

Лабораторная работа №5

Администрирование сетевых подсистем

Машков Илья Евгеньевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Конфигурирование HTTP-сервера для работы через протокол HTTPS	7
3.2	Конфигурирование HTTP-сервера для работы с РНР	11
3.3	Внесение изменений в настройки внутреннего окружения виртуальной машины	13
4	Выводы	16
	Список литературы	17

Список иллюстраций

3.1	Генерация ключа и сертификата	7
3.2	Копирование сертификата в другой каталог	8
3.3	Изменения в файле <code>www.user.net.conf</code>	9
3.4	Настройка межсетевого экрана	10
3.5	Перезапуск HTTP-сервера	10
3.6	Открытие адреса <code>www.user.net</code>	11
3.7	Установка пакетов для PHP	12
3.8	<code>index.php</code>	12
3.9	Настройка http-сервера на работу с php	13
3.10	Неудачное открытие адреса <code>www.user.net</code>	13
3.11	Коррекция настроек внутреннего окружения	14
3.12	<code>http.sh</code>	15

Список таблиц

1 Цель работы

Приобретение практических навыков по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

2 Задание

1. Сгенерируйте криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS.
2. Настройте веб-сервер для работы с PHP.
3. Напишите (или скорректируйте) скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины `server`.

укажите RU; - в строке названия страны укажите Russia; - в строке названия города укажите Moscow; - в строке названия организации укажите свой логин; - в строке названия подразделения укажите свой логин; - в строке названия хоста должно быть указано доменное имя вашего веб-сервера user.net; - в строке e-mail адреса должен быть указан user@user.net.

Затем копирую сертификат в каталог /etc/ssl/certs (рис. [3.2]).

```
[root@server private]# mv www.user.net.crt /etc/pki/tls/certs  
[root@server private]#
```

Рис. 3.2: Копирование сертификата в другой каталог

Затем в конфигурационный файл www.user.net.conf(доменное имя которого у меня не работало ещё в прошлой лабе) вношу строки, которые позволяют активировать протокол HTTPS, т.к. без сертификата безопасности этого сделать не получится. По сути в этих строках мы предоставляем ключ и сертификат безопасности для активации нужного нам протокола (рис. [3.3]).


```
GNU nano 5.6.1 /etc/httpd/conf.d/www.user.net.conf
<VirtualHost *:80>
  ServerAdmin webmaster@user.net
  DocumentRoot /var/www/html/www.user.net
  ServerName www.user.net
  ServerAlias www.user.net
  ErrorLog logs/www.user.net-error_log
  CustomLog logs/www.user.net-access_log common
  RewriteEngine on
  RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [R=301,L]
</VirtualHost>

<IfModule mod_ssl.c>
<VirtualHost *:443>
  SSLEngine on
  ServerAdmin webmaster@user.net
  DocumentRoot /var/www/html/www.user.net
  ServerName www.user.net
  ServerAlias www.user.net
  ErrorLog logs/www.user.net-error_log
  CustomLog logs/www.user.net-access_log common
  SSLCertificateFile /etc/ssl/certs/www.user.net.crt
  SSLCertificateKeyFile /etc/ssl/private/www.user.net.key
</VirtualHost>
</IfModule>
```

