Разбор практического задания

Первое задание

Представьте, что вам нужно безопасно передать кому-то несколько текстовых файлов. Запакуйте эти файлы в tar.gz-архив и зашифруйте его с помощью симметричного шифрования.

Давайте создадим архив из нескольких файлов:

```
mkdir tmp
cd tmp
touch 1 2 3 4 5
tar -czvf files_for_Ivan.tar.gz directory
openssl enc -blowfish -in files_for_Ivan.tar.gz -out
files_for_Ivan.tar.gz.enc
```

Второе задание

Посчитаем хеш текстового файла:

```
md5sum pale_fire.txt
```

Теперь откроем его на редактирование и добавим пустую строку:

```
vim pale_fire.txt
md5sum pale_fire.txt
```

Как мы видим, несмотря на то что никаких полезных данных не добавилось, информация в файле поменялась. И это отразилось в хеше файла.

Третье задание

Сгенерируйте самоподписанный сертификат и настройте nginx на работу по HTTPS для тестовой веб-страницы, как было показано в материале.

Создаём самоподписанный сертификат:

```
openssl genrsa -out server.key 4096
openssl req -new -key server.key  -out server.csr
openssl x509 -req -days 365 -in server.csr -signkey server.key -out
server.crt
```

Кладём сертификат и ключ в папку nginx:

```
ls /etc/nginx/ssl
```

Раскомментируем строки listen 443, добавим пути до сертификата и ключа и впишем доменное имя, которое предварительно добавим в /etc/hosts.

Четвёртое задание

Воспользовавшись инструкцией, сгенерируйте ключевую пару «открытый — закрытый ключ», при помощи открытого ключа зашифруйте файл. Затем расшифруйте его при помощи приватного ключа. Убедитесь, что зашифрованный файл нельзя прочитать как текстовый, а расшифрованный файл совпадает с исходным.

Нам потребуется ключевая пара.
Генерируем приватный ключ:

```
openssl genpkey -algorithm RSA -out private.key -pkeyopt
rsa_keygen_bits:8192
```

Извлекаем из приватного ключа публичный ключ:

```
openssl rsa -in private.key -pubout -out public.key
```

Теперь необходимо зашифровать файл. Для этого мы используем публичный ключ, который не является секретом.
Шифруем файл:

```
openssl rsautl -encrypt -pubin -inkey публичный_ключ.key -in
файл_с_открытым_текстом.txt -out зашифрованный_файл.txt.enc
```

И расшифруем обратно с использованием уже нашего приватного ключа, который есть только у его владельца.
Расшифровываем файл:

```
openssl rsautl -decrypt -inkey приватный_ключ.key -in
зашифрованный_файл.txt.enc  -out файл_с_открытым_текстом.txt.new
```