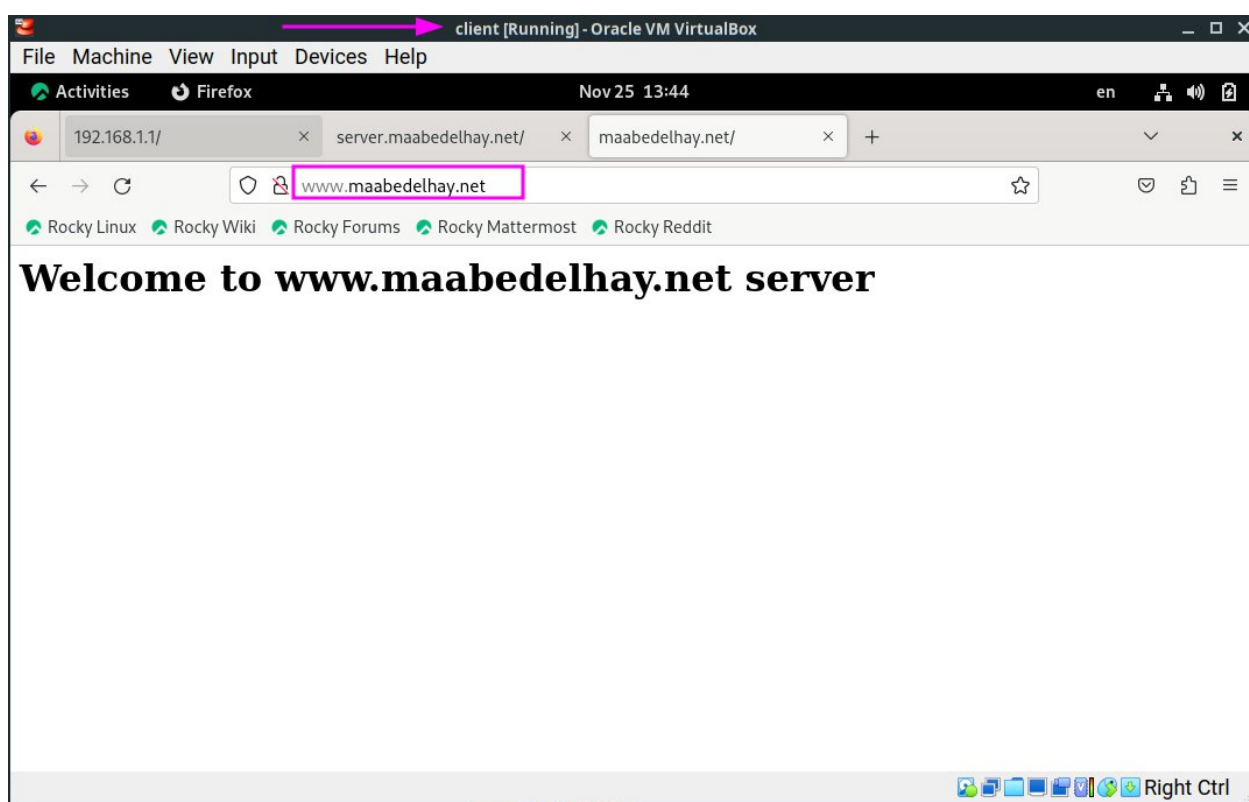
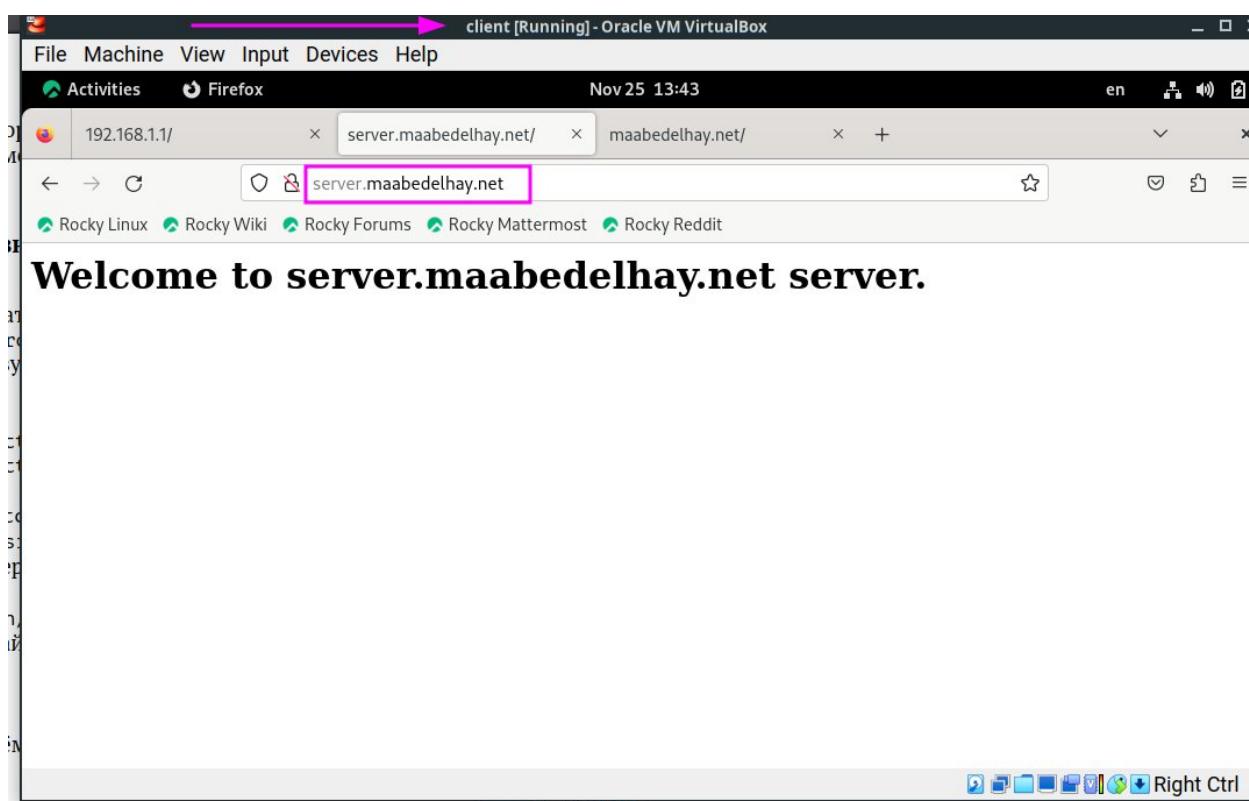
9. Восстанавливаю контекст безопасности в SELinux:

```
restorecon -vR /etc
restorecon -vR /var/named
restorecon -vR /var/www
```

