

**Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Российский государственный университет нефти и газа
(национальный исследовательский университет)
имени И. М. Губкина»**

Кафедра Автоматизированных систем управления

Отчет по лабораторной работе № 4
дисциплины *Основы организации операционных систем*

Настройка SSH-протокола

Группа: АС-23-04

Студент: Ханеский Ярослав
Александрович

К.т.н., доцент Фридлянд
Александр Михайлович

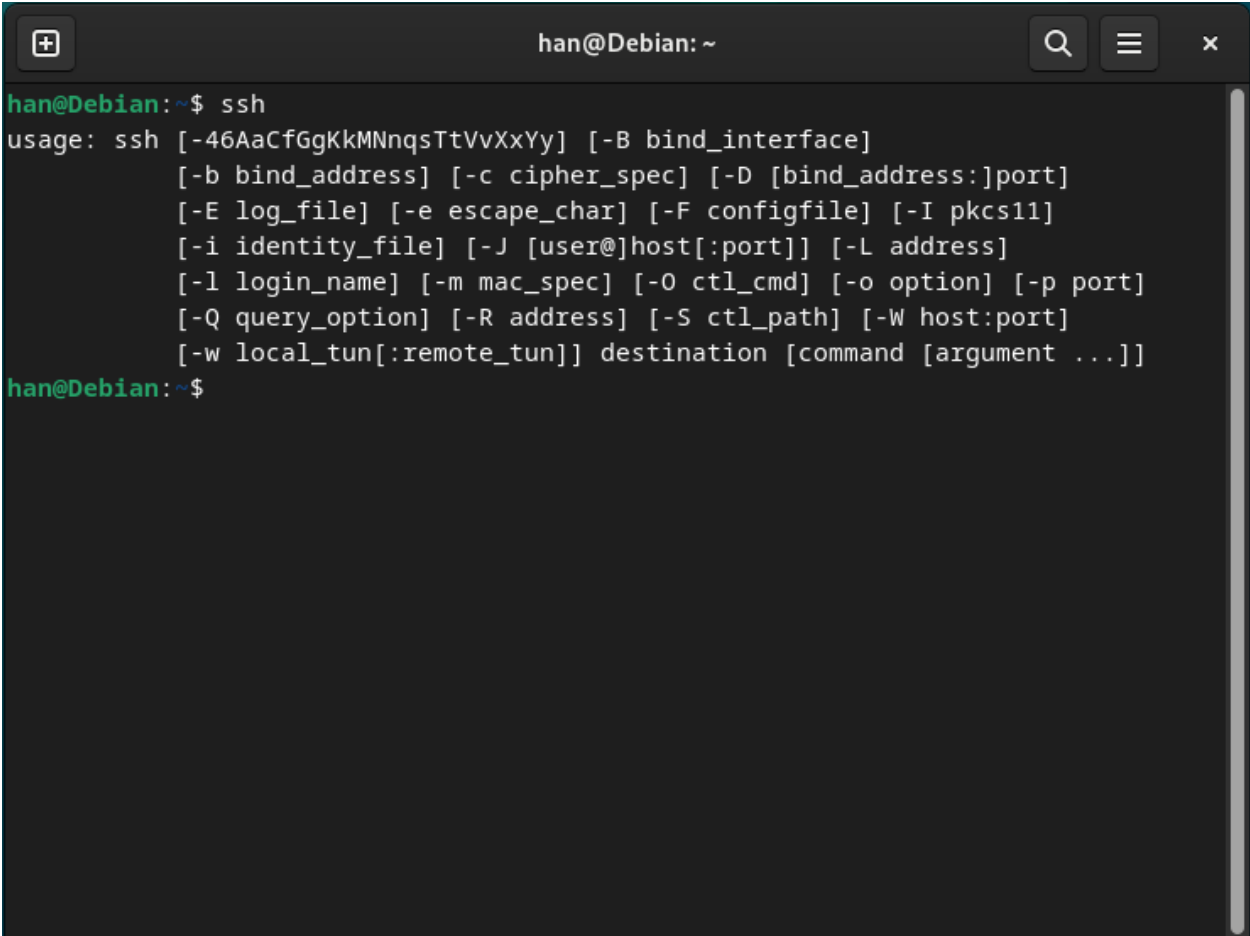
Москва

2024 г.

Цель работы: получить навыки управления пользователями, организации безопасного удаленного доступа к серверу для совместного использования вычислительных ресурсов.

