

# Лабораторная работа №4

Администрирование сетевых подсистем

---

Иванов Сергей Владимирович, НПИбд-01-23

23 сентября 2025

Российский университет дружбы народов, Москва, Россия

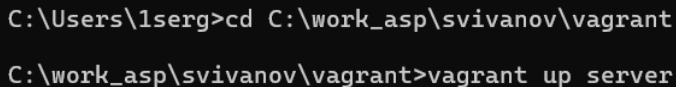
Целью данной работы является приобретение практических навыков по установке и базовому конфигурированию HTTP-сервера Apache.

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).

## **Выполнение работы**

---

Загрузим операционную систему и перейдем в рабочий каталог с проектом. Запустим виртуальную машину server. (рис. 1).



```
C:\Users\lserg>cd C:\work_asp\svivanov\vagrant  
C:\work_asp\svivanov\vagrant>vagrant up server
```

**Рис. 1:** Запуск server

На виртуальной машине server откроем терминал. Перейдем в режим суперпользователя. Установим http (рис. 2).

```
[root@server.svivanov.net ~]# dnf -y groupinstall "Basic web server"
Last metadata expiration check: 0:01:12 ago on Пн 22 сен 2025 08:14:39.
Dependencies resolved.
=====
Package                        Architecture      Version           Repository        Size
=====
Installing group/module packages:
httpd                          x86_64            2.4.63-1.el10_0.2 appstream          52 k
httpd-manual                   noarch            2.4.63-1.el10_0.2 appstream         2.3 M
mod_fcgid                      x86_64            2.3.9-35.el10    appstream          72 k
```

Рис. 2: Установка http

Посмотрим содержание конфигурационных файлов в каталогах `/etc/httpd/conf` и `/etc/httpd/conf.d`.

# Базовое конфигурирование HTTP-сервера

## Файл httpd.conf (рис. 3)

```
ServerRoot "/etc/httpd"

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on a specific IP address, but note that if
# httpd.service is enabled to run at boot time, the address may not be
# available when the service starts. See the httpd.service(8) man
# page for more information.
#
#Listen 12.34.56.78:80
Listen 80

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding 'LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Statically compiled modules (those listed by 'httpd -l') do not need
# to be loaded here.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
Include conf.modules.d/*.conf

#
# If you wish httpd to run as a different user or group, you must run
# httpd as root initially and it will switch.
#
# User/Group: The name (or #number) of the user/group to run httpd as.
```



# Базовое конфигурирование HTTP-сервера

## Файл autoindex.conf (рис. 4)

```
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.
#
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip

AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*
AddIconByType /icons/bomb.gif application/x-coredump

AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrm .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
```

Файл `fcgid.conf` (рис. 5)

```
# This is the Apache server configuration file for providing FastCGI support
# through mod_fcgid
#
# Documentation is available at
# http://httpd.apache.org/mod_fcgid/mod/mod_fcgid.html
#
# Use FastCGI to process .fcg .fcgi & .fpl scripts
AddHandler fcgid-script fcg fcgi fpl
#
# Sane place to put sockets and shared memory file
FcgidIPCDir /run/mod_fcgid
FcgidProcessTableFile /run/mod_fcgid/fcgid_shm
~
```

Рис. 5: Файл `fcgid.conf`

# Базовое конфигурирование HTTP-сервера

Файл manual.conf (рис. 6)

```
#
# This configuration file allows the manual to be accessed at
# http://localhost/manual/
#
Alias /manual /usr/share/httpd/manual

<Directory "/usr/share/httpd/manual">
    Options Indexes
    AllowOverride None
    Require all granted

    RedirectMatch 301 ^/manual/(? :da|de|en|es|fr|ja|ko|pt-br|ru|tr|zh-cn)(/.*)$ "/manual$1"
</Directory>
```

Рис. 6: Файл manual.conf

# Базовое конфигурирование HTTP-сервера

## Файл ssl.conf (рис. 7)

```
Listen 443 https

##
##  SSL Global Context
##
##  All SSL configuration in this context applies both to
##  the main server and all SSL-enabled virtual hosts.
##

#   Pass Phrase Dialog:
#   Configure the pass phrase gathering process.
#   The filtering dialog program ('builtin' is a internal
#   terminal dialog) has to provide the pass phrase on stdout.
SSLPassPhraseDialog exec:/usr/libexec/httpd-ssl-pass-dialog

#   Inter-Process Session Cache:
#   Configure the SSL Session Cache: First the mechanism
#   to use and second the expiring timeout (in seconds).
SSLSessionCache      shmcb:/run/httpd/sslcache(512000)
SSLSessionCacheTimeout 300
```

