

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра Автоматизированных систем управления

Отчет по лабораторной работе № 7

«Работа с SSH»

по курсу «ОС Linux»

Студент
Группа ПМ-18

Полухина Е.Д.

Руководитель

Кургасов В.В.

Липецк 2020 г.

СОДЕРЖАНИЕ

Цель работы	3
Задание.....	4
Ход работы	7
1. Запуск анализатора трафика tcpdump.....	7
2. Попытка установления соединения.....	8
3. Запуск анализатора трафика tcpdump с использованием 22 порта.....	9
4. Установление соединения с удаленным сервером.	10
5. Повторный запуск анализатора трафика tcpdump.	11
6. Установление зашифрованного соединения с удаленным сервером.	12
7. Вывод информации об удаленной системе.....	13
8. Передача файла по зашифрованному каналу.	14
9. Формирование зашифрованных ключей.	15
10. Передача публичного ключа.....	16
11. Подключение к удаленной системе.....	17
12. Повторная передача текстового файла на удаленный узел.....	18
13. Остановка анализатора сетевых пакетов.....	19
Контрольные вопросы.....	20
Вывод	24

Цель работы

Лабораторная работа предназначена для целей практического ознакомления с программным обеспечением удаленного доступа к распределённым системам обработки данных.

Задание

1. Создать подключение удаленного доступа к системе обработки данных, сформировать шифрованные ключи и произвести их обмен с удаленной системой, передать файл по шифрованному туннелю, воспользовавшись беспарольным доступом с аутентфикацией по публичным ключам.
2. Выполнить подключение с использованием полноэкранного консольного оконного менеджера screen.
3. Запустить терминал с командной оболочкой ОС и ввести команду tmux (терминальный мультиплексор). Комбинациями клавиш Ctrl-b с создать новое окно и запустить анализатор трафика tcpdump с фильтром пакетов получаемых и передаваемых от узла `domen.name` с TCP-портом источника и назначения 23. С помощью команды `tee`, вывести отфильтрованные IP-пакеты на терминал и сохранить данные в файл `telnet.log`, в домашнем каталоге пользователя. Для этого следует воспользоваться командой `sudo tcpdump -l -v -nn tcp and src port 23 or dst port 23 | tee telnet.log`.
4. В первом окне терминального мультиплексора попытаться установить соединение с удаленным сервером `domen.name` по протоколу TELNET. Для авторизации следует использовать логин `student`. /при возможности организовать такой доступ инженерами кафедры АСУ ЛГТУ.
5. Воспользовавшись окном сетевого монитора, анализировать прохождение сетевых пакетов между узлами назначения. Отметить пакеты инициации соединения `telnet`.
6. Подключившись к удаленной системе ввести пароль `Password` и выполнить команду `uname -a`, выведя тем самым информацию об удаленной системе. Для разрыва соединения использовать команду `logout`.
7. В окне сетевого монитора отметить пакеты инициирующие разрыв сессии `telnet`. Прервать фильтрацию пакетов сетевым анализатором

- tcpdump, воспользовавшись комбинацией Ctrl-c. В файле telnet.log выделить записи установления и разрыва соединения с сервером telnet.
8. Снова запустить анализатор сетевого трафика с фильтром пакетов получаемых и передаваемых узлу domen.name с TCP-портом источника и назначения 22. С помощью команды tee, вывести отфильтрованные IP-пакеты на терминал и сохранить данные в файл ssh.log, в домашнем каталоге пользователя. Для этого следует воспользоваться командой `sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee ssh.log`.
 9. Переключившись на первое окно терминального мультиплексора, с помощью команды `ssh -l student domen.name` попытаться установить зашифрованное соединение с удаленным сервером domen.name. Проследить передачу и прием пакетов между узлами в окне сетевого анализатора. Отметить взаимодействующие TCP-порты.
 10. Подключившись к удаленной системе ввести пароль Password и выполнить команду `uname -a`, выведя информацию об удаленной системе.
 11. Создать текстовый файл с содержанием ФИО и номера лабораторной работы на локальном узле и с помощью команды `scp -v -o User=student/home/student/имя_файла domen.name:/home/student/` передать его по зашифрованному каналу на удаленную систему. Проверить наличие копии переданного файла на удаленном узле, воспользовавшись файловым менеджером «Midnight Commander» (команда `mc` на удаленной системе).
 12. Отключившись от удаленного узла (команда `exit`), на локальном хосте, сформировать зашифрованные ключи, воспользовавшись командой `ssh-keygen`.
 13. Используя команду `scp` с указанием места расположения файла (публичного ключа) на локальной системе (`/home/student/.ssh/key.pub`), произвести его передачу по зашифрованному туннелю на удаленный узел