Ход работы:

1. На виртуальной машине Debian уже установлен SSH-сервер:



```
han@Debian: ~  
han@Debian:~$ ssh  
usage: ssh [-46AaCfGgKkMnNqsTtVvXxYy] [-B bind_interface]  
          [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]  
          [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]  
          [-i identity_file] [-J [user@]host[:port]] [-L address]  
          [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]  
          [-Q query_option] [-R address] [-S ctl_path] [-W host:port]  
          [-w local_tun[:remote_tun]] destination [command [argument ...]]  
han@Debian:~$
```

Рисунок 1. SSH-сервер

Открываем файл `/etc/ssh/ssh_config` в редакторе nano и разрешаем использование только протокола SSH 2, ограничиваем доступ только по протоколу IPv4 и запрещаем доступ с пустым паролем:

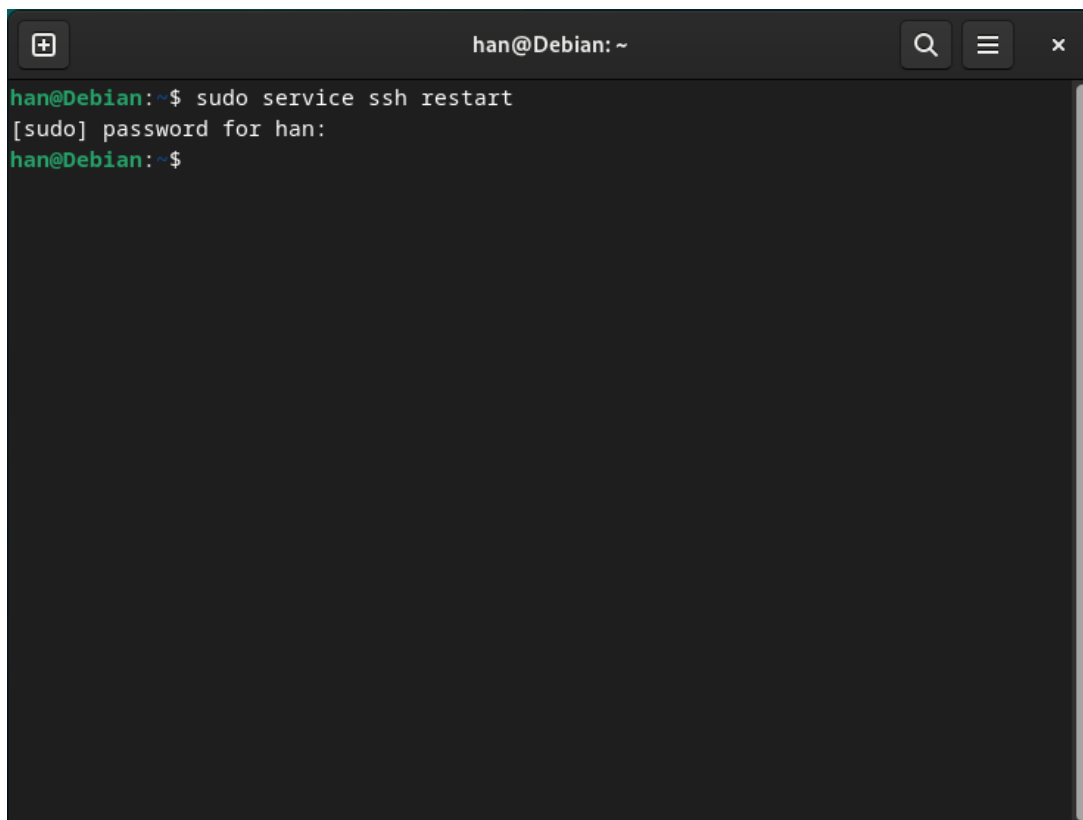
```
han@Debian: ~
GNU nano 7.2 /etc/ssh/ssh_config
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# IdentityFile ~/.ssh/id_ecdsa
