

Липецкий государственный технический университет

Кафедра прикладной математики

Отчет по лабораторной работе № 7
«Работа с SSH»
по курсу «Операционная система Linux»

Студент

подпись, дата

Пустовалова И.П.

фамилия, инициалы

Группа

Руководитель

Доцент, к. пед. наук

ученая степень, ученое звание

подпись, дата

Кургасов В.В.

фамилия, инициалы

Липецк 2021 г.

Содержание

Цель работы	3
Задание кафедры	3
Ход работы	6
Выводы	13
Контрольные вопросы	14

Цель работы

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

Задание кафедры

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.

1. 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 ТСП-портом источника и назначения 22. 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 22 or dst port 22 | tee telnet.log;
```

2. 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/
3. Воспользовавшись окном сетевого монитора, анализировать прохождение сетевых пакетов между узлами назначения. Отметить пакеты инициации соединения `telnet`;

4. 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`;
5. В окне сетевого монитора отметить пакеты иницирующие 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;
6. 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`;
7. Переключившись на первое окно терминального мультиплексора, с помощью команды `ssh -l student domen.name` попытаться установить шифрованное соединение с удаленным сервером `domen.name`. Проследить передачу и прием пакетов между узлами в окне сетевого анализатора. Отметить взаимодействующие TCP-порты;
8. Подключившись к удаленной системе ввести пароль Password и выполнить команду `uname -a`, выведя информацию об удаленной системе;
9. Создать текстовый файл с содержанием ФИО и номера лабораторной работы на локальном узле и с помощью команды `scp -v -o User=student/home/st domen.name:/home/student/` передать его по шифрованному каналу на удаленную систему. Проверить наличие копии переданного файла на удаленном узле, воспользовавшись файловым менеджером «Midnight Commander» (команда `mc` на удаленной системе);
10. Отключившись от удаленного узла (команда `exit`), на локальном хосте, сформировать зашифрованные ключи, воспользовавшись командой `ssh-keygen`;

11. 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;
12. 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;
13. 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;
14. Остановить анализатор сетевых пакетов, воспользовавшись комбинацией `Ctrl-c`. Просмотреть содержимое файла `ssh.log`, отметить пакеты инициации сетевого взаимодействия и разрыва соединений TCP.

Ход работы

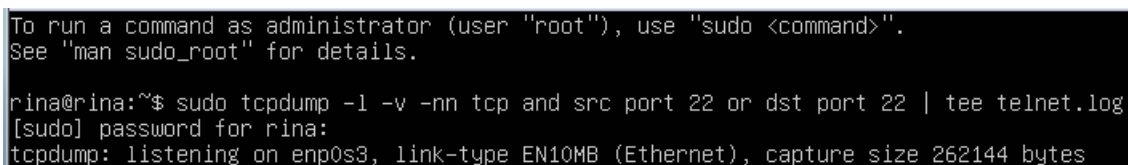
Начальные данные:

Логин – stud8

Пароль – zTuNMW8V7s

Запуск анализатора трафика tcpdump (порт 22)

- tmux (терминальный мультиплексор)
- Ctrl-b с (создание нового окна)
- `sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee telnet.log`



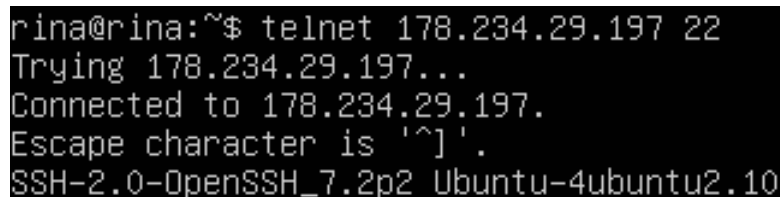
```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

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

Рисунок 1 – Запуск анализатора трафика

Установление соединения

- Ctrl-b 0
- `telnet 178.234.29.197 22`



```
rina@rina:~$ 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
```

Рисунок 2 – Установление соединения

Повторный запуск анализатора трафика tcpdump

- Ctrl-b c
- `sudo tcpdump -l -v -nn tcp and src port 22 or dst port 22 | tee ssh.log`

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

Рисунок 3 – Запуск анализатора трафика

Установка шифрованного соединения с удаленным сервером

- Ctrl-b 0
- `ssh -l stud8 kurgasov.ru`

```
rina@rina:~$ ssh -l stud8 kurgasov.ru
The authenticity of host 'kurgasov.ru (178.234.29.197)' can't be established.
ECDSA key fingerprint is SHA256:c7y8uU2/zFt5w6UuLfUeRk/0hPMih9uki+EY2Vo1qik.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'kurgasov.ru,178.234.29.197' (ECDSA) to the list of known hosts.
stud8@kurgasov.ru's password:
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-210-generic x86_64)

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

22 packages can be updated.
5 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Feb  2 17:40:35 2021 from 91.246.126.37
stud8@kurgasov:~$
```

Рисунок 4 – Установка шифрованного соединения с удаленным сервером

Вывод информации об удаленной системе.