в заданный каталог `/home/student/.ssh/` под именем `authorized_keys`. Проследить процесс пересылки пакетов между удаленными узлами в окне анализатора пакетов.

14. Воспользовавшись командой `ssh -l student domen.name`, снова сделать попытку подключения к удаленной системе. Отметить отличия в процедурах подключения и регистрации пользователя на удаленной системе.
15. Аналогично, с помощью команды `scp`, произвести повторную передачу текстового файла на удаленный узел. Убедиться в наличии переданной копии файла на удаленном хосте. Отметить отличия в процедуре передачи файла.
16. Остановить анализатор сетевых пакетов, воспользовавшись комбинацией `Ctrl-c`. Просмотреть содержимое файла `ssh.log`, отметить пакеты инициации сетевого взаимодействия и разрыва соединений TCP.

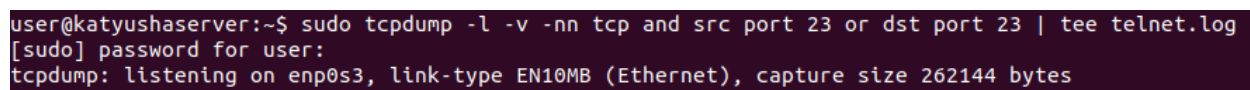
Ход работы

1. Запуск анализатора трафика tcpdump.

Запускаем терминал командной оболочки и вводим команду `tmux` для открытия терминального мультиплексора.

С помощью комбинации клавиш «Ctrl-b c» создаем новое окно и запускаем анализатор трафика `tcpdump`.

После этого вводим команду «`sudo tcpdump -l -v -nn tcp and src port 23 or dst port 23 | tee telnet.log`» для вывода отфильтрованных IP-пакетов на терминал и сохранения данных в `telnet.log`.



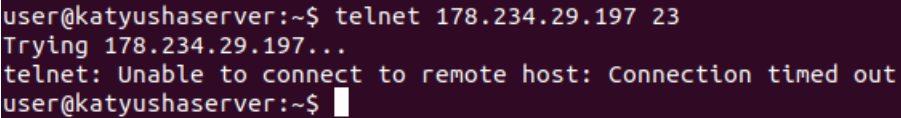
```
user@katyushaserver:~$ sudo tcpdump -l -v -nn tcp and src port 23 or dst port 23 | tee telnet.log
[sudo] password for user:
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Рисунок 1 – Запуск анализатора трафика

2. Попытка установления соединения.

В первом окне терминального мультиплексора пытаемся установить соединение с удаленным сервером 178.234.29.197.

Для этого переходим в первое окно с помощью комбинации клавиш «Ctrl-b 0» и вводим команду «telnet 178.234.29.197 23».

A terminal window with a dark background and light-colored text. The text shows a user at a prompt trying to connect to a remote host via telnet, but the connection times out.