# IdentityFile ~/.ssh/id_ed25519
# Port 22
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com
# EscapeChar ~
# Tunnel no
# TunnelDevice any:any
# PermitLocalCommand no
# VisualHostKey no
# ProxyCommand ssh -q -W %h:%p gateway.example.com
# RekeyLimit 1G 1h
# UserKnownHostsFile ~/.ssh/known_hosts.d/%k
SendEnv LANG LC_*
HashKnownHosts yes
GSSAPIAuthentication yes

Protocol 2
AddressFamily inet
PermitEmptyPasswords no

[ Wrote 57 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Рисунок 2. Редактирование файла `ssh_config`

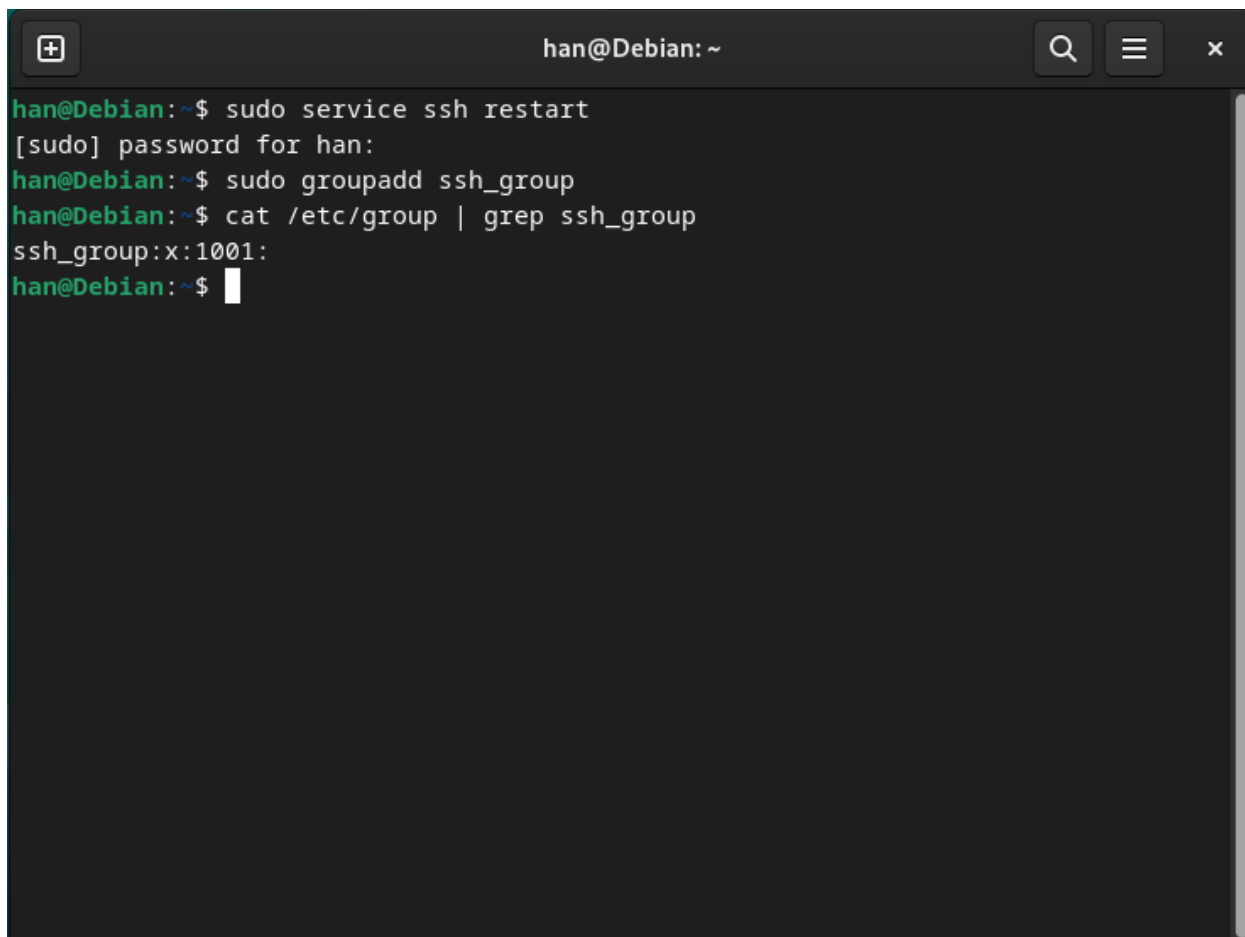
После изменений необходимо сделать перезапуск SSH-сервера:

A terminal window with a dark background. The title bar at the top shows 'han@Debian: ~' and standard window controls (search, menu, close). The terminal content shows a user prompt 'han@Debian:~\$' followed by the command 'sudo service ssh restart'. The next line shows '[sudo] password for han:' followed by a blank line, indicating the password was entered. The final line shows the prompt 'han@Debian:~\$' again, indicating the command has completed.