Рис. 7: Файл ssl.conf

# Базовое конфигурирование HTTP-сервера

## Файл userdir.conf (рис. 8)

```
<IfModule mod_userdir.c>
#
# UserDir is disabled by default since it can confirm the presence
# of a username on the system (depending on home directory
# permissions).
#
#
# To enable requests to /~user/ to serve the user's public_html
# directory, remove the "UserDir disabled" line above, and uncomment
# the following line instead:
#
#UserDir public_html
</IfModule>

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
<Directory "/home/*/public_html">
    AllowOverride FileInfo AuthConfig Limit Indexes
    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    Require method GET POST OPTIONS
</Directory>
```

Рис. 8: Файл userdir.conf

## Файл welcome.conf (рис. 9)

```
# This configuration file enables the default "Welcome" page if there
# is no default index page present for the root URL. To disable the
# Welcome page, comment out all the lines below.
#
# NOTE: if this file is removed, it will be restored on upgrades.
#
<LocationMatch "^/+>">
    Options -Indexes
    ErrorDocument 403 /.noindex.html
</LocationMatch>

<Directory /usr/share/httpd/noindex>
    AllowOverride None
    Require all granted
</Directory>

Alias /.noindex.html /usr/share/httpd/noindex/index.html
Alias /poweredby.png /usr/share/httpd/icons/apache_pb3.png
Alias /system_noindex_logo.png /usr/share/httpd/icons/system_noindex_logo.png
~
```

Рис. 9: Редактирование файлов

Внесем изменения в настройки межсетевого экрана узла server, разрешив работу с http: (рис. 10)

```
[root@server.svivanov.net conf.d]# firewall-cmd --add-service=http  
success  
[root@server.svivanov.net conf.d]# firewall-cmd --add-service=http --permanent  
success  
[root@server.svivanov.net conf.d]#
```

**Рис. 10:** Изменения в настройках firewall

# Конфигурирование ДНСП-сервера

В дополнительном терминале запустим расширенный лог сообщений, чтобы проверить корректность работы системы. В первом терминале запустим НТТР-сервер. Просмотрев расширенный лог сообщений, убедимся, что веб-сервер успешно запустился (рис. 11)

```
Начат процесс запуска юнита httpd.service.  
сен 22 08:23:13 server.svivanov.net (httpd)[14866]: httpd.service: Referenced but unset environment va  
сен 22 08:23:13 server.svivanov.net httpd[14866]: Server configured, listening on: port 443, port 80  
сен 22 08:23:13 server.svivanov.net systemd[1]: Started httpd.service - The Apache HTTP Server.  
Subject: Запуск юнита httpd.service завершен  
Defined-By: systemd  
Support: https://wiki.rockylinux.org/rocky/support  
  
Процесс запуска юнита httpd.service был завершен.  
  
Результат: done.
```

**Рис. 11:** Запуск лога сообщений и запуск веб-сервера



Запустим виртуальную машину client. (рис. 12)

```
C:\work_asp\svivanov\vagrant>vagrant up client  
Bringing machine 'client' up with 'virtualbox' provider  
==> client: Clearing any previously set forwarded ports  
==> client: Fixed port collision for 22 => 2222. Now on
```

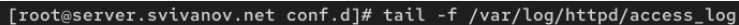
**Рис. 12:** Запуск client

На виртуальной машине server посмотрим лог ошибок работы веб-сервера (рис. 13)

```
[root@server.svivanov.net conf.d]# tail -f /var/log/httpd/error_log
[Mon Sep 22 08:23:13.109516 2025] [suexec:notice] [pid 14866:tid 14866] AH01232: suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
[Mon Sep 22 08:23:13.125339 2025] [lbmethod_heartbeat:notice] [pid 14866:tid 14866] AH02282: No slotmem from mod_heartbeat
[Mon Sep 22 08:23:13.127569 2025] [systemd:notice] [pid 14866:tid 14866] SELinux policy enabled; httpd running as context system_u:system_r:httpd_t:s0
[Mon Sep 22 08:23:13.129698 2025] [mpm_event:notice] [pid 14866:tid 14866] AH00489: Apache/2.4.63 (Rocky Linux) OpenSSL/3.2.2 mod_fcgid/2.3.9 configured -- resuming normal operations
[Mon Sep 22 08:23:13.129720 2025] [core:notice] [pid 14866:tid 14866] AH00094: Command line: '/usr/sbin/httpd -D FOREGROUND'
```

