



ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ
«КОЛЛЕДЖ СВЯЗИ №54»
ИМЕНИ П.М. ВОСТРУХИНА

полное название образовательного учреждения

ДОПУСКАЮ К ЗАЩИТЕ

Заместитель директора по УПР *О.В.Корешков*

(дата)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Настройка сервера удаленного доступа по протоколу ssh

(тема)

Выпускная квалификационная работа должна быть выполнена в виде:
дипломной работы и демонстрационного экзамена

студентом группы

ЗКС11-3

(номер группы)

Киреевым Артёмом Александровичем

(И.О.Фамилия)

(подпись, дата)

Основная профессиональная образовательная программа по специальности
09.02.02 Компьютерные сети

(шифр и наименование специальности)

Форма обучения очная

Руководитель преподаватель

Борис Нилович Дружинин

Руководитель мастер производственного обучения Борис Нилович
Дружинин

(ученая степень, должность, И.О.Фамилия)

(подпись, дата)

Председатель предметной (междисциплинарной, модульной) комиссии
Сергей Николаевич Хохлов

(И.О.Фамилия)

(подпись, дата)

Москва
2020

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
ГЛАВА 1. История развития технологий SSH	7
1.1. Необходимость технологии ssh	7
1.2 Обзор существующих SSH технологий на серверах Linux и роутерах Cisco	16
1.3. Вопросы развития технологий SSH	17
Выводы	18
ГЛАВА 2. Практическая часть	20
2.1. Развёртывание технологии SSH на предприятии	20
2.1.1. Установка OpenSSH на Windows server и Windows 10	20
2.1.2. Установка SSH в UBUNTU	23
2.1.3. Установка и настройка ssh на Debian	27
2.1.4. Настройка сервера ssh на роутерах Cisco	31
2.2. Экономическая часть	35
Выводы	38
Заключение	39
Список литературы:	40

					09.02.02 -3КС11-3					
Изм.	Лист	№ докум.	Подпись	Дата						
Разраб.		Киреев А.А.			«Настройка сервера удаленного доступа по протоколу ssh»			Лит.	Лист	Листов
Провер.		Дружинин Б.Н.								
Рецензент								ГБПОУ КС №54 им. П.М. Вострухина		
Н. Контр.										
Утверд.										

ВВЕДЕНИЕ

Протокол SSH используется для организации безопасного входа в удаленную систему (login) и организации иных безопасных служб через сети, не обеспечивающие безопасности. Протокол включает три основных компоненты:

- ◆ Протокол транспортного уровня.
- ◆ Протокол аутентификации пользователей.
- ◆ Протокол соединений. [1]

Актуальность работы: в настоящее время безопасность данных является критической задачей во многих отраслях, а потому заинтересованность информационного сообщества в более стойких к взлому, а также более быстрых в своей работе алгоритмах может и будет проявляться как сейчас, так и в дальнейшем. [2]

Цель работы: обзор основной работы сервера удаленного доступа по протоколу ssh и его настройка.

Объект исследования: настройка сервера удаленного доступа по протоколу ssh.

Предмет исследования: протокол ssh.

Методы работы: сбор и обработка данных по теме из сети Internet, эксперимент по настройке удаленного доступа по протоколу ssh.

Задачи исследования: опираясь на поставленные цели, включают в себя:

- изучение протоколов шифрования;
- изучение современных потребностей на рынке серверов и соответствия им удаленного доступа по протоколу ssh;
- изучение вопроса о наиболее пригодном использовании указанного сервера удаленного доступа;
- проанализировать количество необходимых затрат для внедрения в эксплуатацию и последующего обслуживания на предприятии сервера удаленного доступа по протоколу ssh;

- анализ плюсов и минусов иных серверов относительно протокола ssh, учитывая его свойства и характеристики;

В основной части данной работы проведено исследование технологии ssh, обзор протокола на серверах Linux и роутерах Cisco, развитие технологии ssh, а также проведена экономическая оценка. Рассмотрим настройку сервера удаленного доступа по протоколу ssh на предприятии. В заключении приведены основные результаты проделанной работы.

