

Лабораторная работа 5

Генералов Даниил, НПИбд-01-21, 1032202280

2023

¹RUDN University, Moscow, Russian Federation

Задача

1. Сгенерируйте криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS (см. раздел 5.4.1).

1. Сгенерируйте криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS (см. раздел 5.4.1).
2. Настройте веб-сервер для работы с PHP (см. раздел 5.4.2).

1. Сгенерируйте криптографический ключ и самоподписанный сертификат безопасности для возможности перехода веб-сервера от работы через протокол HTTP к работе через протокол HTTPS (см. раздел 5.4.1).
2. Настройте веб-сервер для работы с PHP (см. раздел 5.4.2).
3. Напишите (или скорректируйте) скрипт для Vagrant, фиксирующий действия по расширенной настройке HTTP-сервера во внутреннем окружении виртуальной машины server (см. раздел 5.4.3).

Выполнение

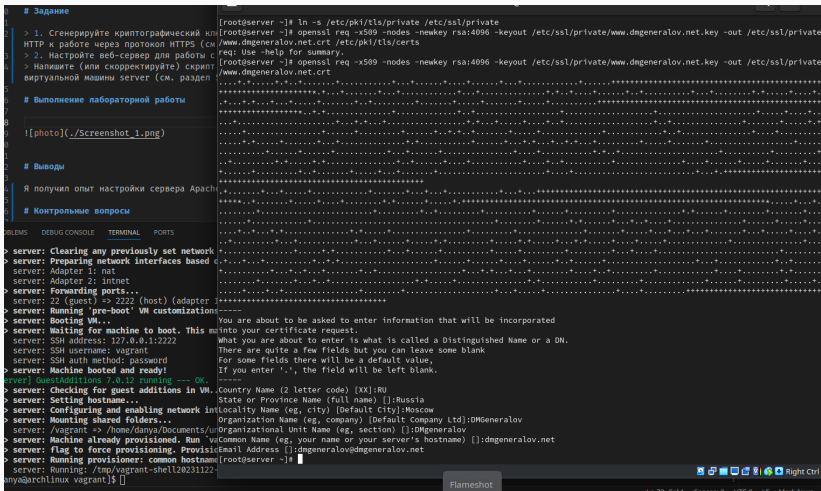


Рис. 1: openssl

```
GNU nano 5.6.1 /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
    ServerAlias www.dmgeneralov.net
    ErrorLog logs/www.dmgeneralov.net-error_log
    CustomLog logs/www.dmgeneralov.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@dmgeneralov.net
    DocumentRoot /var/www/html/www.dmgeneralov.net
    ServerName www.dmgeneralov.net
    ServerAlias www.dmgeneralov.net
    ErrorLog logs/www.dmgeneralov.net-error_log
    CustomLog logs/www.dmgeneralov.net-access_log common
    SSLCertificateFile /etc/ssl/certs/www.dmgeneralov.net.crt
    SSLCertificateKeyFile /etc/ssl/private/www.dmgeneralov.net.key
</VirtualHost>
</IfModule>
```



```
The unit httpd.service has entered the 'failed' state with result 'exit-code'.
Nov 22 19:38:01 server.user.net systemd[1]: Failed to start The Apache HTTP Server.
Subject: A start job for unit httpd.service has failed
Defined-By: systemd
Support: https://access.redhat.com/support

A start job for unit httpd.service has finished with a failure.

The job identifier is 2426 and the job result is failed.
[root@server ~]# cp /etc/pki/tls/
cert.pem      certs/      ct_log_list.cnf  fips_local.cnf  misc/      openssl.cnf      private/
[root@server ~]# cp /etc/pki/tls/private/www.dmgeneralov.net.crt /etc/pki/tls/certs/
[root@server ~]# firewall-cmd --add-service=https
Warning: ALREADY_ENABLED: 'https' already in 'public'
success
[root@server ~]# firewall-cmd --add-service=https --permanent
Warning: ALREADY_ENABLED: https
success
[root@server ~]# firewall-cmd --reload
success
[root@server ~]# systemctl restart httpd
[root@server ~]#
```

