

Отчёт по лабораторной работе 11

Настройка безопасного удалённого доступа по протоколу SSH

Суннатилло Махмудов

Содержание

1	Цель работы	5
2	Теоретические сведения	6
2.1	Протокол SSH	6
2.2	Архитектура SSH	6
2.3	Основные параметры конфигурации SSH	7
3	Выполнение лабораторной работы	8
3.1	Запрет удалённого доступа по SSH для пользователя root	8
3.2	Ограничение списка пользователей для удалённого доступа по SSH	11
3.3	Настройка дополнительных портов для удалённого доступа по SSH	14
3.4	Настройка удалённого доступа по SSH по ключу	17
3.5	Организация туннелей SSH и перенаправление TCP-портов	19
3.6	Запуск консольных приложений через SSH	20
3.7	Запуск графических приложений через SSH (X11 Forwarding)	21
3.8	Внесение изменений в настройки внутреннего окружения виртуальной машины	22
4	Вывод	23
5	Контрольные вопросы	24
6	Список литературы	27

Список иллюстраций

3.1	Попытка подключения по SSH под пользователем root до разрешения	9
3.2	Изменение параметра PermitRootLogin на yes	9
3.3	Успешное подключение к серверу под пользователем root	10
3.4	Возврат параметра PermitRootLogin в состояние no	10
3.5	Неудачная попытка входа после запрета доступа root	11
3.6	Успешное подключение по SSH под пользователем smahmudov	11
3.7	Добавление параметра AllowUsers vagrant	12
3.8	Неудачная попытка подключения при ограничении AllowUsers vagrant	12
3.9	Расширение списка разрешённых пользователей	13
3.10	Успешное подключение по SSH после добавления пользователя smahmudov	13
3.11	Добавление второго порта SSH в конфигурацию	14
3.12	Сообщение об ошибке при запуске sshd на новом порту	15
3.13	Успешное прослушивание портов 22 и 2022 службой sshd	16
3.14	Успешное подключение по стандартному порту 22	16
3.15	Разрешение аутентификации по ключу	17
3.16	Копирование ключа на сервер и успешная авторизация без пароля	18
3.17	Проверка активных соединений после создания туннеля	19
3.18	Результат перенаправления порта 8080 в браузере	19
3.19	Просмотр списка файлов на сервере через SSH	20
3.20	Просмотр почты пользователя через SSH	20
3.21	Разрешение X11 Forwarding на сервере	21
3.22	Неудачная попытка запуска графического приложения через SSH	22
3.23	Скрипт автоматической настройки и перезапуска SSH	22

Список таблиц

1 Цель работы

Приобретение практических навыков по настройке удалённого доступа к серверу с помощью SSH.

2 Теоретические сведения

2.1 Протокол SSH

SSH (Secure Shell) — это сетевой протокол, обеспечивающий безопасный удалённый доступ к серверу. Он заменяет небезопасные методы подключения, такие как Telnet и rlogin, предоставляя зашифрованный канал связи между клиентом и сервером.

Основные функции SSH: * Защищённая аутентификация пользователя; * Шифрование всего сетевого трафика; * Контроль целостности данных; * Возможность туннелирования и переадресации портов.

2.2 Архитектура SSH

- **SSH-клиент** — инициирует соединение с сервером и выполняет команды.
- **SSH-сервер** — ожидает входящих подключений и аутентифицирует пользователей.
- **SSH-ключи** — пара файлов (открытый и закрытый ключ), используемых для входа без пароля.

Схема работы: 1. Клиент устанавливает соединение с сервером.
2. Стороны обмениваются ключами и договариваются о параметрах шифрования.

3. Выполняется аутентификация пользователя.
4. Создаётся защищённый канал передачи данных.

2.3 Основные параметры конфигурации SSH

Файл конфигурации сервера — `/etc/ssh/sshd_config`.

Ключевые параметры: * **Port** — задаёт номер порта SSH (по умолчанию 22);

* **PermitRootLogin** — разрешает или запрещает вход под root;

* **AllowUsers / DenyUsers** — ограничивает список пользователей;

* **PasswordAuthentication** — включает или выключает вход по паролю;

* **PubkeyAuthentication** — разрешает аутентификацию по ключу;

* **X11Forwarding** — включает возможность перенаправления графического интерфейса.

3 Выполнение лабораторной работы

3.1 Запрет удалённого доступа по SSH для пользователя root

1. На сервере был задан пароль для пользователя **root**, чтобы обеспечить возможность первоначального входа при необходимости:

```
sudo -i  
passwd root
```

2. В отдельном терминале на сервере был запущен мониторинг системных событий в реальном времени с помощью команды:

```
sudo -i  
journalctl -x -f
```

3. С клиента предпринята попытка подключения к серверу по SSH под пользователем **root**:

```
ssh root@smahmudov.net
```