- `uname -a`

```
stud8@kurgasov:~$ uname -a
Linux kurgasov.ru 4.4.0-210-generic #242-Ubuntu SMP Fri Apr 16 09:57:56 UTC 2021 x86_64 x86_64 x86_64
4 GNU/Linux
stud8@kurgasov:~$
```

Рисунок 5 – Информация об удаленной системе

Передача файла по зашифрованному каналу

- Ctrl-b c
- Создаем файл lr7.txt в который записываем ФИО и номера лабораторной работы
- `scp -v -o /lr7 stud8@kurgasov.ru:/home/stud8`

```
rina@rina:~$ scp lr7.txt stud8@kurgasov.ru:/home/stud8
stud8@kurgasov.ru's password:
Permission denied, please try again.
stud8@kurgasov.ru's password:
lr7.txt
lr7.txt
100% 95 1.8KB/s 00:00
rina@rina:~$
```

Рисунок 6 – Передача файла по зашифрованному каналу

- mc (Проверяем наличие копии переданного файла на удаленном узле)

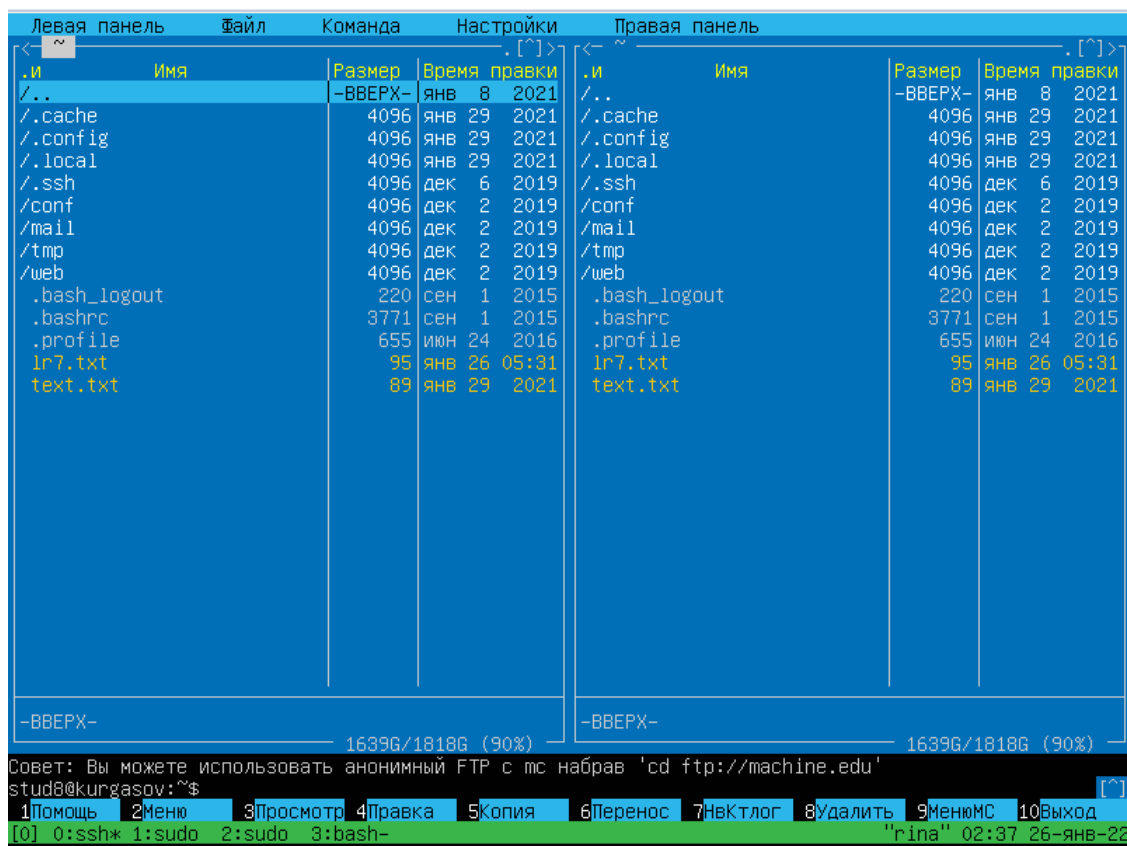


Рисунок 7 – Просмотр файлов на удаленном узле

Формирование зашифрованных ключей.