```
user@katyushaserver:~$ telnet 178.234.29.197 23
Trying 178.234.29.197...
telnet: Unable to connect to remote host: Connection timed out
user@katyushaserver:~$
```

Рисунок 2 – Установление соединения с удаленным сервером

Как мы видим, 23 порт недоступен, нет возможности подключиться к серверу удалённо.

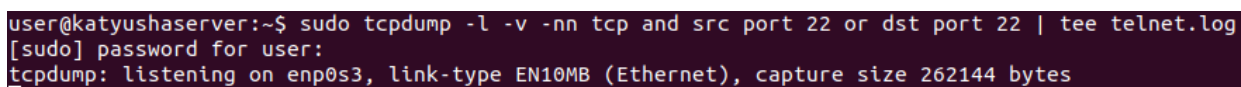
3. Запуск анализатора трафика tcpdump с использованием 22 порта.

Попробуем установить соединение с удаленным сервером через 22 порт.

Запускаем терминал командной оболочки и вводим команду `tmux` для открытия терминального мультиплексора.

С помощью комбинации клавиш «Ctrl-b c» создаем новое окно и запускаем анализатор трафика tcpdump.

После этого вводим команду «`sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee telnet.log`» для вывода отфильтрованных IP-пакетов на терминал и сохранения данных в `telnet.log`.



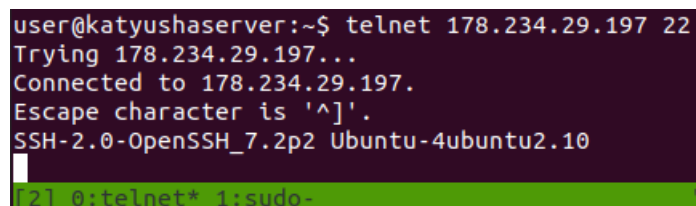
```
user@katyushaserver:~$ sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee telnet.log
[sudo] password for user:
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Рисунок 3 – Запуск анализатора трафика

4. Установление соединения с удаленным сервером.

В первом окне терминального мультиплексора пытаемся установить соединение с удаленным сервером 178.234.29.197.

Для этого переходим в первое окно с помощью комбинации клавиш «Ctrl-b 0» и вводим команду «telnet 178.234.29.197 22».



```
user@katyushaserver:~$ telnet 178.234.29.197 22
Trying 178.234.29.197...
Connected to 178.234.29.197.
Escape character is '^]'.
SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.10
[2] 0:telnet* 1:sudo-
```

Рисунок 4 – Установление соединения

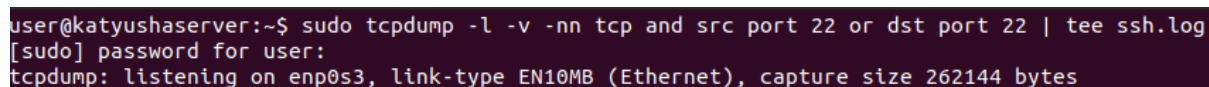
Как видим, связь есть, но нет возможности войти с использованием логина и пароля. Также при вводе любой команды появляется ошибка о несоответствии протокола.

5. Повторный запуск анализатора трафика tcpdump.

Запускаем терминал командной оболочки и вводим команду `tmux` для открытия терминального мультиплексора.

С помощью комбинации клавиш «Ctrl-b c» создаем новое окно и запускаем анализатор трафика `tcpdump`.

После этого вводим команду «`sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee ssh.log`» для вывода отфильтрованных IP-пакетов на терминал и сохранения данных в `ssh.log`.