Система запрашивала пароль, однако вход не был выполнен — сервер отклонил попытку аутентификации. Это указывает на то, что по умолчанию вход под пользователем **root** через SSH запрещён.


```
[smahmudov@client.smahmudov.net ~]$ ssh root@smahmudov.net
The authenticity of host 'smahmudov.net (192.168.1.1)' can't be established.
ED25519 key fingerprint is SHA256:n0w0vphoobfpKiXoFhmCpcTAKAVt01RYFK5Kxnw0XTU.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: [server.smahmudov.net]:2022
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'smahmudov.net' (ED25519) to the list of known hosts.
root@smahmudov.net's password:
Permission denied, please try again.
root@smahmudov.net's password:
Permission denied, please try again.
root@smahmudov.net's password:
root@smahmudov.net: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.1: Попытка подключения по SSH под пользователем root до разрешения

- Для временного разрешения входа пользователь **root** в файл конфигурации **/etc/ssh/sshd_config** было добавлено или изменено значение параметра:

PermitRootLogin yes

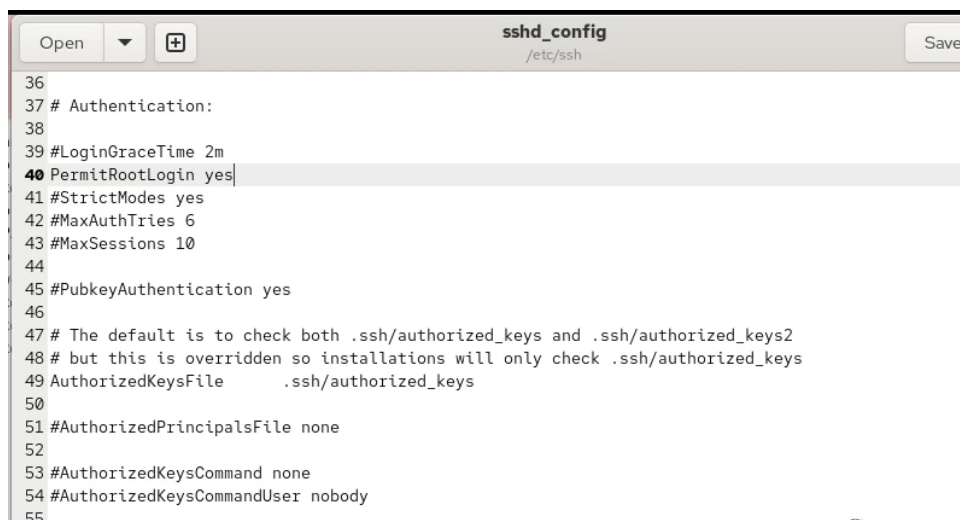


Рис. 3.2: Изменение параметра PermitRootLogin на yes

- После сохранения изменений был перезапущен сервис SSH:

systemctl restart sshd

Повторная попытка подключения к серверу под пользователем **root** завершилась успешно, что подтверждает корректность внесённых изменений.

```

[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh root@smahmudov.net
root@smahmudov.net's password:
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/

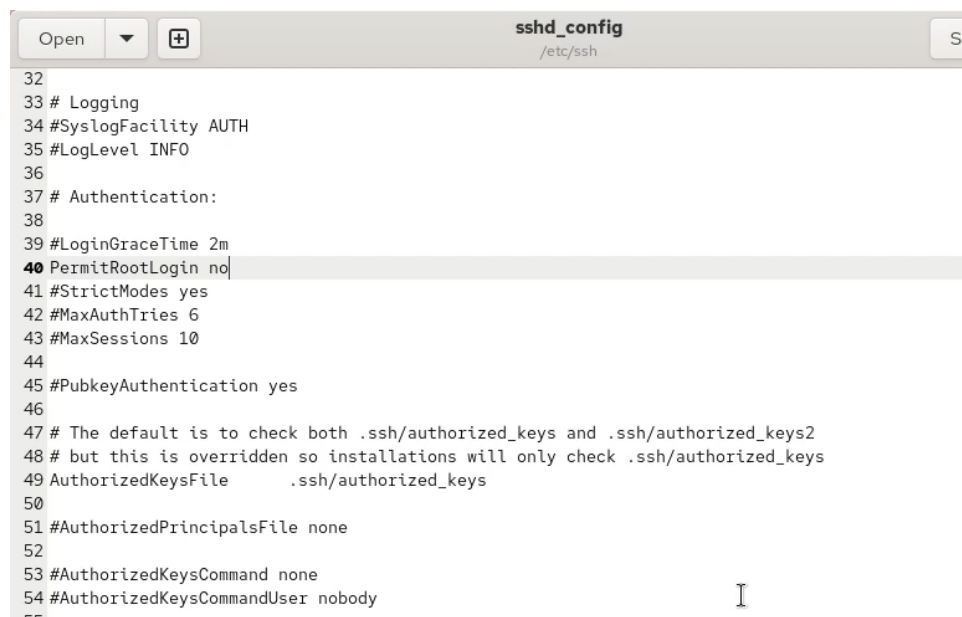
Last failed login: Sat Oct 18 06:39:23 UTC 2025 from 192.168.1.30 on ssh:notty
There were 3 failed login attempts since the last successful login.
Last login: Sat Oct 18 06:38:13 2025
root@server:~#
root@server:~#
logout
Connection to smahmudov.net closed.
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$

```

Рис. 3.3: Успешное подключение к серверу под пользователем root

6. Для восстановления безопасной конфигурации параметр был возвращён в исходное состояние:

PermitRootLogin no



```

32
33 # Logging
34 #SyslogFacility AUTH
35 #LogLevel INFO
36
37 # Authentication:
38
39 #LoginGraceTime 2m
40 PermitRootLogin no
41 #StrictModes yes
42 #MaxAuthTries 6
43 #MaxSessions 10
44
45 #PubkeyAuthentication yes
46
47 # The default is to check both .ssh/authorized_keys and .ssh/authorized_keys2
48 # but this is overridden so installations will only check .ssh/authorized_keys
49 AuthorizedKeysFile .ssh/authorized_keys
50
51 #AuthorizedPrincipalsFile none
52
53 #AuthorizedKeysCommand none
54 #AuthorizedKeysCommandUser nobody

```

Рис. 3.4: Возврат параметра PermitRootLogin в состояние no

7. После перезапуска SSH-службы повторная попытка входа по SSH под пользователем **root** вновь завершилась неудачей, что свидетельствует о срабатывании ограничения.