```
han@Debian:~$ sudo service ssh restart
[sudo] password for han:
han@Debian:~$
```

Рисунок 3. Перезагрузка демона *ssh*

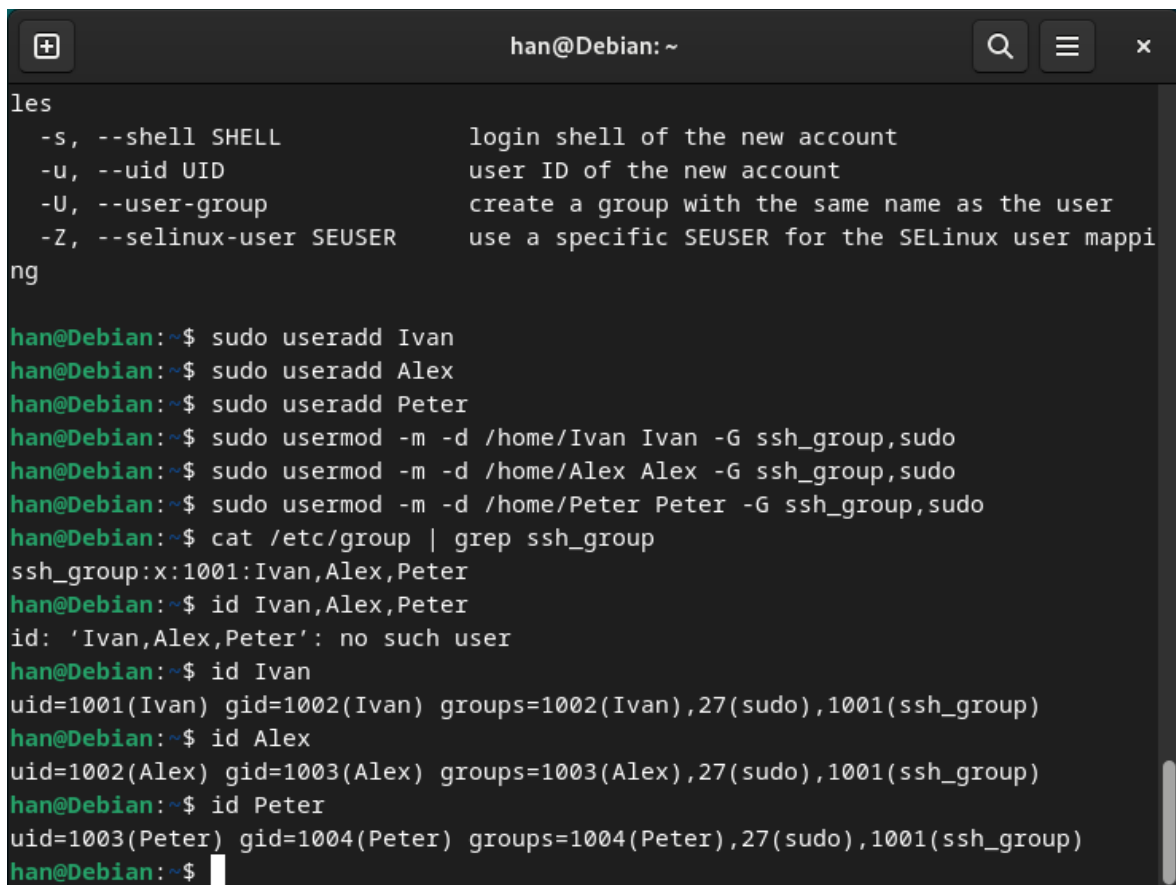
Создадим группу пользователей:

A terminal window titled 'han@Debian: ~' with search, menu, and close buttons in the title bar. The terminal shows the following commands and output:

```
han@Debian:~$ sudo service ssh restart
[sudo] password for han:
han@Debian:~$ sudo groupadd ssh_group
han@Debian:~$ cat /etc/group | grep ssh_group
ssh_group:x:1001:
han@Debian:~$
```

Рисунок 4. Создание группы пользователей

Создадим пользователей Ivan, Alex и Peter, добавим их в группы sudo и ssh_group:

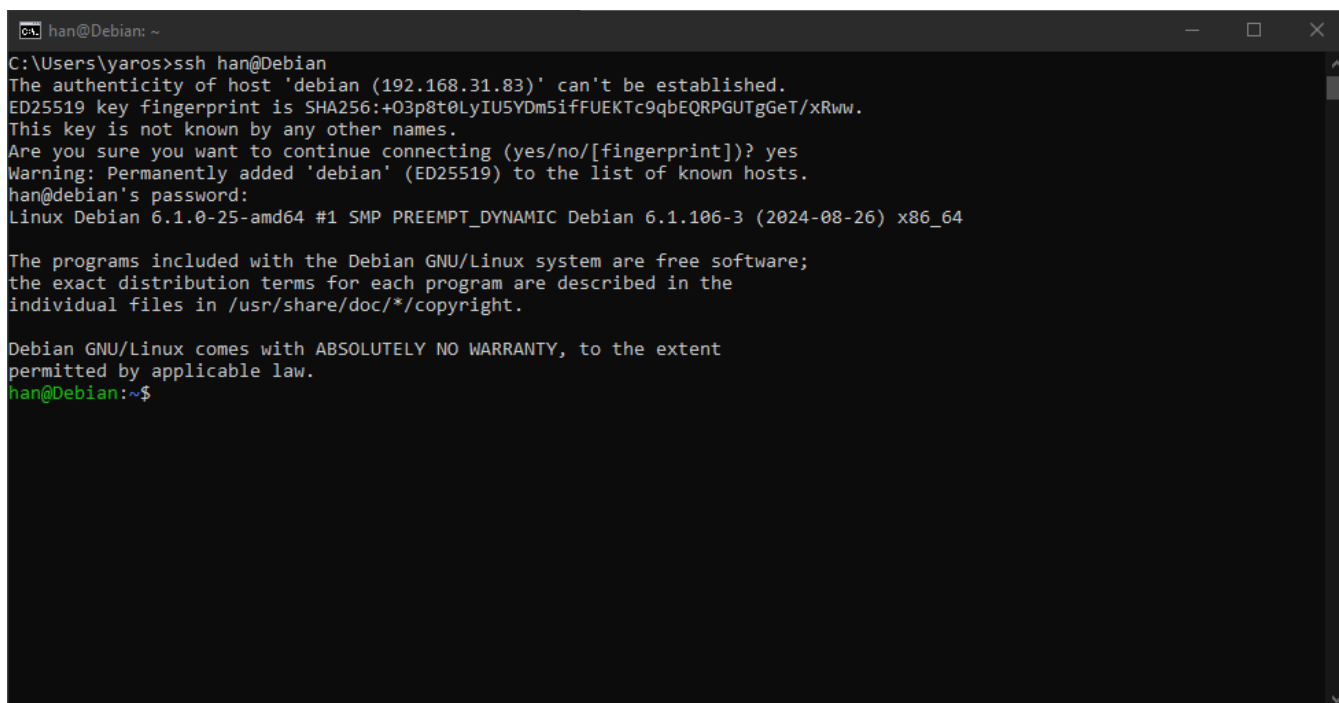