Рис. 3: apache

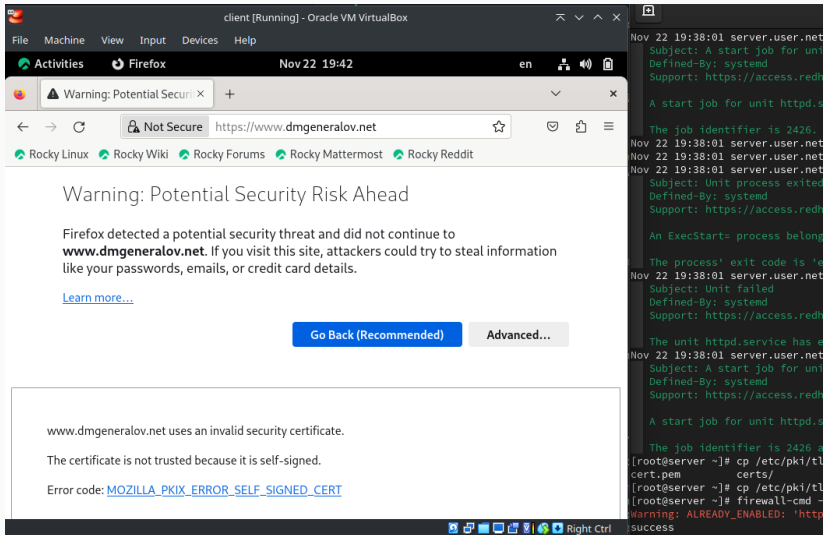
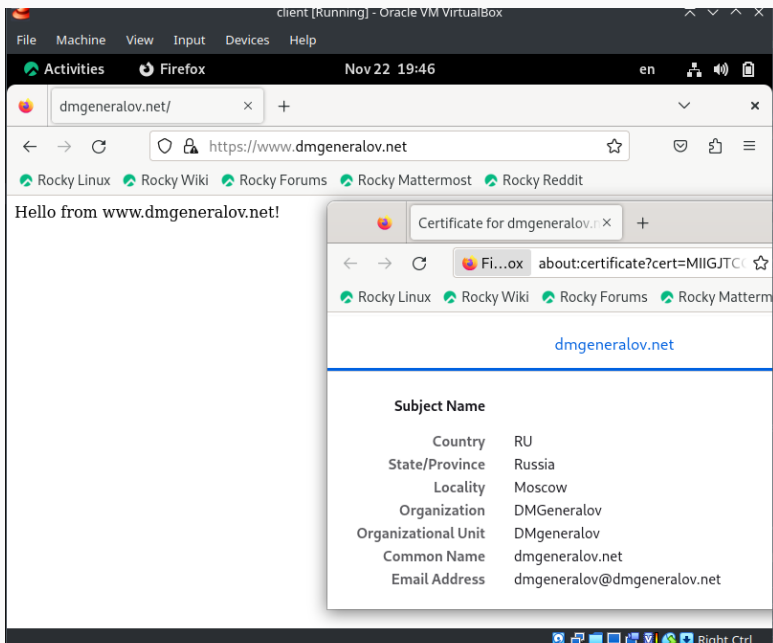