```

[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh root@smahmudov.net
root@smahmudov.net's password:
Permission denied, please try again.
root@smahmudov.net's password:
Permission denied, please try again.
root@smahmudov.net's password:
root@smahmudov.net: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$

```

Рис. 3.5: Неудачная попытка входа после запрета доступа root

3.2 Ограничение списка пользователей для удалённого доступа по SSH

1. С клиента была предпринята попытка подключения к серверу по SSH под пользователем **smahmudov**:

```
ssh smahmudov@server.smahmudov.net
```

Подключение было успешно выполнено, что свидетельствует о том, что на данный момент вход по SSH для всех пользователей разрешён.

```

[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net
The authenticity of host 'server.smahmudov.net (192.168.1.1)' can't be established.
ED25519 key fingerprint is SHA256:n0w0vphoobfpKiXoFhmCpcTAKAVt01RYFK5Kxnw0XTU.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:1: [server.smahmudov.net]:2022
  ~/.ssh/known_hosts:4: smahmudov.net
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'server.smahmudov.net' (ED25519) to the list of known hosts.
smahmudov@server.smahmudov.net's password:
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/

Last login: Sat Oct 18 06:36:09 2025
[smahmudov@server.smahmudov.net ~]$

```

Рис. 3.6: Успешное подключение по SSH под пользователем smahmudov

2. На сервере был открыт для редактирования файл конфигурации **/etc/ssh/sshd_config**, в который была добавлена строка:

```
AllowUsers vagrant
```

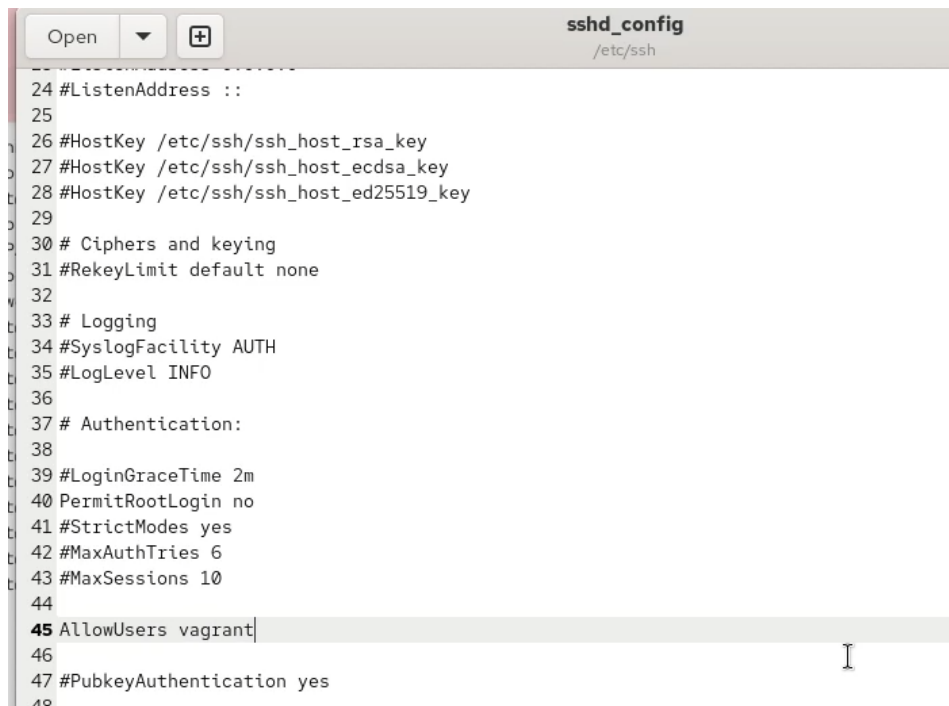


Рис. 3.7: Добавление параметра AllowUsers vagrant

3. После сохранения изменений служба SSH была перезапущена командой:

```
systemctl restart sshd
```

Повторная попытка подключения с клиента под пользователем **smahmudov** завершилась неудачей. Сервер отказал в доступе, так как в списке разрешённых пользователей указан только **vagrant**.

```

[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net
smahmudov@server.smahmudov.net's password:
Permission denied, please try again.
smahmudov@server.smahmudov.net's password:
Permission denied, please try again.
smahmudov@server.smahmudov.net's password:
smahmudov@server.smahmudov.net: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ █

```

Рис. 3.8: Неудачная попытка подключения при ограничении AllowUsers vagrant

4. В конфигурационный файл **/etc/ssh/sshd_config** было внесено изменение, разрешающее доступ также пользователю **smahmudov**:

```
AllowUsers vagrant smahmudov
```

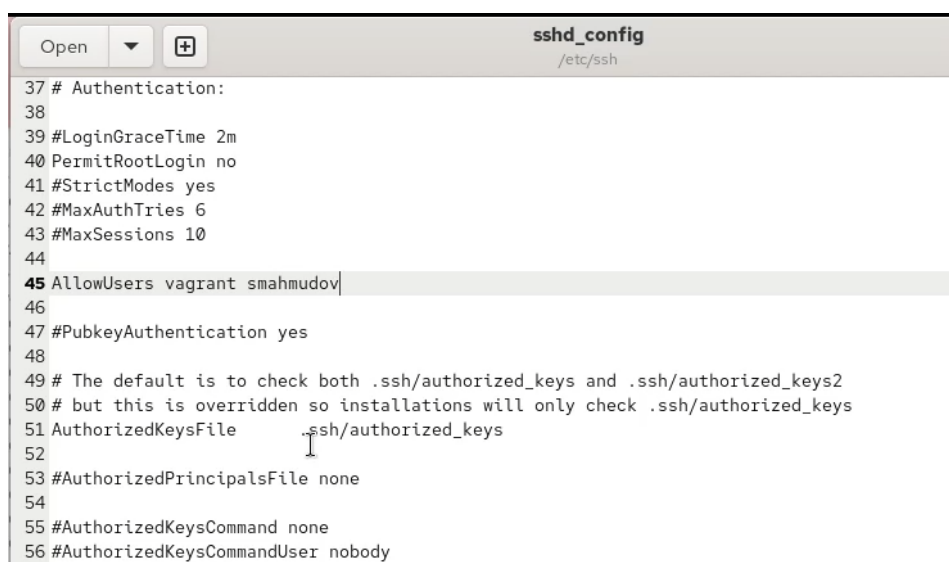


Рис. 3.9: Расширение списка разрешённых пользователей

5. После сохранения изменений и перезапуска SSH-службы:

```
systemctl restart sshd
```

Повторная попытка подключения с клиента под пользователем **smahmudov** прошла успешно, что подтверждает корректность настройки параметра **AllowUsers**.

```
[smahmudov@client.smahmudov.net ~]$  
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net  
smahmudov@server.smahmudov.net's password:  
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/  
  
Last failed login: Sat Oct 18 06:44:18 UTC 2025 from 192.168.1.30 on ssh:notty  
There were 3 failed login attempts since the last successful login.  
Last login: Sat Oct 18 06:43:20 2025 from 192.168.1.30  
[smahmudov@server.smahmudov.net ~]$  
[smahmudov@server.smahmudov.net ~]$  
logout  
Connection to server.smahmudov.net closed.  
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.10: Успешное подключение по SSH после добавления пользователя smahmudov

3.3 Настройка дополнительных портов для удалённого доступа по SSH

1. На сервере в файле конфигурации `/etc/ssh/sshd_config` были добавлены строки:

Port 22

Port 2022

Это позволяет службе **sshd** принимать подключения по двум портам, обеспечивая резервный канал в случае ошибки конфигурации или блокировки стандартного порта.

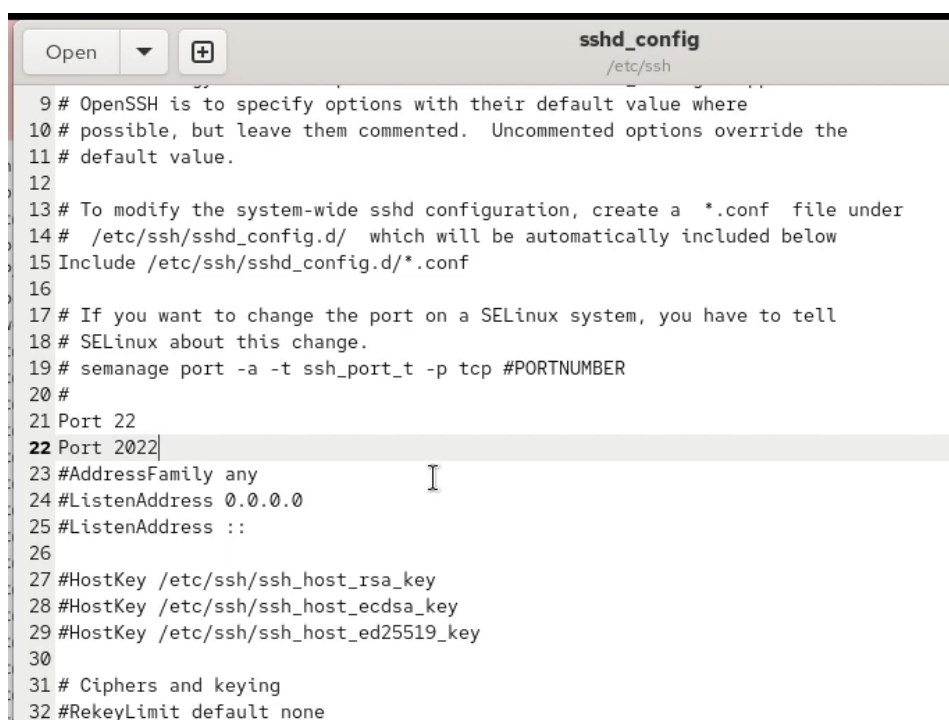


Рис. 3.11: Добавление второго порта SSH в конфигурацию

2. После сохранения изменений была выполнена перезагрузка службы SSH:
`systemctl restart sshd`
3. При просмотре расширенного статуса службы **sshd** команда `systemctl`