Рис. 3.3: Изменения в файле www.user.net.conf

Вношу изменения в межсетевой экран, тем самым разрешая работу с https (рис. [3.4]).

```
[root@server conf.d]# firewall-cmd --list-services
cockpit dhcp dhcpv6-client dns http ssh
[root@server conf.d]# firewall-cmd --get-services
RH-Satellite-6 RH-Satellite-6-capsule afp amanda-client amanda-k5-client amqp
amqps apcupsd audit ausweisapp2 bacula bacula-client bareos-director bareos-fi
ledaemon bareos-storage bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-tes
tnet-rpc bittorrent-lsd ceph ceph-exporter ceph-mon cfengine checkmk-agent coc
kpit collectd condor-collector cratedb ctdb dds dds-multicast dds-unicast dhcp
dhcpv6 dhcpv6-client distcc dns dns-over-tls docker-registry docker-swarm dro
pbox-lansync elasticsearch etcd-client etcd-server finger foreman foreman-prox
y freeipa-4 freeipa-ldap freeipa-ldaps freeipa-replication freeipa-trust ftp g
alera ganglia-client ganglia-master git gpsd grafana gre high-availability htt
p http3 https ident imap imaps ipfs ipp ipp-client ipsec irc ircs iscsi-target
isns jenkins kadmin kdeconnect kerberos kibana klogin kpasswd kprop kshell ku
be-api kube-apiserver kube-control-plane kube-control-plane-secure kube-contro
ller-manager kube-controller-manager-secure kube-nodeport-services kube-schedu
ler kube-scheduler-secure kube-worker kubelet kubelet-readonly kubelet-worker
ldap ldaps libvirt libvirt-tls lightning-network llmnr llmnr-client llmnr-tcp
llmnr-udp managesieve matrix mdns memcache minidlna mongodb mosh mountd mqtt m
qtt-tls ms-wbt mssql murmur mysql nbd nebula netbios-ns netdata-dashboard nfs
nfs3 nmea-0183 nrpe ntp nut openvpn ovirt-imageio ovirt-storageconsole ovirt-v
mconsole plex pmcd pmproxy pmwebapi pmwebapis pop3 pop3s postgresql privoxy pr
ometheus prometheus-node-exporter proxy-dhcp ps2link ps3netsrv ptp pulseaudio
puppetmaster quassel radius rdp redis redis-sentinel rpc-bind rquotad rsh rsyn
cd rtsp salt-master samba samba-client samba-dc sane sip sips slp smtp smtp-su
bmission smtps snmp snmptls snmptls-trap snmptrap spideroak-lansync spotify-sy
nc squid ssdp ssh steam-streaming svdrp svn syncthing syncthing-gui syncthing-
relay synergy syslog syslog-tls telnet tentacle tftp tile38 tinc tor-socks tra
nsmision-client upnp-client vdsm vnc-server warpinator wbem-http wbem-https w
ireguard ws-discovery ws-discovery-client ws-discovery-tcp ws-discovery-udp ws
man wsmans xdmcp xmpp-bosh xmpp-client xmpp-local xmpp-server zabbix-agent zab
bix-server zerotier
[root@server conf.d]# firewall-cmd --add-service=https
success
[root@server conf.d]# firewall-cmd --add-service=https --permanent
success
[root@server conf.d]# firewall-cmd --reload
success
[root@server conf.d]#
```

Рис. 3.4: Настройка межсетевого экрана

Перезапускаю HTTP-сервер (рис. [3.5]).

```
[root@server conf.d]# systemctl restart httpd
[root@server conf.d]#
```

Рис. 3.5: Перезапуск HTTP-сервера

Т.к. и в прошлой лабе этот адрес у меня не работал, то и в этой лабе, ожидаемо, он тоже работать не будет, но в url мы должны были увидеть, что мы используем протокол HTTPS (рис. [3.6]).