Рис. 4: firefox



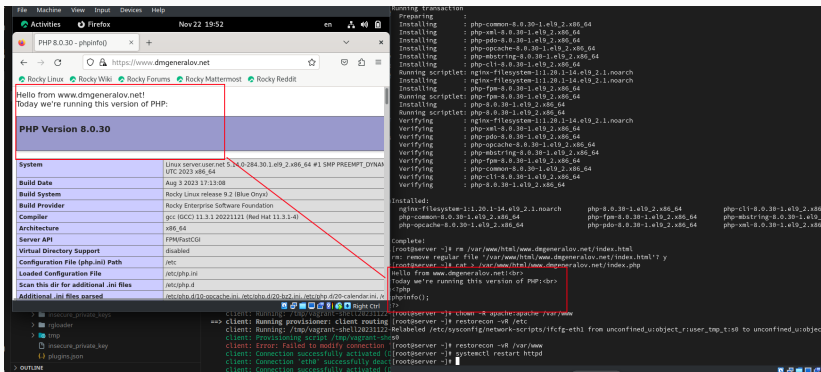
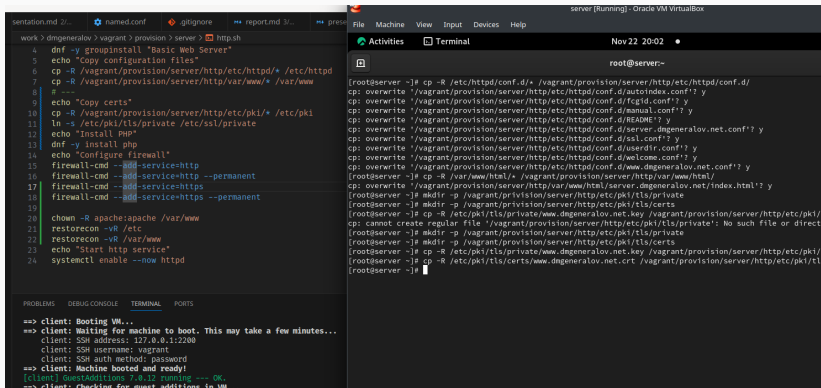


Рис. 6: firefox



```

sentation.md 2/... named.conf .gitignore report.md 3/... pres
work > dngeneralov > vagrant > provision > server > http.sh
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 # ---
9 echo "Copy certs"
10 cp -R /vagrant/provision/server/http/etc/pki/* /etc/pki
11 ln -s /etc/pki/tls/private /etc/ssl/private
12 echo "Install php"
13 dnf -y install php
14 echo "Configure firewall"
15 firewall-cmd --add-service=http
16 firewall-cmd --add-service=http --permanent
17 firewall-cmd --add-service=https
18 firewall-cmd --add-service=https --permanent
19
20 chown -R apache:apache /var/www
21 restorecon -vR /etc
22 restorecon -vR /var/www
23 echo "Start http service"
24 systemctl enable --now httpd

[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'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/fcgid.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/manual.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/README'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/server.dngeneralov.net.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/ssl.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/userdir.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/welcome.conf'? y
cp: overwrite '/vagrant/provision/server/http/etc/httpd/conf.d/www.dngeneralov.net.conf'? y
[root@server ~]# cp -R /var/www/html/* /vagrant/provision/server/http/var/www/html/
cp: overwrite '/vagrant/provision/server/http/var/www/html/server.dngeneralov.net/index.html'? y
[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.dngeneralov.net.key /vagrant/provision/server/http/etc/pki/
cp: cannot create regular file '/vagrant/provision/server/http/etc/pki/tls/private': No such file or direct
[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.dngeneralov.net.key /vagrant/provision/server/http/etc/pki/
[root@server ~]# cp -R /etc/pki/tls/certs/www.dngeneralov.net.crt /vagrant/provision/server/http/etc/pki/
[root@server ~]#
  
```

PROBLEMS DEBUG CONSOLE TERMINAL PORTS

```

==> client: Booting VM...
==> client: Waiting for machine to boot. This may take a few minutes...
client: SSH address: 127.0.0.1:2200
client: SSH username: vagrant
client: SSH auth method: password
==> client: Machine booted and ready!
[client] GuestAdditions 7.0.12 running --- OK.
==> client: Checking for guest additions in VM...
  
```

Рис. 7: vagrant

Я получил опыт настройки сервера Apache, чтобы он использовал HTTPS и PHP.