					09.02.02 -3КС11-3	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дата		

ГЛАВА 1. История развития технологий SSH

1.1. Необходимость технологии ssh

SSH-протокол — это протокол защищенного удаленного доступа. В основном ssh используется для доступа к роутерам и свитчам, менее к серверам. Если вы администратор нескольких серверов или даже продвинутый веб-мастер, то, наверное, вы часто сталкиваетесь с необходимостью работать с тем или иным компьютером по ssh. В Linux для этого используется сервер ssh на машине, к которой нужно подключиться и клиент, на той из которой подключаются.

SSH обеспечивает защищённый канал связи между клиентом и сервером, через который можно передавать данные (почтовые, видео, файлы), работать в командной строке, удаленно запускать программы, в том числе графические. SSH-сервер должен быть установлен на удаленной операционной системе. SSH-клиент должен быть запущен на машине, с которой будет осуществляться удаленное подключение.

Основные функции, доступные при использовании SSH-протокола:

- Передача любых данных через защищенное SSH-соединение, включая сжатие данных для последующего шифрования.
- X11 Forwarding — механизм, позволяющий запускать программы UNIX/Linux-сервера в виде графической оболочки, как в Windows (использовать X Window System).
- Переадресация портов — передача шифрованного трафика между портами разных машин.

Безопасность SSH-соединения обеспечивается:

- шифрованием данных одним из существующих алгоритмов
- аутентификацией сервера и клиента одним из нескольких доступных методов
- наличием дополнительных функций протокола, направленных на предупреждение различных хакерских атак

Аутентификация сервера дает защиту от:

- взлома путем подмены IP-адресов (IP-spoofing), когда один удаленный хост посылает пакеты от имени другого удаленного хоста
- подмены DNS-записей (DNS-spoofing), когда подменяется запись на DNS-сервере, в результате чего соединение устанавливается с хостом, который указан в подмененной записи, вместо необходимого
- перехвата открытых паролей и других данных, передаваемых в открытом виде через установленное соединение

SSH был разработан как замена для Telnet и для незащищенных протоколов удаленной оболочки, таких как Berkeley rsh и связанных с ними протоколов rlogin и rhexes. Эти протоколы отправляют информацию, в частности пароли, в виде открытого текста, что делает их уязвимыми для перехвата и раскрытия с помощью пакетного анализа. Шифрование, используемое SSH, предназначено для обеспечения конфиденциальности и целостности данных по незащищенной сети, такой как Интернет, хотя файлы, утекшие Эдвардом Сноуденом, указывают на то, что Агентство национальной безопасности иногда можно расшифровать SSH, что позволяет им читать содержимое SSH-сессий.

Коммерческая реализация SSH-протокола — SSH Communications Security — разработана одноименной организацией. Имеет небольшие отличия от бесплатной версии, такие как доступность коммерческой технической поддержки, наличие инструментов веб-управления и др. Основной набор команд и возможностей практически одинаковый у обоих продуктов.

Для ОС Windows выпущены различные SSH-клиенты и оболочки, самые распространенные из них — это бесплатные PuTTY и WinSCP. Для других операционных систем также существуют свои SSH-клиенты. [3]