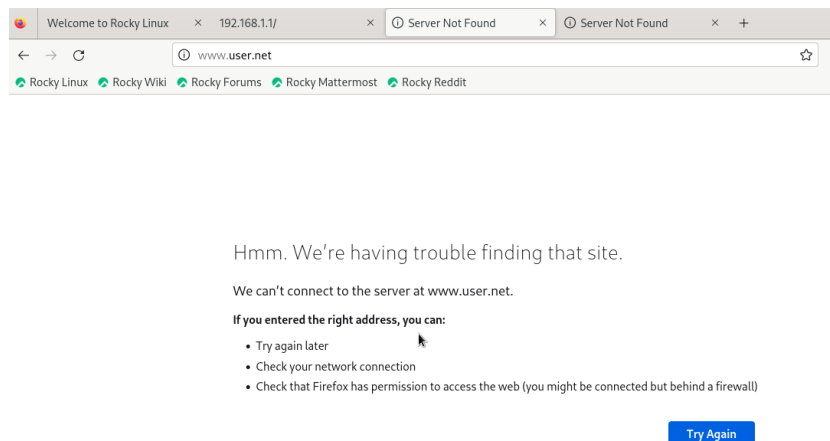


Рис. 3.6: Открытие адреса www.user.net

3.2 Конфигурирование HTTP-сервера для работы с PHP

Устанавливаю пакеты для PHP (рис. [3.7]).

```
[root@server conf.d]# dnf -y install php
Last metadata expiration check: 1:22:52 ago on Tue 02 Dec 2025 04:36:32 PM MSK
.
Dependencies resolved.
=====
Package                Arch      Version              Repository           Size
=====
Installing:
php                    x86_64    8.0.30-3.el9_6      appstream            8.2 k
Installing dependencies:
nginx-filessystem      noarch    2:1.20.1-24.el9     appstream            9.2 k
php-common             x86_64    8.0.30-3.el9_6      appstream            666 k
Installing weak dependencies:
php-cli               x86_64    8.0.30-3.el9_6      appstream            3.1 M
php-fpm               x86_64    8.0.30-3.el9_6      appstream            1.6 M
php-mbstring          x86_64    8.0.30-3.el9_6      appstream            466 k
php-opcache            x86_64    8.0.30-3.el9_6      appstream            510 k
php-pdo                x86_64    8.0.30-3.el9_6      appstream            81 k
php-xml                x86_64    8.0.30-3.el9_6      appstream            132 k
=====
Transaction Summary
=====
Install 9 Packages

Total download size: 6.5 M
Installed size: 35 M
Downloading Packages:
(1/9): nginx-filessystem-1.20.1-24.el9.noarch. 7.0 kB/s | 9.2 kB    00:01
(2/9): php-8.0.30-3.el9_6.x86_64.rpm         6.2 kB/s | 8.2 kB    00:01
(3/9): php-fpm-8.0.30-3.el9_6.x86_64.rpm      2.6 MB/s | 1.6 MB    00:00
(4/9): php-mbstring-8.0.30-3.el9_6.x86_64.rpm 1.8 MB/s | 466 kB    00:00
(5/9): php-common-8.0.30-3.el9_6.x86_64.rpm   742 kB/s | 666 kB    00:00
(6/9): php-opcache-8.0.30-3.el9_6.x86_64.rpm 4.0 MB/s | 510 kB    00:00
(7/9): php-pdo-8.0.30-3.el9_6.x86_64.rpm     650 kB/s | 81 kB     00:00
(8/9): php-xml-8.0.30-3.el9_6.x86_64.rpm     1.3 MB/s | 132 kB    00:00
```

Рис. 3.7: Установка пакетов для PHP

В каталоге `www.user.net` меняю файл `index.html` на `index.php` (рис. [3.8]).

```
GNU nano 5.6.1 index.php
<?php
phpinfo();
?>
```

Рис. 3.8: `index.php`

Затем корректирую права доступа, восстанавливаю контекст безопасности и перезапускаю `http`-сервер (рис. [3.9]).

```
[root@server www.user.net]# chown -R apache:apache /var/www
[root@server www.user.net]# 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
[root@server www.user.net]# restorecon -vR /var/www
[root@server www.user.net]# systemctl restart httpd
[root@server www.user.net]#
```

Рис. 3.9: Настройка http-сервера на работу с php

Т.к. оно не работало и до этого, то и здесь оно тоже не работает, а должно было вывести информация об используемой на этом веб-сервере версии PHP (рис. [3.10]).

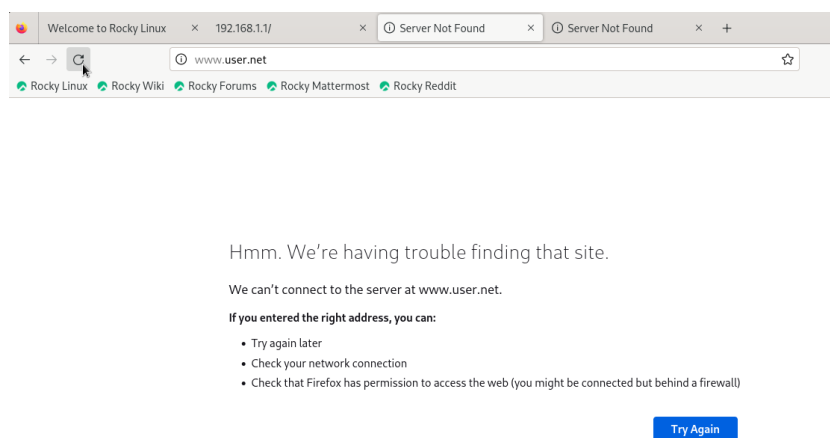


Рис. 3.10: Неудачное открытие адреса www.user.net

3.3 Внесение изменений в настройки внутреннего окружения виртуальной машины

Вношу изменения в настройки внутреннего окружения машины server путём добавления конфигурационных файлов, которые мы создали или изменяли за время выполнения этой лабораторной (рис. [3.11]).