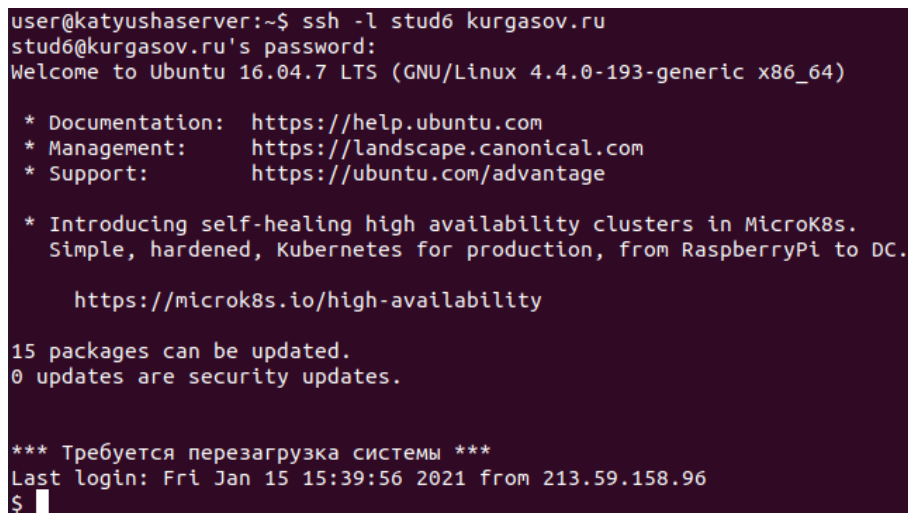


```
user@katyushaserver:~$ sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee ssh.log
[sudo] password for user:
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
```

Рисунок 5 – Запуск анализатора трафика

6. Установление шифрованного соединения с удаленным сервером.

Переключаемся на первое окно терминального мультиплексора. Затем с помощью команды «ssh -l stud6 kurgasov.ru» устанавливаем соединение с удаленным сервером, вводим пароль.



```
user@katyushaserver:~$ ssh -l stud6 kurgasov.ru
stud6@kurgasov.ru's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-193-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

   https://microk8s.io/high-availability

15 packages can be updated.
0 updates are security updates.

*** Требуется перезагрузка системы ***
Last login: Fri Jan 15 15:39:56 2021 from 213.59.158.96
$ █
```

Рисунок 6 – Установление шифрованного соединения с удаленным сервером

7. Вывод информации об удаленной системе.

С помощью команды «uname -a» выводим информацию об удаленной системе.

```
$ uname -a  
Linux kurgasov.ru 4.4.0-193-generic #224-Ubuntu SMP Tue Oct 6 17:15:28 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux  
$
```

Рисунок 7 – Информация об удаленной системе

8. Передача файла по зашифрованному каналу.

С сочетания клавиш «Ctrl-b c» создаем новое окно. Затем создаем файл, в который записываем ФИО и номера лабораторной работы.

Затем с помощью команды «scp -v -o ~/lr7 stud6@kurgasov.ru:/home/stud6» передаем файл по зашифрованному каналу на удаленную систему.

```
GNU nano 4.8
Full name: Polukhina Ekaterina Dmitrievna
Lab: 7
```

Рисунок 8 – Содержание файла lr7

```
user@katyushaserver:~$ scp ~/lr7 stud6@kurgasov.ru:/home/stud6
stud6@kurgasov.ru's password:
lr7
user@katyushaserver:~$
```

Рисунок 9 – Передача файла по зашифрованному каналу

Проверяем наличие копии переданного файла на удаленном узле, воспользовавшись файловым менеджером «Midnight Commander» с помощью команды «mc».