10. Перезапускаю HTTP-сервер:


```
systemctl restart httpd
```

11. На виртуальной машине client проверяю доступ к веб-серверу по адресам `server.maabedelhay.net` и `www.maabedelhay.net` в адресной строке веб-браузера.



4. Внесение изменений в настройки внутреннего окружения виртуальной машины

1. На виртуальной машине server перехожу в каталог для внесения изменений в настройки внутреннего окружения /vagrant/provision/server/, создаю в нём каталог http, в который поместите в соответствующие подкаталоги конфигурационные файлы HTTP-сервера:

```
cd /vagrant/provision/server
mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
mkdir -p /vagrant/provision/server/http/var/www/html
cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
```

2. Заменяю конфигурационные файлы DNS-сервера:

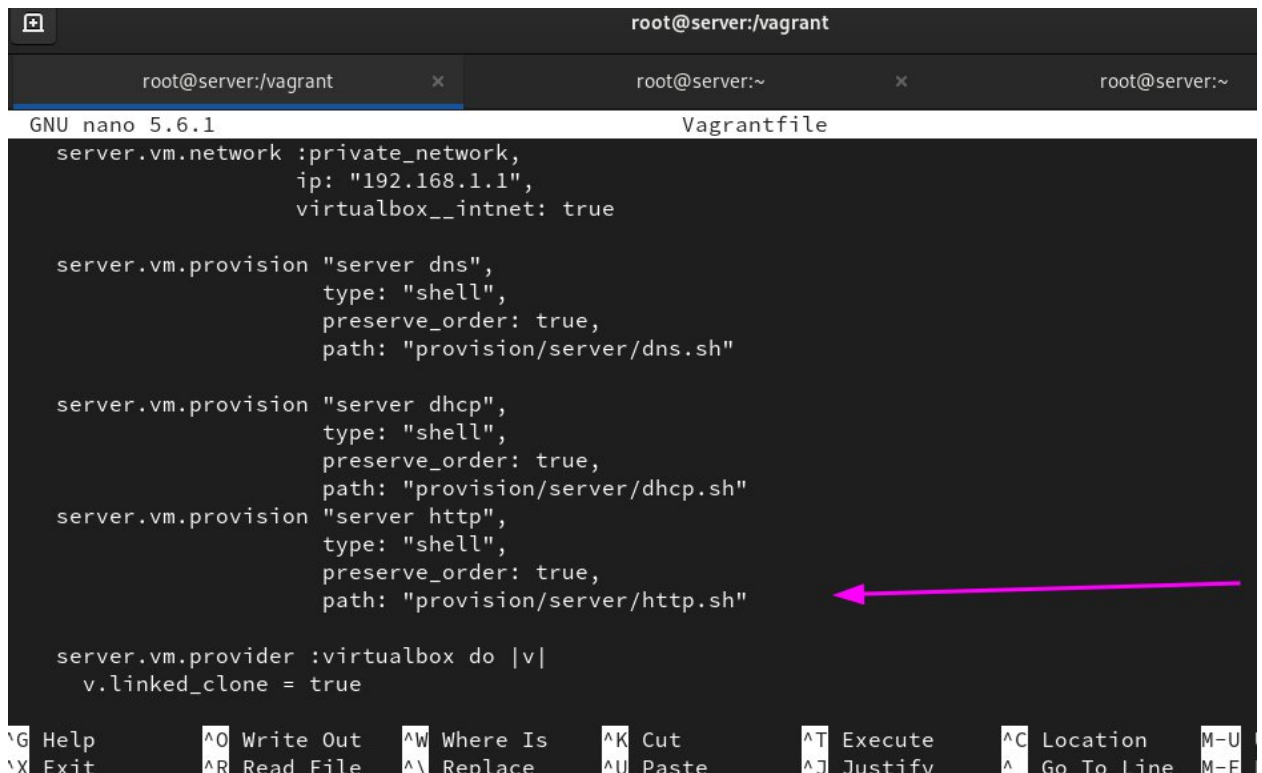
```
cd /vagrant/dns/
cp -R /var/named/* /vagrant/dns/var/named/
```