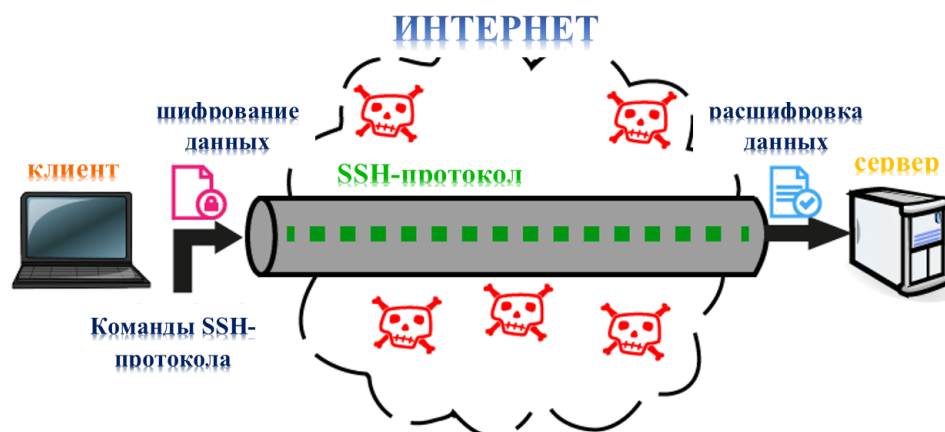


Рисунок 1 Передача данных по SSH-протоколу через небезопасную сеть

Чтобы использовать Secure Shell-доступ достаточно скачать, а затем установить любой SSH-клиент. Специалисты советуют отдать предпочтение популярному и бесплатному клиенту Filezilla. Также существует WinSCP2, так как это одна из наиболее эффективных программ, которая может работать по протоколу SSH2. Программа визуально похожа на достаточно известный FTP-клиент – CuteFTP, она имеет хороший графический интерфейс, а также предоставляет возможность сравнения контента каталогов. Третьей в этом списке идет программа PuTTY, которая тоже имеет своих поклонников, на нашим экспертам понравилась меньше всего.

Для того чтобы осуществить настройку SSH-клиента, достаточно указать ваше доменное имя, IP адрес, пароль и выбрать SSH-протокол. После того, как соединение будет установлено, сервер отправит запрос на введение пароля и имени пользователя. вам следует ввести данные, аналогичные тем, что вы используете для получения FTP-доступа.

Часто бывает такое, что у хостинг-провайдера SSH-доступ является дополнительной платной услугой. Поэтому, если у вас что-то не получается настроить, то обратитесь в поддержку хостера для выяснения причин. [4]

По протоколу SSH работает набор программ, служащих для выполнения различных действий на удаленной операционной системе. Например, программа sshd обеспечивает серверную функциональность SSH, она должна быть запущена на SSH-сервере. Программа ssh запускается на SSH-клиенте и

позволяет устанавливать соединение с удаленным хостом, регистрироваться на нем, работать с удаленной машиной через SSH-соединение.

Для запуска тех или иных программ SSH-протокола существуют специальные команды с набором различных опций. Эти команды могут отличаться в зависимости от используемой клиентской операционной системы и оболочки SSH-клиента. Команды запускаются либо из командной строки, если речь идет о UNIX-подобных системах, либо посредством графического интерфейса в соответствующих SSH-оболочках. [3]

Основной задачей протокола SSH является повышение уровня безопасности в Internet. Протокол пытается сделать это обеспечением максимальной простоты даже за счет некоторого снижения уровня безопасности. [15]

1. Все алгоритмы шифрования, обеспечения целостности и открытых ключей относятся к числу известных и проверенных.

2. Все алгоритмы используются с криптографически обоснованным размером ключей, который позволяет надеяться на обеспечение защиты от самых мощных криптоаналитических атак в течение десятилетий.

3. Все алгоритмы согласуются и в тех случаях, когда тот или иной алгоритм не поддерживается, обеспечивается простой переход к использованию другого алгоритма без изменения базового протокола.

Будут предприниматься специальные меры для того, чтобы упростить широкое и быстрое развертывание протокола. Частным случаем таких мер является возможность работы протокола без проверки ключа хоста при первом подключении, однако такая мера не рекомендуется. Предполагается, что принимаемые меры помогут значительно расширить использование протокола в короткие сроки, пока не будет широко развернута инфраструктура обмена открытыми ключами в сети Internet. [1]

Удаленная работа с графическими приложениями

Перенаправление графического вывода удаленной подсистемы позволяет работать напрямую с графическими приложениями среды Linux на

компьютере с графическим сервером Xming на стороне Windows. Данный режим реализуется с помощью SSH-подключения, в котором ssh-сервер sshd на стороне Linux перехватывает графический ввод-вывод и перенаправляет его ssh-клиенту на стороне Windows, который в свою очередь, перенаправляет его графическому серверу Xming, развернутому в среде Windows. Таким образом, для реализации данного режима не требуется настройка X-сервера и менеджера дисплея для работы по сети, но требуется установка и настройка демона ssh на стороне Linux. В большинстве дистрибутивов Linux для рабочих станций, сервер SSH по умолчанию, не устанавливается, поэтому его нужно установить командой:

```
sudo apt-get install ssh
```

Обычно ставят сервер openssh

```
sudo apt-get install openssh-server
```

Для режима перенаправления графического вывода X11 forwarding в настройках демона sshd необходимо включить некоторые параметры. Все действия требуют права root.