- exit
- ssh-keygen

```
stud8@kurgasov:~$ exit
ВЫХОД
Connection to edu.kurgasov.ru closed.
rina@rina:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/rina/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/rina/.ssh/id_rsa
Your public key has been saved in /home/rina/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:aGw71e0c5/IYLls60uECKmFXjUMdcNcT9XI42yEbuTc rina@rina
The key's randomart image is:
+----[RSA 3072]-----+
|      .00....0+..    |
|      .... 0 .* .    |
|      . 0      .=. =  |
|      = 0 .    .*E    |
|      . * S 0      . . |
|.....0.0 0 0        |
|.0.. .0. * .        |
|0   . 0.0+. =       |
|.   ..00+000        |
+-----[SHA256]-----+
rina@rina:~$
```

Рисунок 8 – Формирование зашифрованных ключей

Передача публичного ключа

- ssh-copy-id -i ~/.ssh/id_rsa.pub stud8@kurgasov.ru

```
rina@rina:~$ ssh-copy-id -i ~/.ssh/id_rsa.pub stud8@kurgasov.ru
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/rina/.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 alr
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
all the new keys
stud8@kurgasov.ru's password:

Number of key(s) added: 1

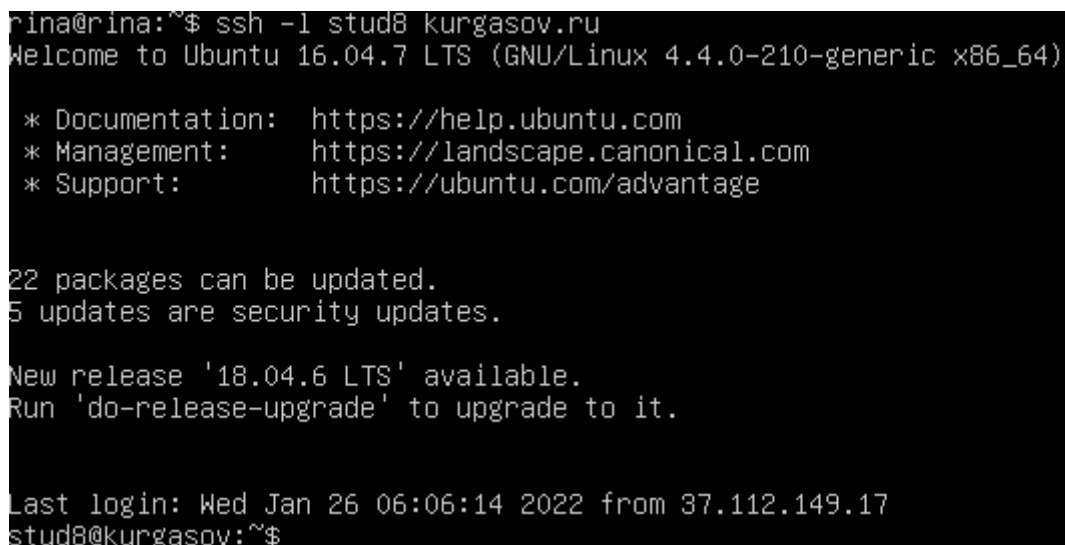
Now try logging into the machine, with:  "ssh 'stud8@kurgasov.ru'"
and check to make sure that only the key(s) you wanted were added.
rina@rina:~$
```

Рисунок 9 – Передача публичного ключа

Подключение к удаленной системе

- `ssh -l stud8 kurgasov.ru`

Как мы видим, благодаря ssh пароль при входе не потребовался



```
rina@rina:~$ ssh -l stud8 kurgasov.ru
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-210-generic x86_64)

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

22 packages can be updated.
5 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Jan 26 06:06:14 2022 from 37.112.149.17
stud8@kurgasov:~$
```

Рисунок 10 – Подключение к удаленной системе

Повторная передача текстового файла на удаленный узел

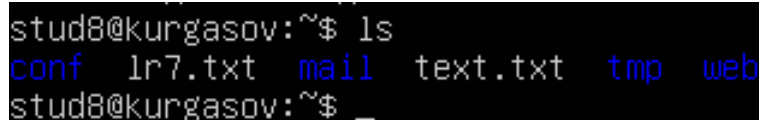
- `scp ~/lr7.txt stud8@kurgasov.ru:/home/stud8`