```
han@Debian: ~
les
-s, --shell SHELL          login shell of the new account
-u, --uid UID              user ID of the new account
-U, --user-group           create a group with the same name as the user
-Z, --selinux-user SEUSER  use a specific SEUSER for the SELinux user mapping

han@Debian:~$ sudo useradd Ivan
han@Debian:~$ sudo useradd Alex
han@Debian:~$ sudo useradd Peter
han@Debian:~$ sudo usermod -m -d /home/Ivan Ivan -G ssh_group,sudo
han@Debian:~$ sudo usermod -m -d /home/Alex Alex -G ssh_group,sudo
han@Debian:~$ sudo usermod -m -d /home/Peter Peter -G ssh_group,sudo
han@Debian:~$ cat /etc/group | grep ssh_group
ssh_group:x:1001:Ivan,Alex,Peter
han@Debian:~$ id Ivan,Alex,Peter
id: 'Ivan,Alex,Peter': no such user
han@Debian:~$ id Ivan
uid=1001(Ivan) gid=1002(Ivan) groups=1002(Ivan),27(sudo),1001(ssh_group)
han@Debian:~$ id Alex
uid=1002(Alex) gid=1003(Alex) groups=1003(Alex),27(sudo),1001(ssh_group)
han@Debian:~$ id Peter
uid=1003(Peter) gid=1004(Peter) groups=1004(Peter),27(sudo),1001(ssh_group)
han@Debian:~$
```

Рисунок 5. Создание пользователей, добавление их в группы

2. В качестве клиентской системы выбран встроенный клиент SSH в Windows 10, на которой запущена виртуальная машина Debian. Подключимся к серверу через SSH-клиент:



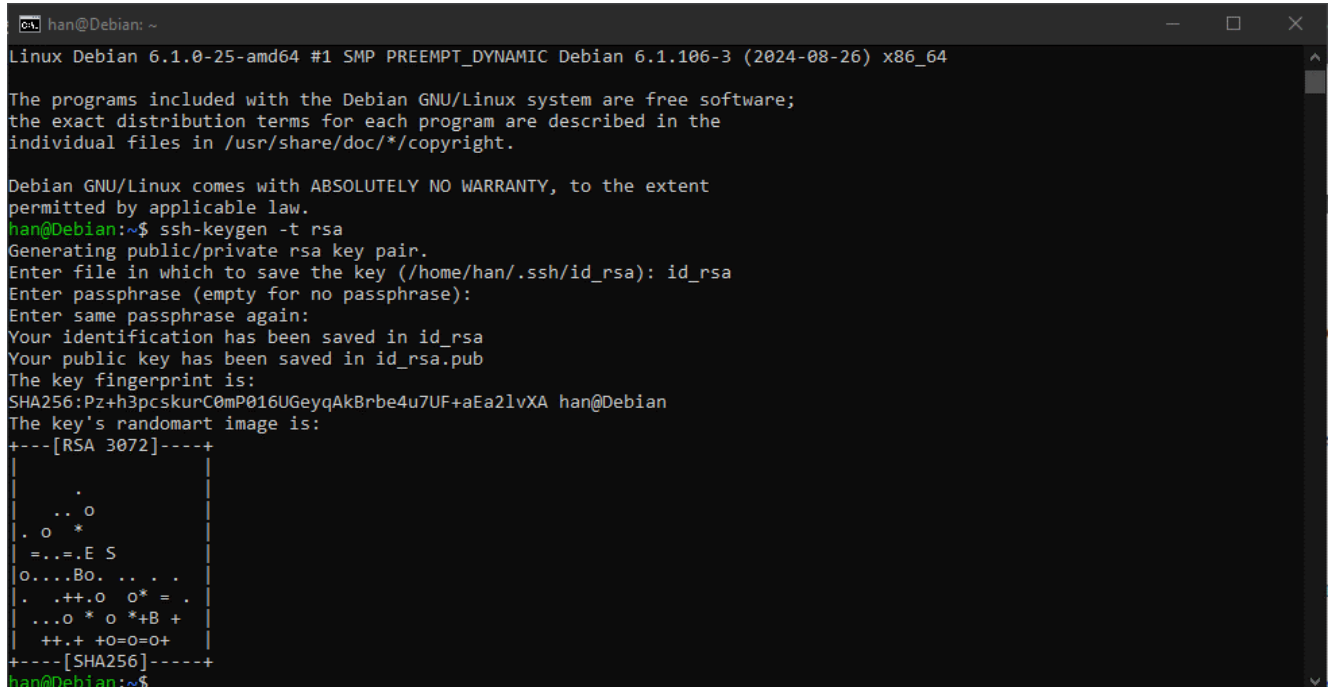
```
C:\Users\Iyaros>ssh han@Debian
The authenticity of host 'debian (192.168.31.83)' can't be established.
ED25519 key fingerprint is SHA256:+03p8t0LyIU5YDm5iffFUEKtc9qbEQRPgUTgGeT/xRww.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'debian' (ED25519) to the list of known hosts.
han@debian's password:
Linux Debian 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
han@Debian:~$
```

Рисунок 6. Подключение к серверу через SSH-клиент

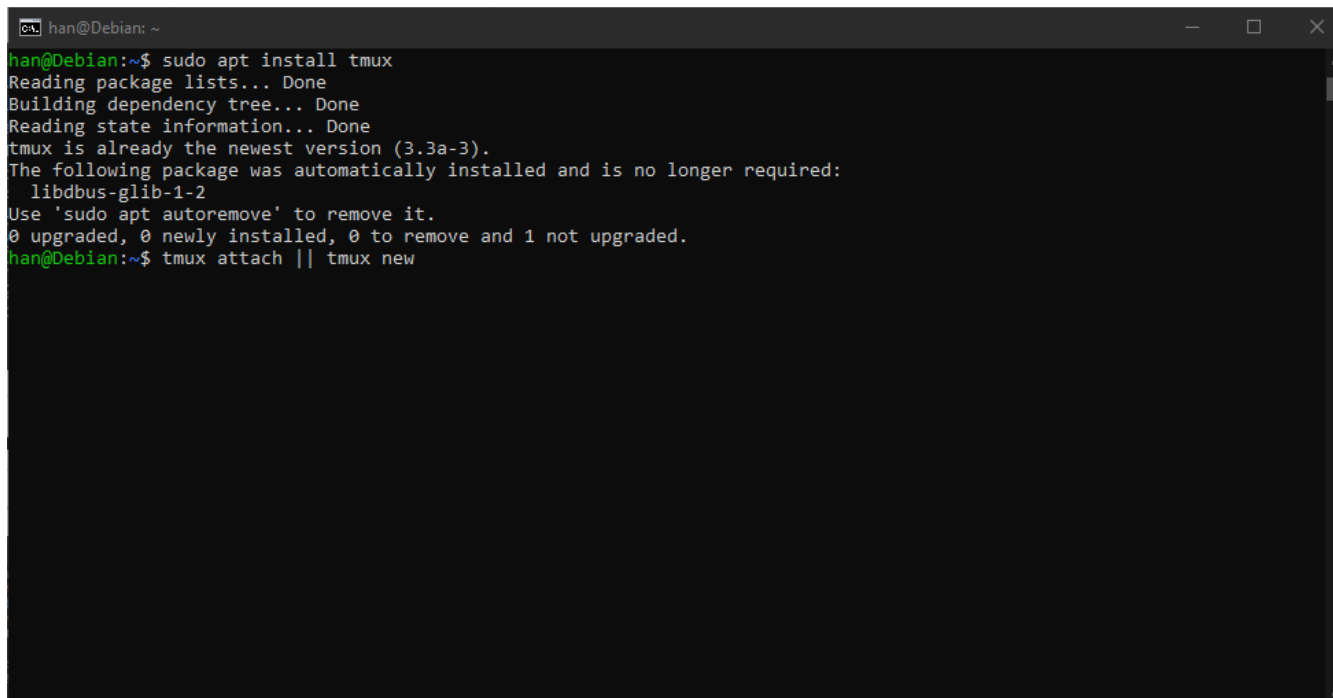
Настроим аутентификацию через RSA-ключ:



```
han@Debian: ~  
Linux Debian 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
han@Debian:~$ ssh-keygen -t rsa  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/han/.ssh/id_rsa): id_rsa  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in id_rsa  
Your public key has been saved in id_rsa.pub  
The key fingerprint is:  
SHA256:Pz+h3pcskurC0mP016UGeyqAkBrbe4u7UF+aEa2lvXA han@Debian  
The key's randomart image is:  
+----[RSA 3072]-----+  
|  
| .  
| .. o  
| . o *  
| =..=.E S  
| o....Bo. . . .  
| . .++..o o* = .  
| ...o * o *+B +  
| ++.+ +o=o=o+  
+----[SHA256]-----+  
han@Debian:~$
```

Рисунок 7. Создание RSA-пары

Далее необходимо скопировать на сервер исполняемый файл и запустить его на выполнение. Установим `tmux` (уже установлен) и попытаемся подключиться к активному окну `tmux`, если не существует – создадим новое:



```
han@Debian: ~  
han@Debian:~$ sudo apt install tmux  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
tmux is already the newest version (3.3a-3).  
The following package was automatically installed and is no longer required:  
  libdbus-glib-1-2  
Use 'sudo apt autoremove' to remove it.  
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.  
han@Debian:~$ tmux attach || tmux new
```

Рисунок 8. Установка `tmux`

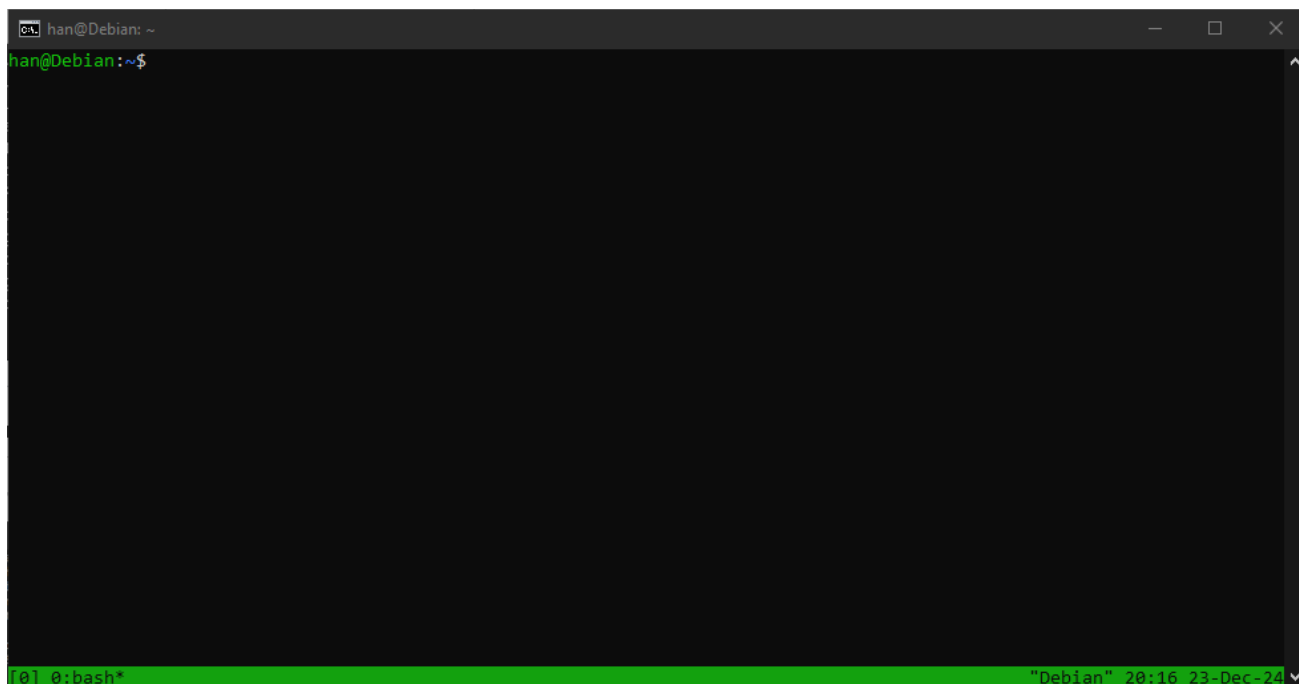


Рисунок 9. Окно tmux

Копирование исполняемого файла с клиентской машины на сервер:

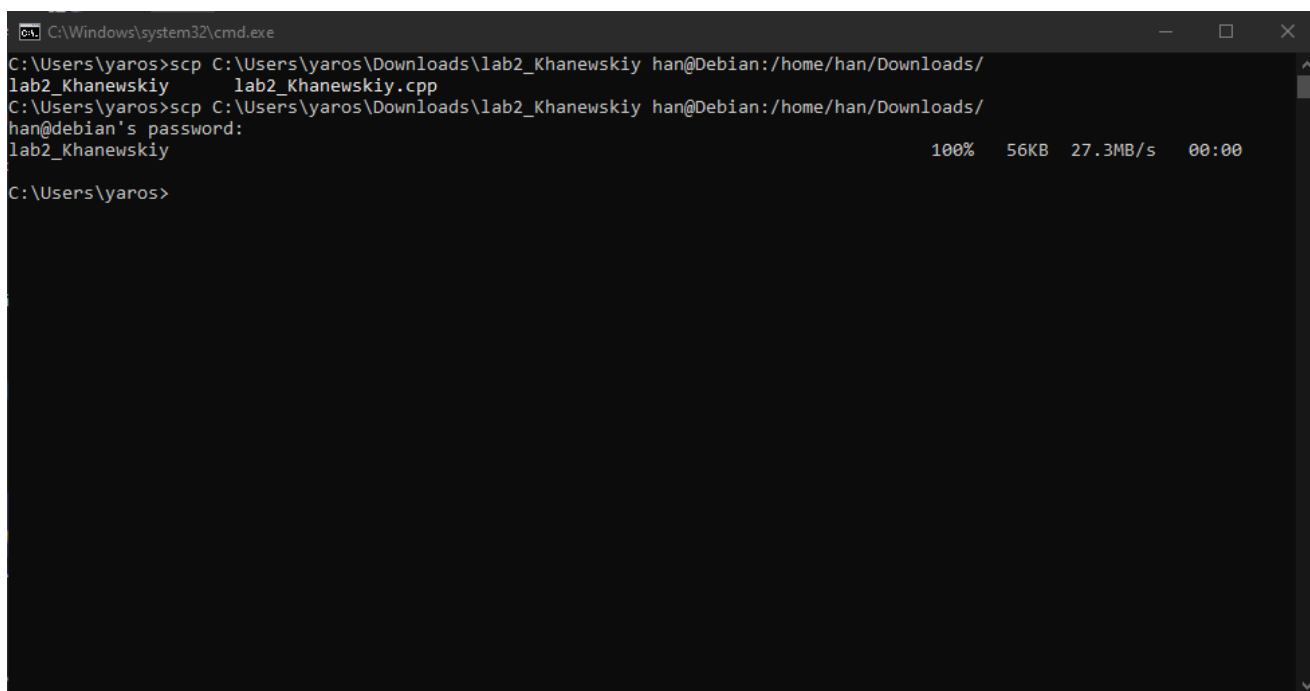


Рисунок 10. Копирование файла на сервер

Запуск исполняемого файла с сервера (в правом окне открыт код программы):