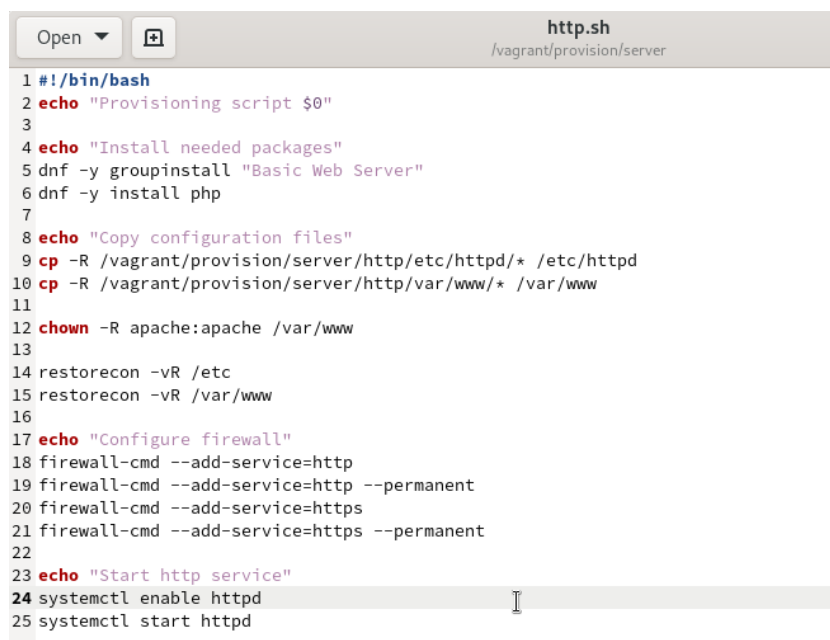
```

[root@server ~]# cp -R /etc/httpd/conf.d/* /vagrant/provision/server/http/etc/
httpd/conf.d
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/autoindex.conf'
? yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/fcgid.conf'? y
es
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/manual.conf'? y
es
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/README'? yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/server.user.net
.conf'? yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/ssl.conf'? yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/userdir.conf'?
yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/welcome.conf'?
yes
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/www.user.net.co
nf'? yes
[root@server ~]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/
html
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.user.net/ind
ex.html'? yes
[root@server ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/private
[root@server ~]# mkdir -p /vagrant/provision/server/http/etc/pki/tls/certs
[root@server ~]# cp -R /etc/pki/tls/private/www.user.net.key /vagrant/provisio
n/server/http/etc/pki/tls/private
[root@server ~]# cp -R /etc/pki/tls/certs/www.user.net.crt
cp: missing destination file operand after '/etc/pki/tls/certs/www.user.net.cr
t'
Try 'cp --help' for more information.
[root@server ~]# cp -R /etc/pki/tls/certs/www.user.net.crt /vagrant/provision/
server/http/etc/pki/tls/certs
[root@server ~]#

```

Рис. 3.11: Коррекция настроек внутреннего окружения

Т.к. мы ничего глобального здесь не сделали, то нам всего лишь надо добавить установку РНР и настройку межсетевого экрана, позволяющую работать с https в скрипт http.sh (рис. [3.12]).

A screenshot of a terminal window with a title bar. The title bar contains an 'Open' button with a dropdown arrow, a file icon, and the text 'http.sh' and '/vagrant/provision/server'. The terminal displays a shell script with 25 lines. Lines 2, 4, 8, 17, 23, and 25 are highlighted in red. The script includes comments and commands for installing packages, copying files, setting permissions, restoring permissions, configuring the firewall, and starting the http service.

```
1 #!/bin/bash
2 echo "Provisioning script $0"
3
4 echo "Install needed packages"
5 dnf -y groupinstall "Basic Web Server"
6 dnf -y install php
7
8 echo "Copy configuration files"
9 cp -R /vagrant/provision/server/http/etc/httpd/* /etc/httpd
10 cp -R /vagrant/provision/server/http/var/www/* /var/www
11
12 chown -R apache:apache /var/www
13
14 restorecon -vR /etc
15 restorecon -vR /var/www
16
17 echo "Configure firewall"
18 firewall-cmd --add-service=http
19 firewall-cmd --add-service=http --permanent
20 firewall-cmd --add-service=https
21 firewall-cmd --add-service=https --permanent
22
23 echo "Start http service"
24 systemctl enable httpd
25 systemctl start httpd
```

Рис. 3.12: http.sh

4 Выводы

В процессе выполнения данной лабораторной работы я освоил практические навыки по расширенному конфигурированию HTTP-сервера Apache в части безопасности и возможности использования PHP.

Список литературы

Администрирование сетевых подсистем