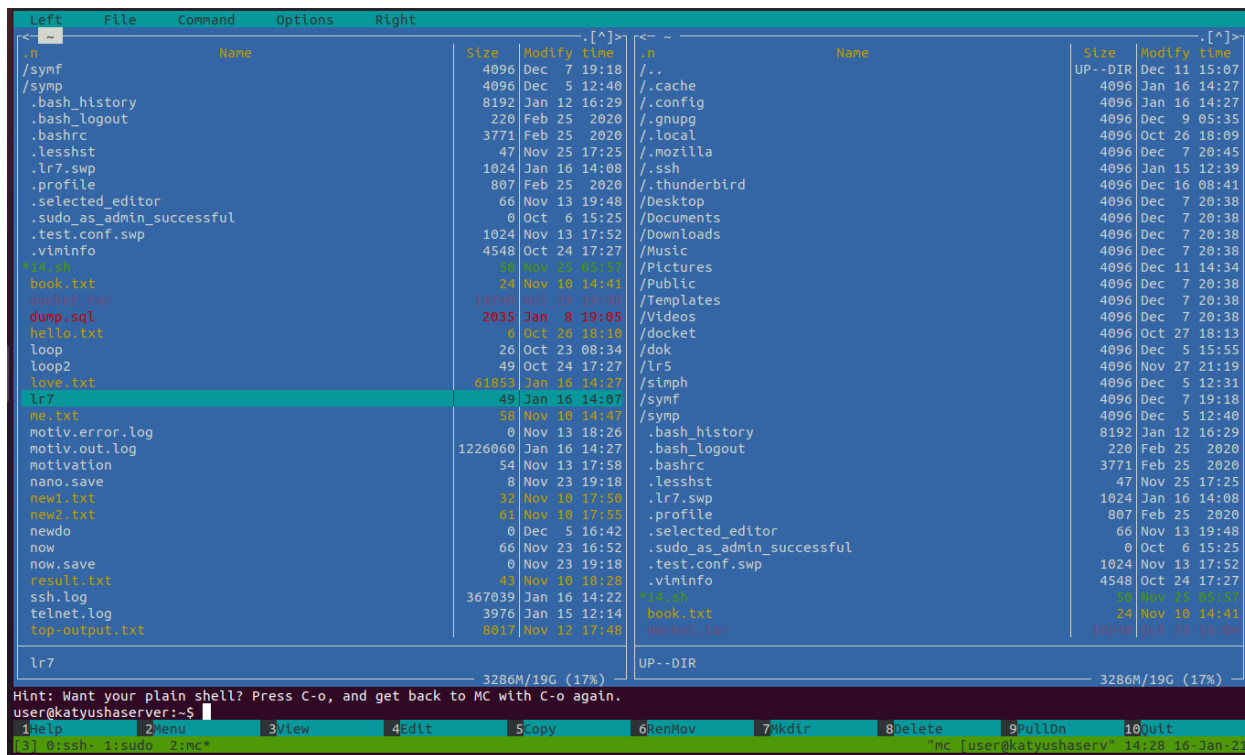


Рисунок 10 – Просмотр файлов на удаленном узле

9. Формирование зашифрованных ключей.

Отключаемся от удаленного узла с помощью команды `exit`.

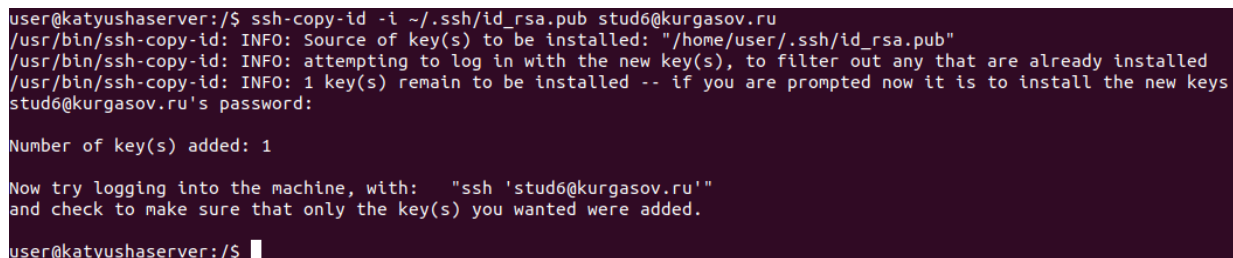
Используя команду «`ssh-keygen`» формируем зашифрованные ключи.

```
user@katyushaserver:/$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
/home/user/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa
Your public key has been saved in /home/user/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1oS+jEm8WMjtV1LIuXrV1DVzT7m+/S4vZDIp+z4LHkY user@katyushaserver
The key's randomart image is:
+---[RSA 3072]-----+
|           o=|
|      . +   . +=|
|     = o . . o|
|  . + . = o   .|
|   o = S E ...|
|   = B *. + o.|
|  . * = +o = o|
|    o o.o. o..|
|     .o+o ==|
+----[SHA256]-----+
user@katyushaserver:/$
```

Рисунок 11 – Формирование зашифрованных ключей

10. Передача публичного ключа.