Как мы видим, благодаря ssh при передаче файла не потребовался ввод пароля



```
rina@rina:~$ scp ~/lr7.txt stud8@kurgasov.ru:/home/stud8
lr7.txt                                100% 95    1.3KB/s   00:00
rina@rina:~$
```

Рисунок 11 – Повторная передача файла



```
stud8@kurgasov:~$ ls
conf  lr7.txt  mail  text.txt  tmp  web
stud8@kurgasov:~$
```

Рисунок 12 – Проверка наличия файла на удаленном хосте

Остановка анализатора сетевых пакетов. Содержимое файлов telnet.log и ssh.log

- Ctrl-c
- nano telnet.log
- nano ssh.log

```
GNU nano 4.8 telnet.log
03:00:38.945214 IP (tos 0x10, ttl 64, id 41458, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0xa0eb), seq 39889492
03:00:39.951767 IP (tos 0x10, ttl 64, id 41459, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0x9cfd), seq 39889492
03:00:41.967982 IP (tos 0x10, ttl 64, id 41460, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0x951c), seq 39889492
03:00:46.192069 IP (tos 0x10, ttl 64, id 41461, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0x849c), seq 39889492
03:00:54.383981 IP (tos 0x10, ttl 64, id 41462, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0x649c), seq 39889492
03:01:10.512261 IP (tos 0x10, ttl 64, id 41463, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0x259c), seq 39889492
03:01:42.768168 IP (tos 0x10, ttl 64, id 41464, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.47454 > 178.23.29.197.22: Flags [S], cksum 0xdc19 (incorrect -> 0xa79b), seq 39889492
03:03:06.066250 IP (tos 0x10, ttl 64, id 38463, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.34280 > 178.234.29.197.22: Flags [S], cksum 0xdcec (incorrect -> 0xe3e4), seq 1044747
03:03:06.120153 IP (tos 0x0, ttl 64, id 79, offset 0, flags [none], proto TCP (6), length 44)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [S.], cksum 0xe7c7 (correct), seq 20544001, ack 1044
03:03:06.120267 IP (tos 0x10, ttl 64, id 38464, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34280 > 178.234.29.197.22: Flags [.], cksum 0xdcd8 (incorrect -> 0x0494), ack 1, win
03:03:06.174968 IP (tos 0x0, ttl 64, id 80, offset 0, flags [none], proto TCP (6), length 82)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [P.], cksum 0xd8fd (correct), seq 1:43, ack 1, win 6
03:03:06.175030 IP (tos 0x10, ttl 64, id 38465, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34280 > 178.234.29.197.22: Flags [.], cksum 0xdcd8 (incorrect -> 0x0494), ack 43, win
03:05:06.181640 IP (tos 0x0, ttl 64, id 81, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [F.], cksum 0xff59 (correct), seq 43, ack 1, win 655
03:05:06.181947 IP (tos 0x10, ttl 64, id 38466, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34280 > 178.234.29.197.22: Flags [F.], cksum 0xdcd8 (incorrect -> 0x0493), seq 1, ack
03:05:06.182578 IP (tos 0x0, ttl 64, id 82, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [.], cksum 0xff58 (correct), ack 2, win 65535, lengt
03:05:42.151402 IP (tos 0x0, ttl 64, id 15870, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [S], cksum 0xdcec (incorrect -> 0x0a80), seq 3577312
[ Read 2138 lines ]
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line    M-E Redo
```

