SSH-сервер OpenSSH

Докладчик: Ширкунов Артем Викторович

Назначение технологии

SSH - (Secure Shell — «безопасная оболочка») - это протокол удаленного управления компьютером с операционной системой Linux. В основном ssh используется для удаленного управления серверами через терминал.

- SSH допускает выбор различных алгоритмов шифрования.
- SSH-клиенты и SSH-серверы доступны для большинства сетевых операционных систем.
- SSH позволяет безопасно передавать в незащищённой среде практически любой другой сетевой протокол.
- Также SSH может использовать сжатие передаваемых данных для последующего их шифрования.

Установка SSH-сервера в Debian GNU/Linux

Если во время установки Debian вы выбрали установку SSH сервера, то в готовой системе он будет уже установлен и запущен. Для проверки его состояния нужно выполнить:

```
sudo systemctl status sshd
```

Иначе нужно установить ssh-server

```
sudo apt install openssh-server ssh
```

Затем для запуска использовать команду:

```
sudo systemctl start sshd
```

Установка ssh в Debian завершена. Теперь вы можете попытаться подключится к SSH серверу локально:

```
ssh root@localhost
```

Остановка ssh-сервера

- Закрытие сеанса оболочки с помощью **exit** или **Ctrl - d.**
- Когда соединение с удаленным сервером разорвано, команды и сочетания клавиш **Ctrl+C**, **Ctrl+Z**, **Ctrl+D** не работают, поскольку клиент пытается отправить эти команды на сервер. В этом случае можно использовать Escape последовательности. Чтобы активировать их поддержку, в файл **/etc/ssh/ssh_config** нужно добавить строку :

  ~.

  Другие управляющие символы можно узнать нажав:

  ~?

Настройка SSH-сервера в Debian GNU/Linux.

Все настройки сервера SSH находятся в файле **/etc/ssh/sshd_config**. Перед тем, как его редактировать, обычно рекомендуется сделать резервную копию.

```
sudo cp /etc/ssh/sshd_config{,_back}
```

1. СМЕНА ПОРТА

Откройте конфигурационный файл и найдите строчку Port, расскоментируйте её, если нужно. Затем пропишите нужное значение порта.

Для того, чтобы изменения вступили в силу надо перезагрузить SSH сервер

```
sudo systemctl restart sshd
```

Теперь, чтобы подключится к этому серверу надо будет явно указать порт с помощью опции **-p**.

2. ОТКЛЮЧЕНИЕ ВХОДА СУПЕРПОЛЬЗОВАТЕЛЯ

Авторизацию для суперпользователя можно отключить. Перед тем, как это делать нужно убедиться, что в системе есть ещё как минимум один пользователь от имени которого можно будет авторизоваться. Затем нужно найти строку **PermitRootLogin** и заменить её значение на **no.**

Чтобы разрешить подключение ssh Debian для пользователя root, нужно заменить значение этого параметра на **yes**.

3. НАСТРОЙКА ПОДКЛЮЧЕНИЙ

Конфигурацию можно настроить таким образом, чтобы разрешить (или запретить) подключение для определённых пользователей и групп, а также ограничить количество максимальных подключений.

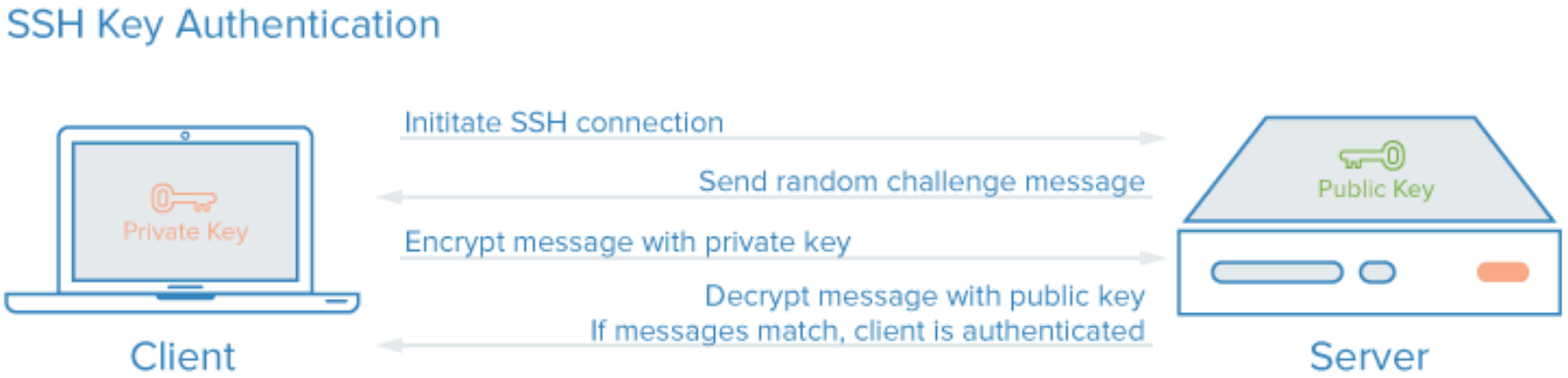
```
AllowUsers student

AllowGroups wheel ssh

DenyUsers otheruser
```