Производим передачу публичного ключа по зашифрованному туннелю на удаленный узел с помощью команды «ssh-copy-id -i ~/.ssh/id_rsa.pub stud6@kurgasov.ru».



```
user@katyushaserver:/$ ssh-copy-id -i ~/.ssh/id_rsa.pub stud6@kurgasov.ru
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/user/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
stud6@kurgasov.ru's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'stud6@kurgasov.ru'"
and check to make sure that only the key(s) you wanted were added.
user@katyushaserver:/$
```

Рисунок 12 – Передача публичного ключа

11. Подключение к удаленной системе.

Используя команду «ssh -l stud6 kurgasov.ru» подключаемся к удаленной системе. Как мы видим, благодаря ssh пароль при входе не потребовался.

```
user@katyushaserver:/$ ssh -l stud6 kurgasov.ru
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-193-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

   https://microk8s.io/high-availability

15 packages can be updated.
0 updates are security updates.

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** Требуется перезагрузка системы ***
Last login: Sat Jan 16 16:53:55 2021 from 213.59.158.96
$
```

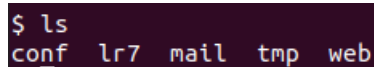
Рисунок 13 – Подключение к удаленной системе

12. Повторная передача текстового файла на удаленный узел.

С помощью команды «`scp ~/lr7 stud6@kurgasov.ru:/home/stud6`» передаем файл по зашифрованному каналу на удаленную систему.

При передаче файла не потребовался ввод пароля.

Видим, что переданная копия файла действительно существует на удаленном хосте.



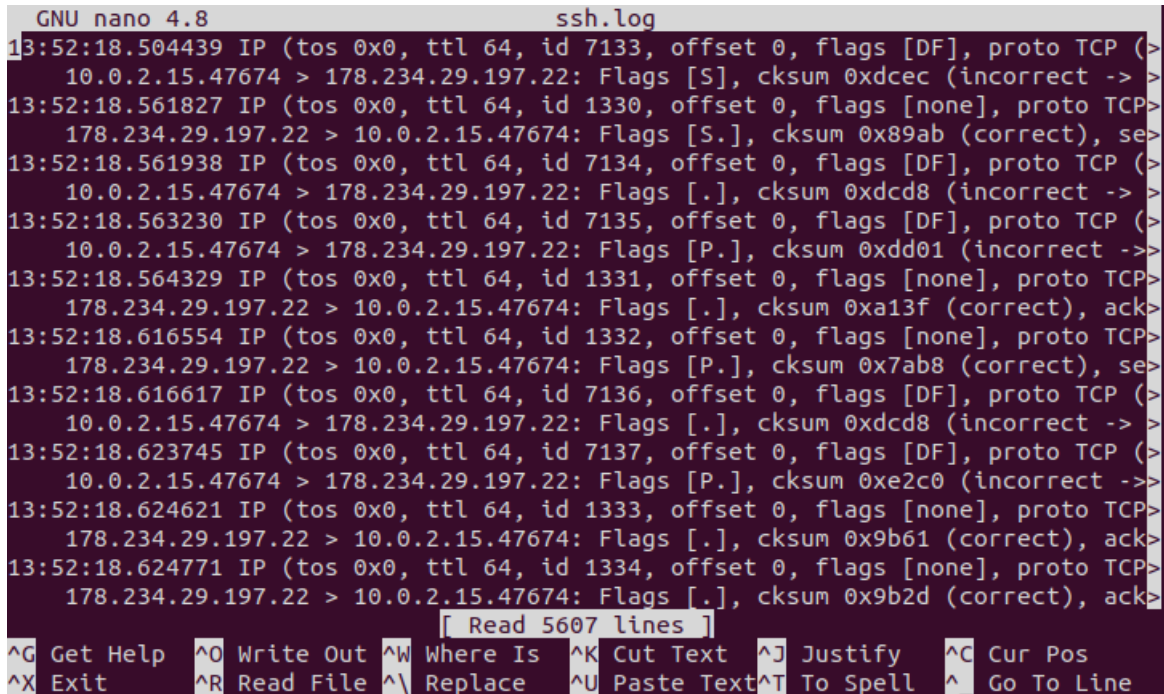
```
$ ls
conf lr7 mail tmp web
```

Рисунок 14 – Повторная передача файла

13. Остановка анализатора сетевых пакетов.