`status -l sshd` показала сообщения об ошибке:

error: Bind to port 2022 failed: Permission denied

Это означает, что SELinux не разрешил процессу **sshd** использовать новый порт 2022, поскольку для него не была назначена соответствующая метка безопасности.

```
[root@server.smahudov.net ~]#  
[root@server.smahudov.net ~]# systemctl status -l sshd  
● sshd.service - OpenSSH server daemon  
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)  
   Active: active (running) since Sat 2025-10-18 06:46:16 UTC; 11s ago  
  Invocation: cb77408fbd984b0fb4b0e57e4d54fe8  
     Docs: man:sshd(8)  
           man:sshd_config(5)  
  Main PID: 11049 (sshd)  
    Tasks: 1 (limit: 10381)  
  Memory: 1M (peak: 1.2M)  
    CPU: 4ms  
   OGroup: /system.slice/ssh.service  
           └─11049 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"  
  
Oct 18 06:46:16 server.smahudov.net systemd[1]: Starting sshd.service - OpenSSH server daemon...  
Oct 18 06:46:16 server.smahudov.net (sshd)[11049]: sshd.service: Referenced but unset environment variable evaluates to an empty string: OPTIONS  
Oct 18 06:46:16 server.smahudov.net sshd[11049]: error: Bind to port 2022 on 0.0.0.0 failed: Permission denied.  
Oct 18 06:46:16 server.smahudov.net sshd[11049]: error: Bind to port 2022 on :: failed: Permission denied.  
Oct 18 06:46:16 server.smahudov.net sshd[11049]: Server listening on 0.0.0.0 port 22.  
Oct 18 06:46:16 server.smahudov.net sshd[11049]: Server listening on :: port 22.  
Oct 18 06:46:16 server.smahudov.net systemd[1]: Started sshd.service - OpenSSH server daemon.  
[root@server.smahudov.net ~]#
```

Рис. 3.12: Сообщение об ошибке при запуске sshd на новом порту

4. Для разрешения работы **sshd** на порту 2022 была выполнена команда:

```
semanage port -a -t ssh_port_t -p tcp 2022
```

После этого порт был добавлен в список разрешённых для типа **ssh_port_t**.

5. Далее в настройках брандмауэра был открыт новый порт для протокола TCP:

```
firewall-cmd --add-port=2022/tcp
```

```
firewall-cmd --add-port=2022/tcp --permanent
```

6. После повторного перезапуска службы **sshd** её статус показал, что сервер успешно слушает подключения одновременно на портах **22** и **2022**.

```

[root@server.smahmudov.net ~]# semanage port -a -t ssh_port_t -p tcp 2022
[root@server.smahmudov.net ~]# firewall-cmd --add-port=2022/tcp
success
[root@server.smahmudov.net ~]# firewall-cmd --add-port=2022/tcp --permanent
success
[root@server.smahmudov.net ~]# systemctl restart sshd
[root@server.smahmudov.net ~]# systemctl status -l sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
   Active: active (running) since Sat 2025-10-18 06:47:43 UTC; 2s ago
   Invocation: 31af615e6b634559b513a92f5be6d96c
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 11267 (sshd)
     Tasks: 1 (limit: 10381)
    Memory: 1M (peak: 1.2M)
       CPU: 5ms
   CGroup: /system.slice/ssh.service
           └─11267 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Oct 18 06:47:43 server.smahmudov.net systemd[1]: Starting sshd.service - OpenSSH server daemon...
Oct 18 06:47:43 server.smahmudov.net (sshd)[11267]: sshd.service: Referenced but unset environment variable evaluates to an empty string: OPTIONS
Oct 18 06:47:43 server.smahmudov.net sshd[11267]: Server listening on 0.0.0.0 port 2022.
Oct 18 06:47:43 server.smahmudov.net sshd[11267]: Server listening on :: port 2022.
Oct 18 06:47:43 server.smahmudov.net sshd[11267]: Server listening on 0.0.0.0 port 22.
Oct 18 06:47:43 server.smahmudov.net sshd[11267]: Server listening on :: port 22.
Oct 18 06:47:43 server.smahmudov.net systemd[1]: Started sshd.service - OpenSSH server daemon.
[root@server.smahmudov.net ~]#

```

Рис. 3.13: Успешное прослушивание портов 22 и 2022 службой sshd

7. С клиента выполнено подключение по стандартному порту 22 к серверу под пользователем **smahmudov**. Подключение прошло успешно, после чего был получен доступ администратора с помощью команды **sudo -i**.

```

[smahmudov@client.smahmudov.net ~]#
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net
smahmudov@server.smahmudov.net's password:
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/

Last login: Sat Oct 18 06:44:45 2025 from 192.168.1.30
[smahmudov@server.smahmudov.net ~]$ sudo -i
[sudo] password for smahmudov:
[root@server.smahmudov.net ~]#
logout
[smahmudov@server.smahmudov.net ~]$
logout
Connection to server.smahmudov.net closed.
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh -p2022 smahmudov@server.smahmudov.net
smahmudov@server.smahmudov.net's password:
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/

Last login: Sat Oct 18 06:50:26 2025 from 192.168.1.30
[smahmudov@server.smahmudov.net ~]$ sudo -i
[sudo] password for smahmudov:
[root@server.smahmudov.net ~]#
logout
[smahmudov@server.smahmudov.net ~]$
logout
Connection to server.smahmudov.net closed.
[smahmudov@client.smahmudov.net ~]$

```

Рис. 3.14: Успешное подключение по стандартному порту 22

8. Затем было выполнено подключение с указанием нового порта **2022**:


```
ssh -p2022 smahmudov@server.smahmudov.net
```

Подключение также прошло успешно, что подтверждает корректную настройку дополнительного SSH-порта.