Авторизация по ключу ssh

Ключ состоит из открытой и закрытой части. Секретный ключ сохраняется на стороне клиента и не должен быть доступен кому-либо еще. Открытый ключ используется для шифрования сообщений, которые можно расшифровать только закрытым ключом. Это свойство и используется для аутентификации с помощью пары ключей.



Генерация ключа

Чтобы сгенерировать ключи ssh для аутентификации на локальном сервере. необходимо использовать команду

```
ssh-keygen
```

По умолчанию ключи располагаются в папке ~/.ssh/. Лучше ничего не менять, чтобы все работало по умолчанию и ключи автоматически подхватывались. Секретный ключ будет называться id_rsa, а публичный id_rsa.pub.

Подключение через ssh-copy-id

Когда генерация ключей завершена, необходимо загрузить ключ на сервер. Самый простой способ скопировать ключ на удаленный сервер - это использовать утилиту ssh-copy-id. Она тоже входит в пакет программ OpenSSH. Но для работы этого метода нужно иметь пароль доступа к серверу по SSH. Синтаксис команды:

```
ssh-copy-id username@remote_host
```

Подключение вручную

Чтобы скопировать ключ по ssh вручную, нужно создать каталог ~/.ssh, а затем добавить сгенерированный ключ в файл authorized_keys без перезаписи существующих ключей:

```
cat ~/.ssh/id_rsa.pub | ssh username@remote_host "mkdir -p ~/.ssh && cat >> ~/.ssh/authorized_keys"
```

Отключение проверки пароля

В /etc/ssh/sshd_configначала заменить значение **PubkeyAuthentication** на **yes.**

Перед тем как отключать возможность входа по паролю, нужно убедиться, что существует ключ для авторизации. Затем необходимо заменить значения параметров **ChallengeResponseAuthentication**, **PasswordAuthentication**, **UsePAM** на **no.**

Выполнение одной команды на удалённом сервере

Утилита ssh позволяет сразу выполнить нужную команду без открытия терминала удаленной машины. Например:

```
ssh user@host ls
```

Выполнит команду ls на удаленном сервере и вернет ее вывод в текущий терминал.

Выполнение локального скрипта на удалённом сервере

Выполнение скрипта можно осуществить вызвав интерпретатор bash на удаленном сервере и передав ему локальный скрипт с помощью перенаправления ввода Bash:

```
ssh user@host 'bash -s' < script.sh
```

Передача файлов по SSH

Кроме выполнения команд, можно копировать файлы по ssh. Для этого используется утилита scp. Необходимо указать файл, который нужно передать, удаленный сервер и папку на сервере:

```
scp /адрес/локального/файла пользователь@хост:адрес/папки
```

Настройка клиента

Для каждого пользователя можно сделать отдельные настройки клиента SSH. Для этого используется файл **.ssh/config**. Синтаксис файла такой:

```
Хост имя_хоста
    Параметр значение
```

С помощью * можно задать настройки для всех хостов, например, чтобы задать пользователя при подключении.

```
Host *
    User root
```

Настройка псевдонимов для подключения

```
Host home
    Hostname myhome.dyndns.org
```

Практические задания

1. Проверьте статус ssh-сервера через systemcl. Настройте конфигурационный файл, чтобы подключаться к серверу, используя псевдоним ‘study_server’. Локально создайте текстовый файл с любым содержимым. Не переходя в терминал удалённого сервера, создайте на нём папку и переместите туда только что созданный файл.
2. Сгенерируйте ключ и настройте подключение к ssh-серверу по ключу. Сам ключ защитить паролем ‘keypass’. Конфигурацию сервера настроить таким образом, чтобы ограничить количество подключений - 12. Запретить подключение для группы ‘strangers’. Разрешить для пользователей ‘student’ и ‘teacher’

Список ссылок на полезные источники

- Официальный сайт
- Русскоязычные ресурсы:
  - Установка ssh-сервера
  - Настройка сервера ssh на Debian
  - Авторизация по ключу
  - Памятка на habr
- Англоязычные ресурсы:
  - Статья на archlinux