Останавливаем анализатор сетевых пакетов, используя комбинацию клавиш «Ctrl-c».

Затем смотрим содержимое файла ssh.log.



```
GNU nano 4.8 ssh.log
13:52:18.504439 IP (tos 0x0, ttl 64, id 7133, offset 0, flags [DF], proto TCP (>
  10.0.2.15.47674 > 178.234.29.197.22: Flags [S], cksum 0xdcec (incorrect -> >
13:52:18.561827 IP (tos 0x0, ttl 64, id 1330, offset 0, flags [none], proto TCP>
  178.234.29.197.22 > 10.0.2.15.47674: Flags [S.], cksum 0x89ab (correct), se>
13:52:18.561938 IP (tos 0x0, ttl 64, id 7134, offset 0, flags [DF], proto TCP (>
  10.0.2.15.47674 > 178.234.29.197.22: Flags [.], cksum 0xdcd8 (incorrect -> >
13:52:18.563230 IP (tos 0x0, ttl 64, id 7135, offset 0, flags [DF], proto TCP (>
  10.0.2.15.47674 > 178.234.29.197.22: Flags [P.], cksum 0xdd01 (incorrect -> >
13:52:18.564329 IP (tos 0x0, ttl 64, id 1331, offset 0, flags [none], proto TCP>
  178.234.29.197.22 > 10.0.2.15.47674: Flags [.], cksum 0xa13f (correct), ack>
13:52:18.616554 IP (tos 0x0, ttl 64, id 1332, offset 0, flags [none], proto TCP>
  178.234.29.197.22 > 10.0.2.15.47674: Flags [P.], cksum 0x7ab8 (correct), se>
13:52:18.616617 IP (tos 0x0, ttl 64, id 7136, offset 0, flags [DF], proto TCP (>
  10.0.2.15.47674 > 178.234.29.197.22: Flags [.], cksum 0xdcd8 (incorrect -> >
13:52:18.623745 IP (tos 0x0, ttl 64, id 7137, offset 0, flags [DF], proto TCP (>
  10.0.2.15.47674 > 178.234.29.197.22: Flags [P.], cksum 0xe2c0 (incorrect -> >
13:52:18.624621 IP (tos 0x0, ttl 64, id 1333, offset 0, flags [none], proto TCP>
  178.234.29.197.22 > 10.0.2.15.47674: Flags [.], cksum 0x9b61 (correct), ack>
13:52:18.624771 IP (tos 0x0, ttl 64, id 1334, offset 0, flags [none], proto TCP>
  178.234.29.197.22 > 10.0.2.15.47674: Flags [.], cksum 0x9b2d (correct), ack>
[ Read 5607 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace  ^U Paste Text ^T To Spell  ^_ Go To Line
```

Рисунок 15 – Содержимое файла ssh.log

Контрольные вопросы

- 1) Определите основные цели и задачи решаемые с помощью ПО удаленного доступа?

ПО удаленного доступа дает пользователю возможность подключаться к компьютеру с помощью другого устройства через интернет.

Для создания удаленного подключения используют специальные программы. Обязательное условие — наличие постоянного доступа в интернет, компьютеров, обладающих определенными характеристиками и сервера. Удаленное подключение связывает две рабочие станции через интернет. В стандартном приложении Windows соединение происходит между двумя IP-адресами, но если компьютер находится в локальной сети, то подключиться к нему извне можно только с помощью специальных программ удаленного доступа. Такое ПО делает возможным подключение к другому компьютеру из любой точки мира.