**Рис. 13:** Запуск лога ошибок

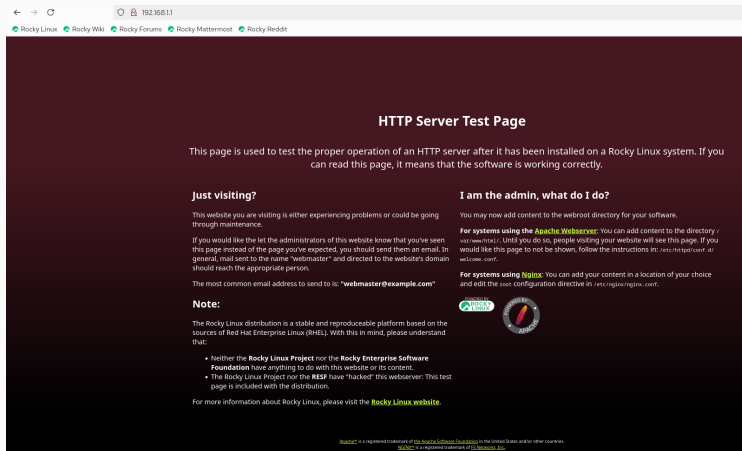
На виртуальной машине server запустим мониторинг доступа к веб-серверу (рис. 14)

A terminal window with a dark background and light gray text. The prompt is [root@server.svivanov.net conf.d]# and the command being entered is tail -f /var/log/httpd/access\_log. A white cursor is visible at the end of the command.

```
[root@server.svivanov.net conf.d]# tail -f /var/log/httpd/access_log
```

**Рис. 14:** Запуск мониторинга доступа

На виртуальной машине client запустим браузер и в адресной строке введем 192.168.1.1 (рис. 15)



Просмотрим информацию, отобразившуюся при мониторинге (рис. 16)

```
[root@server.svivanov.net conf.d]# tail -f /var/log/httpd/access_log
192.168.1.30 - - [22/Sep/2025:08:52:09 +0000] "GET / HTTP/1.1" 403 7620 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.1.30 - - [22/Sep/2025:08:52:10 +0000] "GET /icons/poweredby.png HTTP/1.1" 200 15443 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.1.30 - - [22/Sep/2025:08:52:10 +0000] "GET /poweredby.png HTTP/1.1" 200 5714 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.1.30 - - [22/Sep/2025:08:52:10 +0000] "GET /favicon.ico HTTP/1.1" 404 196 "http://192.168.1.1/" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
192.168.1.30 - - [22/Sep/2025:08:52:48 +0000] "GET / HTTP/1.1" 403 7620 "-" "Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0"
```

**Рис. 16:** Мониторинг

# Настройка виртуального хостинга для HTTP-сервера

Остановим работу DNS-сервера для внесения изменений в файлы описания DNSзон. Добавим запись для HTTP-сервера в конце файла прямой DNS-зоны (рис. 17)

```
$TTL 1D
@      IN SOA  a server.svivanov.net. (
                                2025091500      ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )  ; minimum
      NS   a
      A    192.168.1.1
$ORIGIN svivanov.net.
server  A    192.168.1.1
ns      A    192.168.1.1
dhcp    A    192.168.1.1
www     A    192.168.1.1
```

Рис. 17: Добавление записи

# Настройка виртуального хостинга для HTTP-сервера

И в конце файла обратной зоны (рис. 18)

```
$TTL 1D
@      IN SOA  @ server.svivanov.net. (
                                2025091500      ; serial
                                1D      ; refresh
                                1H      ; retry
                                1W      ; expire
                                3H )    ; minimum

      NS      @
      A       192.168.1.1
      PTR     server.svivanov.net.
$ORIGIN 1.168.192.in-addr.arpa.
1      PTR     server.svivanov.net.
1      PTR     ns.svivanov.net.
1      PTR     dhcp.svivanov.net.
1      PTR     www.svivanov.net.
```