3.4 Настройка удалённого доступа по SSH по ключу

1. На сервере в файле конфигурации `/etc/ssh/sshd_config` был включён параметр, разрешающий аутентификацию по ключу:

`PubkeyAuthentication yes`

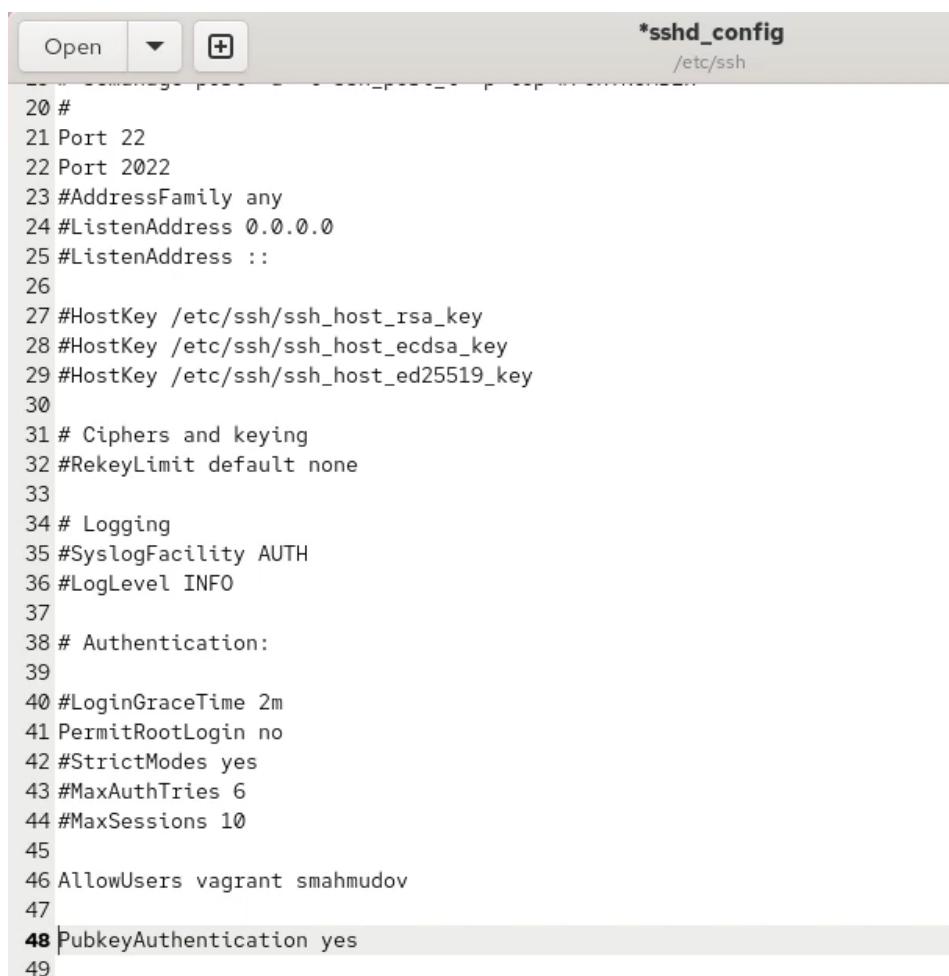


Рис. 3.15: Разрешение аутентификации по ключу

2. После сохранения изменений служба SSH была перезапущена для приме-

нения новых настроек:

```
systemctl restart sshd
```

3. На клиенте была создана пара SSH-ключей при помощи команды **ssh-keygen**. После выполнения команды были сгенерированы два файла:

- закрытый ключ `~/.ssh/id_rsa`
- открытый ключ `~/.ssh/id_rsa.pub`

4. С помощью команды **ssh-copy-id** открытый ключ был скопирован на сервер:

```
ssh-copy-id smahmudov@server.smahmudov.net
```

После этого пользователь смог входить на сервер без запроса пароля, что подтверждает успешную настройку аутентификации по ключу.

```
|oX .. o      |
|X.=+o  E o    |
|,==+o.  S .   |
|oo.=.o.o .    |
|++= . =..     |
|o+.o . +. .   |
|... . .+.     |
+----[SHA256]-----+
[smahmudov@client.smahmudov.net ~]$ ssh-copy-id smahmudov@server.smahmudov.net
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ssh-add -L
/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
smahmudov@server.smahmudov.net's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'smahmudov@server.smahmudov.net'"
and check to make sure that only the key(s) you wanted were added.

[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net
Web console: https://server.smahmudov.net:9090/ or https://10.0.2.15:9090/

Last login: Sat Oct 18 06:51:35 2025 from 192.168.1.30
[smahmudov@server.smahmudov.net ~]$
logout
Connection to server.smahmudov.net closed.
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.16: Копирование ключа на сервер и успешная авторизация без пароля

3.5 Организация туннелей SSH и перенаправление

TCP-портов

1. На клиенте была выполнена проверка активных TCP-соединений:

```
lsof | grep TCP
```

2. Для организации туннеля между клиентом и сервером был перенаправлен порт 80 сервера на локальный порт 8080:

```
ssh -fNL 8080:localhost:80 smahmudov@server.smahmudov.net
```

3. После выполнения команды на клиенте появились новые соединения, подтверждающие создание SSH-туннеля.