```
han@Debian: ~  
han@Debian:~/Downloads$ ls  
lab2_Khanewskiy  test1  test2  
han@Debian:~/Downloads$ chmod +x lab2_Khanewskiy  
han@Debian:~/Downloads$ ls  
lab2_Khanewskiy  test1  test2  
han@Debian:~/Downloads$ ./lab2_Khanewskiy test1 test2  
test1  
Size of file = 10482435 Bytes  
Summ = 1073756018481283  
Time spent reading from the file = 0.0359686s  
Time spent on summation = 0.0622435s  
Total time = 0.0982183s  
  
test2  
Size of file = 10483008 Bytes  
Summ = 1073588978462901  
Time spent reading from the file = 0.0368096s  
Time spent on summation = 0.0630553s  
Total time = 0.099865s  
  
han@Debian:~/Downloads$  
  
/home/han/Desktop/lab2_Khanewskiy.cpp  
std::chrono::duration<double> summDuration = summEnd ->  
  
std::chrono::duration<double> totalDuration = summEnd ->  
  
//coutMutex.lock();  
std::cout << fileName << "\n";  
std::cout << "Size of file = " << fileSize << " Bytes\>  
std::cout << "Summ = "<< summ << "\n";  
std::cout << "Time spent reading from the file = " << >  
std::cout << "Time spent on summation = " << summDurat>  
std::cout << "Total time = " << totalDuration.count() >  
//coutMutex.unlock();  
}  
  
int main(int argc, char* argv[]){  
    std::vector<std::thread> threads;  
    for (int i = 1; i < argc; i++){  
        threads.push_back(std::thread(summ, argv[i]));  
    }  
    for (int i = 1; i < argc; i++){  
        threads[i-1].join();  
    }  
    return 0;  
}
```

Рисунок 11. Запуск исполняемого файла