Рис. 18: Добавление записи

# Настройка виртуального хостинга для HTTP-сервера

Перезапустим DNS-сервер. В каталоге `/etc/httpd/conf.d` создадим файлы `server.user.net.conf` и `www.user.net.conf`: (рис. 19)

```
[root@server.svivanov.net conf.d]# systemctl start named
[root@server.svivanov.net conf.d]# cd /etc/httpd/conf.d
[root@server.svivanov.net conf.d]# touch server.svivanov.net.conf
[root@server.svivanov.net conf.d]# touch www.svivanov.net.conf
[root@server.svivanov.net conf.d]#
```

**Рис. 19:** Перезапуск сервера и создание файлов



# Настройка виртуального хостинга для HTTP-сервера

Откроем на редактирование файл `server.svivanov.net.conf` и внесем следующее содержание: (рис. 20)

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

Рис. 20: Редактирование конф. файла

# Настройка виртуального хостинга для HTTP-сервера

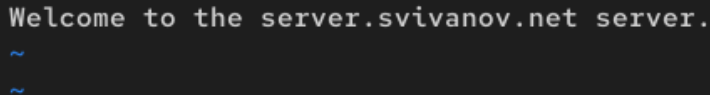
Откроем на редактирование файл `www.svivanov.net.conf` и внесем следующее содержание: (рис. 21)

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

Рис. 21: Редактирование конф. файла

# Настройка виртуального хостинга для HTTP-сервера

Откроем на редактирование файл `index.html` и внесем следующее содержание (рис. 22)

A screenshot of a code editor with a dark background. The text 'Welcome to the server.svivanov.net server.' is written in a light gray monospace font. Below this line, there are two lines, each starting with a blue tilde '~' character, indicating continuation or a new line in the code.

```
Welcome to the server.svivanov.net server.  
~  
~
```

**Рис. 22:** Редактирование `index.html`

Для виртуального веб-сервера `www.svivanov.net`: (рис. 23)

```
[root@server.svivanov.net server.svivanov.net]# cd /var/www/html
[root@server.svivanov.net html]# mkdir www.svivanov.net
[root@server.svivanov.net html]# cd /var/www/html/www.svivanov.net
[root@server.svivanov.net www.svivanov.net]# touch index.html
[root@server.svivanov.net www.svivanov.net]#
```

**Рис. 23:** Создание файлов и папок

## Настройка виртуального хостинга для HTTP-сервера

Откроем на редактирование файл `index.html` и внесем следующее содержание: `Welcome to the www.user.net server.` (рис. 24)

A screenshot of a code editor with a dark background. The text 'Welcome to the www.svivanov.net server.' is written in a light gray monospace font. Below the text, there are two blue tilde characters (~) on separate lines, indicating the end of the file.

```
Welcome to the www.svivanov.net server.  
~  
~
```

**Рис. 24:** Редактирование `index.html`

# Настройка виртуального хостинга для HTTP-сервера

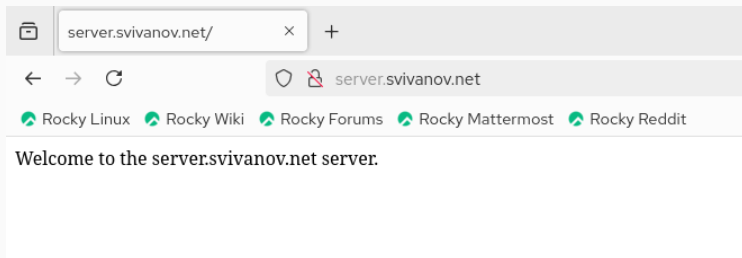
Скорректируем права доступа в каталог с веб-контентом. Восстановим контекст безопасности в SELinux. Перезапустим HTTP-сервер: (рис. 25)

```
[root@server.svivanov.net www.svivanov.net]# chown -R apache:apache /var/www
[root@server.svivanov.net www.svivanov.net]# restorecon -vR /etc
Relabeled /etc/NetworkManager/system-connections/eth1.nmconnection from unconfined_u:
confined_u:object_r:NetworkManager_etc_rw_t:s0
[root@server.svivanov.net www.svivanov.net]# restorecon -vR /var/named
[root@server.svivanov.net www.svivanov.net]# restorecon -vR /var/www
[root@server.svivanov.net www.svivanov.net]# systemctl restart httpd
[root@server.svivanov.net www.svivanov.net]#
```