```
[smahmudov@client.smahmudov.net ~]$  
[smahmudov@client.smahmudov.net ~]$ lsof | grep TCP  
[smahmudov@client.smahmudov.net ~]$ ssh -fNL 8080:localhost:80 smahmudov@server.smahmudov.net  
[smahmudov@client.smahmudov.net ~]$ lsof | grep TCP  
ssh      11504      smahmudov    3u  IPv4      88240      0t0      TCP  
client.smahmudov.net:39610->server.smahmudov.net:ssh (ESTABLISHED)  
ssh      11504      smahmudov    4u  IPv6      88245      0t0      TCP  
localhost:webcache (LISTEN)  
ssh      11504      smahmudov    5u  IPv4      88246      0t0      TCP  
localhost:webcache (LISTEN)  
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.17: Проверка активных соединений после создания туннеля

4. При открытии в браузере страницы **localhost:8080** отобразилось приветствие сервера, что подтверждает успешное перенаправление порта.

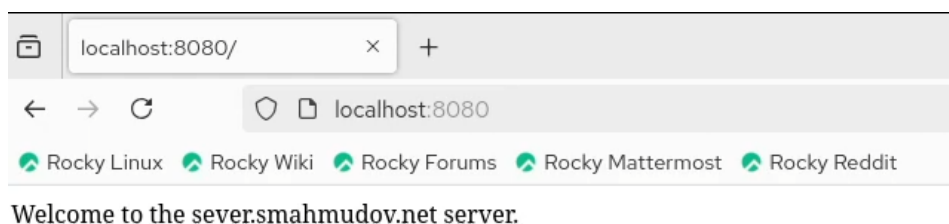


Рис. 3.18: Результат перенаправления порта 8080 в браузере

3.6 Запуск консольных приложений через SSH

1. С клиента был выполнен запрос имени узла сервера:

```
ssh smahmudov@server.smahmudov.net hostname
```

Результатом стал вывод имени хоста **server.smahmudov.net**.

2. Далее был просмотрен список файлов в домашнем каталоге пользователя:

```
ssh smahmudov@server.smahmudov.net ls -Al
```

```
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net hostname
server.smahmudov.net
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net ls -Al
total 56
-rw-----. 1 smahmudov smahmudov 345 Oct 18 06:50 .bash_history
-rw-r--r--. 1 smahmudov smahmudov 18 Oct 29 2024 .bash_logout
-rw-r--r--. 1 smahmudov smahmudov 144 Oct 29 2024 .bash_profile
-rw-r--r--. 1 smahmudov smahmudov 549 Sep 7 13:30 .bashrc
drwx-----. 11 smahmudov smahmudov 4096 Sep 11 14:45 .cache
drwx-----. 10 smahmudov smahmudov 4096 Sep 14 07:45 .config
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Desktop
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Documents
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Downloads
drwx-----. 4 smahmudov smahmudov 32 Sep 7 13:33 .local
drwx-----. 5 smahmudov smahmudov 4096 Oct 13 13:24 Maildir
drwxr-xr-x. 5 smahmudov smahmudov 54 Sep 11 14:45 .mozilla
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Music
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Pictures
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Public
drwx-----. 2 smahmudov smahmudov 29 Oct 18 06:54 .ssh
drwxr-xr-x. 2 smahmudov smahmudov 6 Sep 7 13:33 Templates
```

Рис. 3.19: Просмотр списка файлов на сервере через SSH

3. Проверка почтового ящика показала наличие сообщений в каталоге

~/Maildir/:

```
[smahmudov@client.smahmudov.net ~]$
[smahmudov@client.smahmudov.net ~]$ ssh smahmudov@server.smahmudov.net MAIL=~/Maildir mail
s-nail version v14.9.24. Type '?' for help
/home/smahmudov/Maildir: 3 messages 1 unread
 1 smahmudov 2025-10-10 05:29 18/651 "test1"
 2 smahmudov@client.sma 2025-10-13 12:40 21/834 "LMTP test"
▶U 3 smahmudov 2025-10-13 13:24 22/829 "test hello"
q
Held 3 messages in /home/smahmudov/Maildir
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.20: Просмотр почты пользователя через SSH

3.7 Запуск графических приложений через SSH (X11 Forwarding)

1. На сервере в файле `/etc/ssh/sshd_config` был разрешён показ графических интерфейсов через параметр:

X11Forwarding yes

```
90 # be allowed through the KbdInteractiveAuthentication and
91 # PasswordAuthentication. Depending on your PAM configuration,
92 # PAM authentication via KbdInteractiveAuthentication may bypass
93 # the setting of "PermitRootLogin prohibit-password".
94 # If you just want the PAM account and session checks to run without
95 # PAM authentication, then enable this but set PasswordAuthentication
96 # and KbdInteractiveAuthentication to 'no'.
97 # WARNING: 'UsePAM no' is not supported in this build and may cause several
98 # problems.
99 #UsePAM no
100
101 #AllowAgentForwarding yes
102 #AllowTcpForwarding yes
103 #GatewayPorts no
104 X11Forwarding yes
105 #X11DisplayOffset 10
106 #X11UseLocalhost yes
107 #PermitTTY yes
108 #PrintMotd yes
109 #PrintLastLog yes
110 #TCPKeepAlive yes
111 #PermitUserEnvironment no
112 #Compression delayed
113 #ClientAliveInterval 0
114 #ClientAliveCountMax 3
```

Рис. 3.21: Разрешение X11 Forwarding на сервере

2. После перезапуска службы SSH была предпринята попытка запуска браузера Firefox через SSH с параметром перенаправления X11:

```
ssh -YC smahmudov@server.smahmudov.net firefox
```

Однако выполнение завершилось ошибкой из-за отсутствия переменной окружения `DISPLAY`, что свидетельствует об отсутствии графической среды на клиенте.