Рисунок 13 – Содержимое файла telnet.log

```
GNU nano 4.8 ssh.log
03:05:06.181640 IP (tos 0x0, ttl 64, id 81, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [F.], cksum 0xff59 (correct), seq 20544044, ack 1044
03:05:06.181947 IP (tos 0x10, ttl 64, id 38466, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34280 > 178.234.29.197.22: Flags [F.], cksum 0xcdcd8 (incorrect -> 0x0493), seq 1, ack
03:05:06.182578 IP (tos 0x0, ttl 64, id 82, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34280: Flags [L], cksum 0xff58 (correct), ack 2, win 65535, lengt
03:05:42.151402 IP (tos 0x0, ttl 64, id 15870, offset 0, flags [DF], proto TCP (6), length 60)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [S], cksum 0xdcec (incorrect -> 0x0a80), seq 3577312
03:05:42.212612 IP (tos 0x0, ttl 64, id 85, offset 0, flags [none], proto TCP (6), length 44)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [S.], cksum 0x8d1b (correct), seq 37248001, ack 3577
03:05:42.212739 IP (tos 0x0, ttl 64, id 15871, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [L], cksum 0xcdcd8 (incorrect -> 0xa9e7), ack 1, win
03:05:42.214882 IP (tos 0x0, ttl 64, id 15872, offset 0, flags [DF], proto TCP (6), length 81)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [P.], cksum 0xdd01 (incorrect -> 0x858e), seq 1:42,
03:05:42.215615 IP (tos 0x0, ttl 64, id 86, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [L], cksum 0xa4af (correct), ack 42, win 65535, leng
03:05:42.279358 IP (tos 0x0, ttl 64, id 87, offset 0, flags [none], proto TCP (6), length 82)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [P.], cksum 0x7e28 (correct), seq 1:43, ack 42, win
03:05:42.279417 IP (tos 0x0, ttl 64, id 15873, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [L], cksum 0xcdcd8 (incorrect -> 0xa9be), ack 43, win
03:05:42.281331 IP (tos 0x0, ttl 64, id 15874, offset 0, flags [DF], proto TCP (6), length 1552)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [P.], cksum 0xe2c0 (incorrect -> 0x963e), seq 42:155
03:05:42.282186 IP (tos 0x0, ttl 64, id 88, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [L], cksum 0x9ed1 (correct), ack 1502, win 65535, le
03:05:42.282453 IP (tos 0x0, ttl 64, id 89, offset 0, flags [none], proto TCP (6), length 40)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [L], cksum 0x9e9d (correct), ack 1554, win 65535, le
03:05:42.341719 IP (tos 0x0, ttl 64, id 90, offset 0, flags [none], proto TCP (6), length 1016)
  178.234.29.197.22 > 10.0.2.15.34282: Flags [P.], cksum 0x3696 (correct), seq 43:1019, ack 1554,
03:05:42.341788 IP (tos 0x0, ttl 64, id 15876, offset 0, flags [DF], proto TCP (6), length 40)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [L], cksum 0xcdcd8 (incorrect -> 0xa2fc), ack 1019, w
03:05:42.352268 IP (tos 0x0, ttl 64, id 15877, offset 0, flags [DF], proto TCP (6), length 88)
  10.0.2.15.34282 > 178.234.29.197.22: Flags [P.], cksum 0xdd08 (incorrect -> 0x3095), seq 1554:1
[ Read 2114 lines ]
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo
^X Exit ^R Read File ^_ Replace ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```

Рисунок 14 – Содержимое файла ssh.log.

Вывод

В ходе выполнения данной лабораторной работы мной были получены знания о создании БД, удалении БД и восстановлении БД из дампа MySQL.

Контрольные вопросы

- Что такое ключ ssh? В чем преимущество их использования?

SSH-ключи используются для идентификации клиента при подключении к удалённому серверу. SSH-ключи представляют собой пару ключей – приватный и публичный. Приватный ключ хранится в закрытом доступе у клиента, публичный отправляется на сервер. Преимущество использования ключей в удобстве (не нужно запоминать пароли) и безопасности (взломать приватный ssh-ключ достаточно сложно).

- Как сгенерировать ключи ssh в разных ОС?

Генерация ssh-ключа в ОС Linux возможна с помощью команды `sshkeygen`. В ОС Windows можно использовать программу PuTTY для генерации ssh-ключей и подключения по ssh-протоколу.

- Возможно ли из «секретного» ключа сгенерировать «публичный» и/или наоборот?

Нет, невозможно.

- Будут ли отличаться пары ключей, сгенерированные на одном ПК несколько раз с исходными условиями (наличие/отсутствие пароля на «секретный» ключ и т.п.)

Да, будут. Утилита `ssh-keygen` каждый раз случайно генерирует пару ключей.

- Перечислите доступные ключи для `ssh-keygen.exe`

1. DSA;
2. RSA;
3. ECDASA;
4. Ed25519;

- Можно ли использовать один «секретный» ключ доступа с разных ОС, установленных на одном ПК/на разных ПК?

Можно, но безопасность такого ключа уже не гарантирована.

- Возможно ли организовать подключение «по ключу» ssh к системе с ОС Windows, в которой запущен OpenSSH сервер?

Да, возможно, с использованием программы PuTTY.

- Какие известные Вам сервисы сети Интернет позволяют организовать доступ к ресурсам посредством SSH ключей?

Один из самых известных – GitHub.