**Рис. 25:** Восстановление меток и перезапуск сервера

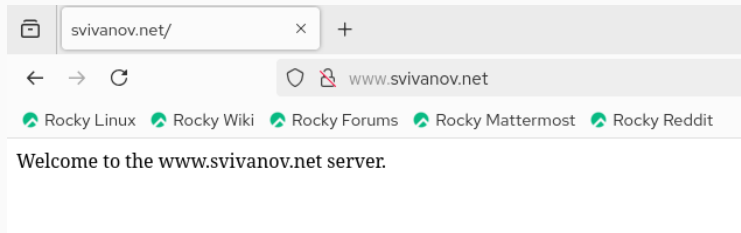
# Настройка виртуального хостинга для HTTP-сервера

На виртуальной машине client убедимся в корректном доступе к веб-серверу по адресам `server.svivanov.net` и `www.svivanov.net` в адресной строке веб-браузера. (рис. 26, 27)



**Рис. 26:** Доступ к веб-серверу в адресной строке

# Настройка виртуального хостинга для HTTP-сервера



**Рис. 27:** Доступ к веб-серверу в адресной строке



# Внесение изменений в настройки внутреннего окружения виртуальной машины

На виртуальной машине `server` перейдем в каталог для внесения изменений в настройки внутреннего окружения `/vagrant/provision/server/`, создадим в нём каталог `http`, в который поместим в соответствующие подкаталоги конфигурационные файлы HTTP-сервера (рис. 28)

```
[root@server.svivanov.net server]# mkdir -p /vagrant/provision/server/http/etc/httpd/conf.d
[root@server.svivanov.net server]# mkdir -p /vagrant/provision/server/http/var/www/html
[root@server.svivanov.net server]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/httpd/conf.d/
[root@server.svivanov.net server]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html
[root@server.svivanov.net server]#
```

**Рис. 28:** Создание каталогов и копирование файлов

# Внесение изменений в настройки внутреннего окружения виртуальной машины

Заменяем конфигурационные файлы DNS-сервера (рис. 29)

```
[root@server.svivanov.net server]# cd /vagrant/provision/server/dns/
[root@server.svivanov.net dns]# cp -R /var/named/* /vagrant/provision/server/dns/var/named/
cp: overwrite '/vagrant/provision/server/dns/var/named/data/named.run'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/dynamic/managed-keys.bind.jnl'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/master/rz/192.168.1'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/master/fz/svivanov.net'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.ca'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.empty'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.localhost'? yes
cp: overwrite '/vagrant/provision/server/dns/var/named/named.loopback'? yes
[root@server.svivanov.net dns]#
```

**Рис. 29:** Замена конф. файлов

# Внесение изменений в настройки внутреннего окружения виртуальной машины

В каталоге `/vagrant/provision/server` создадим исполняемый файл `http.sh`.  
Открыв его на редактирование, пропишем в нём скрипт: (рис. 30)

```
#!/bin/bash
echo "Provisioning script $0"
echo "Install needed packages"
dnf -y groupinstall "Basic Web Server"
echo "Copy configuration files"
cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
cp -R /vagrant/provision/server/http/var/www/* /var/www
chown -R apache:apache /var/www
restorecon -vR /etc
restorecon -vR /var/www
echo "Configure firewall"
firewall-cmd --add-service=http
firewall-cmd --add-service=http --permanent
echo "Start http service"
systemctl enable httpd
systemctl start httpd
```

Рис. 30: Создание скрипта

## Внесение изменений в настройки внутреннего окружения виртуальной машины

Для отработки созданного скрипта во время загрузки виртуальных машин в файле Vagrantfile добавим в конфигурации сервера следующую запись: (рис. 31)

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

Рис. 31: Редактирование Vagrantfile

## Вывод

---

В ходе выполнения лабораторной работы мы приобрели практически навыки по установке и базовому конфигурированию HTTP-сервера Apache.