```
[smahmudov@client.smahmudov.net ~]$ ssh -YC smahmudov@server.smahmudov.net firefox
Warning: No xauth data; using fake authentication data for X11 forwarding.
X11 forwarding request failed on channel 0
Error: no DISPLAY environment variable specified
[smahmudov@client.smahmudov.net ~]$
```

Рис. 3.22: Неудачная попытка запуска графического приложения через SSH

3.8 Внесение изменений в настройки внутреннего окружения виртуальной машины

1. В каталоге `/vagrant/provision/server/` был создан подкаталог `ssh/etc/ssh`, в который был скопирован конфигурационный файл `sshd_config`:

```
mkdir -p /vagrant/provision/server/ssh/etc/ssh
```

```
cp -R /etc/ssh/sshd_config /vagrant/provision/server/ssh/etc/ssh/
```

2. В этом же каталоге был создан исполняемый скрипт `ssh.sh`, содержащий команды для автоматического применения сетевых настроек и перезапуска SSH-службы.

```
1  #!/bin/bash
2  echo "Provisioning script $0"
3  echo "Copy configuration files"
4  cp -R /vagrant/provision/server/ssh/etc/* /etc
5  restorecon -vR /etc
6  echo "Configure firewall"
7  firewall-cmd --add-port=2022/tcp
8  firewall-cmd --add-port=2022/tcp --permanent
9  echo "Tuning SELinux"
10 semanage port -a -t ssh_port_t -p tcp 2022
11 echo "Restart sshd service"
12 systemctl restart sshd
```

Рис. 3.23: Скрипт автоматической настройки и перезапуска SSH

4 Вывод

В ходе лабораторной работы были изучены методы конфигурирования безопасного удалённого доступа по протоколу SSH. Выполнены операции по запрету входа под пользователем root, ограничению списка пользователей, добавлению дополнительных портов для подключения и настройке SELinux и брандмауэра. Реализована аутентификация по ключам, настройка SSH-туннелей, перенаправление портов и запуск консольных приложений на удалённом сервере. Также была продемонстрирована автоматизация конфигурации SSH посредством сценария bash.

5 Контрольные вопросы

1. **Вы хотите запретить удалённый доступ по SSH на сервер пользователю root и разрешить доступ пользователю alice. Как это сделать?**

Для запрета удалённого входа под пользователем root и разрешения входа пользователю alice необходимо отредактировать файл конфигурации `/etc/ssh/sshd_config`, добавив строки:

```
PermitRootLogin no
```

```
AllowUsers alice
```

После внесения изменений нужно перезапустить службу SSH командой `systemctl restart sshd`.

2. **Как настроить удалённый доступ по SSH через несколько портов? Для чего это может потребоваться?**

Чтобы SSH-сервер принимал подключения через несколько портов, в файле `/etc/ssh/sshd_config` добавляют строки:

```
Port 22
```

```
Port 2022
```

Затем необходимо настроить SELinux и брандмауэр для разрешения нового порта.

Это может потребоваться для повышения отказоустойчивости, тестирования, а также при необходимости использовать альтернативный порт для обхода фильтров или ограничений провайдера.

3. **Какие параметры используются для создания туннеля SSH, когда команда `ssh` устанавливает фоновое соединение и не ожидает какой-либо**

конкретной команды?

Для создания туннеля SSH в фоновом режиме используются параметры:

`-fNL [локальный_порт]:[удалённый_хост]:[удалённый_порт] [пользователь]@[сервер]`.

`-f` переводит процесс SSH в фоновый режим,

`-N` указывает не выполнять удалённые команды,

`-L` задаёт локальную переадресацию портов.

4. Как настроить локальную переадресацию с локального порта 5555 на порт 80 сервера server2.example.com?

Для настройки локальной переадресации необходимо выполнить команду:

`ssh -L 5555:server2.example.com:80 [пользователь]@[сервер]`.

После выполнения команда создаёт туннель, через который локальный порт 5555 перенаправляется на порт 80 удалённого сервера server2.example.com.

5. Как настроить SELinux, чтобы позволить SSH связываться с портом 2022?

Чтобы разрешить SSH-серверу использовать порт 2022 при активном SELinux, нужно добавить его в список разрешённых портов командой:

`semanage port -a -t ssh_port_t -p tcp 2022`.

Проверить результат можно с помощью `semanage port -l | grep ssh`.

6. Как настроить межсетевой экран на сервере, чтобы разрешить входящие подключения по SSH через порт 2022?

Для разрешения подключения через порт 2022 необходимо добавить правило в `firewalld`:

`firewall-cmd --add-port=2022/tcp`

`firewall-cmd --add-port=2022/tcp --permanent`

После этого нужно перезагрузить службу брандмауэра командой `systemctl restart firewalld`.

Теперь SSH-сервер сможет принимать подключения как на стандартном порту 22, так и на дополнительном 2022.

6 Список литературы

1. Как пользоваться SSH - <https://losst.pro/kak-polzovatsya-ssh>