Программы позволяют видеть рабочий стол и выполнять все действия на удаленном устройстве, изменять настройки ПО, обмениваться файлами, шифровать передаваемые данные, проводить конференции, подключать веб-камеры, удаленные проекторы и прочие сетевые устройства.

- 2) Выделите отличительные особенности между режимами работы удаленного доступа по протоколам TELNET и SSH?

- SSH и Telnet - это сетевые протоколы, которые позволяют пользователям входить в удаленные системы и выполнять на них команды.
- Доступ к командной строке удаленного хоста одинаков для обоих протоколов, но основное различие этих протоколов зависит от меры безопасности каждого из них. SSH более защищен, чем Telnet.
- По умолчанию SSH использует порт 22, а Telnet использует порт 23 для связи, и оба используют стандарт TCP.

- SSH отправляет все данные в зашифрованном формате, а Telnet отправляет данные в виде обычного текста. Поэтому SSH использует безопасный канал для передачи данных по сети, а Telnet использует обычный способ подключения к сети и связи.
- SSH использует шифрование с открытым ключом для аутентификации удаленных пользователей, а Telnet не использует механизмов аутентификации.
- Учитывая безопасность, доступную в каждом протоколе, SSH подходит для использования в общедоступных сетях, а Telnet больше подходит для частных сетей.

3) Опишите способы установления соединения при использовании протокола SSH? Охарактеризуйте положительные и отрицательные аспекты приведенных методов.

Конфигурация	Вероятность взлома	Потери от флуда**
22 порт, авторизация по паролю, без защиты	Высокая	Высокие
22 порт, авторизация по ключам, без защиты	Средняя***	Высокие
22 порт, авторизация по ключам, защита на основе ограничения неудачных попыток авторизации	Низкая	Средние****
Нестандартный порт, авторизация по паролю, без защиты	Высокая	Низкие

Нестандартный порт, авторизация по ключам, без защиты	Средняя***	Низкие
Нестандартный порт, авторизация по ключам, защита на основе ограничения неудачных попыток авторизации	Низкая	Низкие

* – значения параметров (высокий, средний, низкий) носят относительный характер и служат только для сравнения показателей.

** – расход ресурсов сервера (процессор, диск, сетевой канал) на обработку запросов, обычно идущих на 22-й порт.

*** – произвести взлом, если для авторизации используются RSA-ключи, сложно, однако неограниченное количество попыток авторизации делает это возможным.

**** – количество попыток авторизации ограничено, но серверу приходится обрабатывать их от большого количества злоумышленников.

4) Основываясь на заданиях лабораторной работы, приведите практический пример использования систем удаленного доступа?

Системы удаленного доступа нужны тем компаниям, где большинство сотрудников находится за пределами офиса, на частичном фрилансе, аутсорсинге или в командировках, но при этом они нуждаются в обновлении рабочей информации, просмотре корпоративной почты и др. Им не нужно будет скачивать все необходимые для работы данные на внешний носитель или отправлять их по почте – достаточно связаться с офисным компьютером.

Удаленный доступ используют системные администраторы для управления системой и устранения сбоев в ее работе, и руководители, желающие проконтролировать процесс выполнения задачи своими подчиненными.

- 5) Перечислите распространенные сетевые службы, основанные на использовании шифрованного соединения по протоколу SSH? Приведите пример использования службы передачи файлов по безопасному туннелю? Распространенные сетевые службы, основанные на использовании шифрованного соединения по протоколу SSH: OpenSSH, PuTTY/KiTTY, SecureCRT, Xshell. Службы передачи файлов по безопасному туннелю можно использовать для передачи паролей.

Вывод

В результате выполнения лабораторной работы я получила знания по программному обеспечению удаленного доступа к распределённым системам обработки данных. Научилась устанавливать шифрованное соединение с удаленным сервером, передавать файлы по шифрованному каналу на удаленную систему. Также поняла, как передавать публичный ключ по шифрованному туннелю на удаленный узел и подключаться к удаленной системе без использования пароля.