3. В каталоге /vagrant/provision/server создаю исполняемый файл http.sh:

```
cd /vagrant/provision/server
touch http.sh
chmod +x http.sh
```

4. Для отработки созданного скрипта во время загрузки виртуальных машин в конфигурационном файле Vagrantfile необходимо добавить в конфигурации сервера следующую запись:

```
server.vm.provision "server http",
type: "shell",
preserve_order: true,
path: "provision/server/http.sh"
```



```
root@server:/vagrant
GNU nano 5.6.1 Vagrantfile
server.vm.network :private_network,
                  ip: "192.168.1.1",
                  virtualbox____intnet: true

server.vm.provision "server dns",
                  type: "shell",
                  preserve_order: true,
                  path: "provision/server/dns.sh"

server.vm.provision "server dhcp",
                  type: "shell",
                  preserve_order: true,
                  path: "provision/server/dhcp.sh"

server.vm.provision "server http",
                  type: "shell",
                  preserve_order: true,
                  path: "provision/server/http.sh"

server.vm.provider :virtualbox do |v|
  v.linked_clone = true
end

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E
```

Вывод:

Я приобрел практические навыки по установке и базовому конфигурированию HTTP-сервера Apache.

Ответы на контрольные вопросы:

- 1. Через какой порт по умолчанию работает Apache?**
Порт 80 и порт 8000.
- 2. Под каким пользователем запускается Apache и к какой группе относится этот пользователь?**
По соображениям безопасности при запуске сервера Apache от имени суперпользователя (напрямую или через скрипт инициализации) происходит смена идентификатора пользователя (UID), от имени которого выполняется процесс сервера. По умолчанию используется пользователь http, который не имеет привилегированных полномочий в системе. Группа – http.
Пользователь Apache.
- 3. Где располагаются лог-файлы веб-сервера? Что можно по ним отслеживать?**
Лог-файлы хранятся в файле access.log.
Лог-файлы используются системными администраторами для анализа посетителей, изучения закономерности поведения определенных групп пользователей, а также получения различной информации о них, такой как: используемый браузер, IP адрес, данные о географическом положении клиента и многое другое. По мимо анализа, таким образом можно узнать о несанкционированных доступах к сайту, точнее узнать, кем именно он был произведен.

#Добавит путь

4. Где по умолчанию содержится контент веб-серверов?

В каталоге `/var/www/html` находятся файлы с содержимым (контентом) веб-серверов.

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

Термин виртуальный хост относится к практике запуска нескольких веб-сайтов (таких как `company1.example.com` и `company2.example.com`) на одной машине. Виртуальный хост может быть "IP-based", это означает, что у вас различные IP-адреса для каждого веб-сайта, или "name-based", это означает, что у вас есть несколько имен, работающих на каждом IP-адресе.

«IP-based»: Виртуальный хостинг на основе IP-адресов-это способ применения различных директив на основе IP-адреса и порта. Обычно, это используется для обслуживания различных веб-сайтов на разных портах или интерфейсах.

Во многих случаях хостинг, основанный на имени виртуальных хостов более удобен, ведь он позволяют нескольким виртуальным хостам совместно использовать один адрес / порт.

«name-based»: При использовании виртуального хостинга на основе имен сервер полагается на имя узла как часть заголовков HTTP. Используя этот виртуальный механизм, много разных хостов могут использовать один и тот же IP-адрес.

#Добавить как реализуется виртуальный хостинг.

Виртуальный хостинг на основе имен обычно проще, так как необходимо только настроить свой DNS-сервер, чтобы сопоставить каждое имя хоста с правильным IP-адресом, а затем настроить Apache HTTP Server для распознавания различных имен хостов. Именной виртуальный хостинг также облегчает спрос на дефицитные IP-адреса.

Виртуальный хостинг на основе имен строится на основе виртуального хоста на основе IP алгоритма выбора, то есть поиск нужного имени сервера происходит только между виртуальными узлами, имеющими лучший IP-адрес.