Переходим в каталог /etc/ssh и открываем конфигурационный файл демона SSH sshd_config. Для работы через X11 Forwarding в нем должна присутствовать не закомментированная строка

```
X11Forwarding yes
```

Естественно, в данном режиме, работа с удаленной графической подсистемой Linux, выполняется напрямую с графическими приложениями, без использования рабочего стола Ubuntu. Если количество нужных для работы приложений невелико, то такой способ предпочтительнее, поскольку позволяет снизить степень использования ресурсов удаленной системы и позволяет получить более высокое быстродействие, по сравнению с технологией, основанной на использовании XDMCP. Таким образом, при перенаправлении графического вывода, (X11 forwarding) программа Xming, используется в качестве X-сервера, работающего поверх вашего рабочего стола Windows, с запускаемыми на удаленной системе с ОС Linux

графическими приложениями. При этом, графический сервер на удаленном Linux не используется и может быть даже не установлен.

Кроме Xming, в данной технологии используются клиент и сервер SSH. Клиентская часть - на компьютере с ОС Windows, сервер - на компьютере с Linux. Разработчики Xming с некоторых пор, включили клиентское программное обеспечение для реализации режима X11 Forwarding в состав инсталляционных пакетов (Standart PuTTY и Portable PuTTY). При установке пакета Xming имеется возможность выбрать устанавливаемые версии PuTTY.

Однако, лучшим выбором будет скачать актуальную версию бесплатного SSH - клиента для Windows на странице загрузки PuTTY, где размещены ссылки для скачивания файлов утилиты putty.exe и дополнительных программных модулей, которые могут использоваться для работы с ней. Имеется также ссылка для скачивания архива, включающего putty.exe и дополнительных программ для 32-х и 64-х разрядных ОС. Инсталляция не требуется. Просто копируем исполняемый файл putty.exe в каталог с установленным Xming, или любой другой, по вашему выбору. [13]

Для работы с Xming в режиме перенаправления графического вывода достаточно подправить секцию SSH:

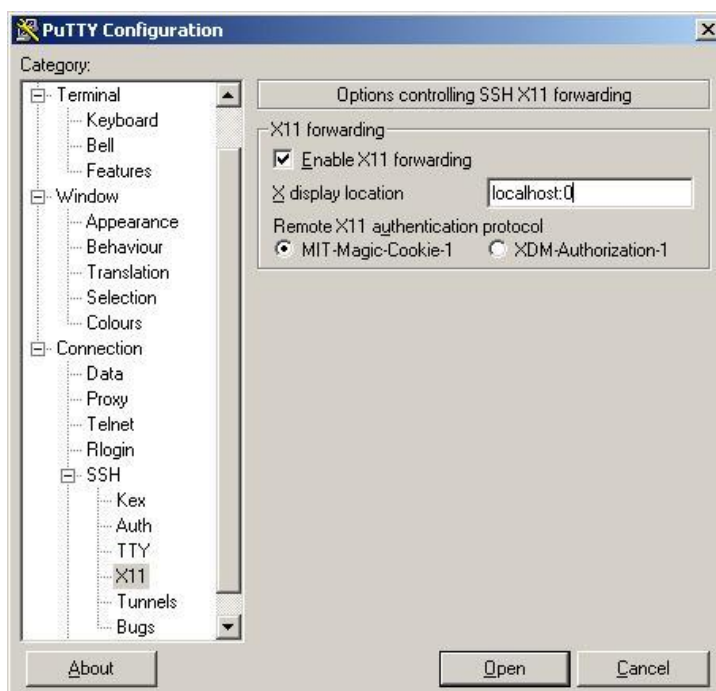


Рисунок 2 Настройка Putty

Данные настройки означают, что разрешено X11 Forwarding и для него будет использоваться графический дисплей (X-дисплей) с номером 0. Использовать 0-й номер дисплея не обязательно, но важно, чтобы этот номер совпадал с номером дисплея, указанном при запуске Xming (поле Display number):

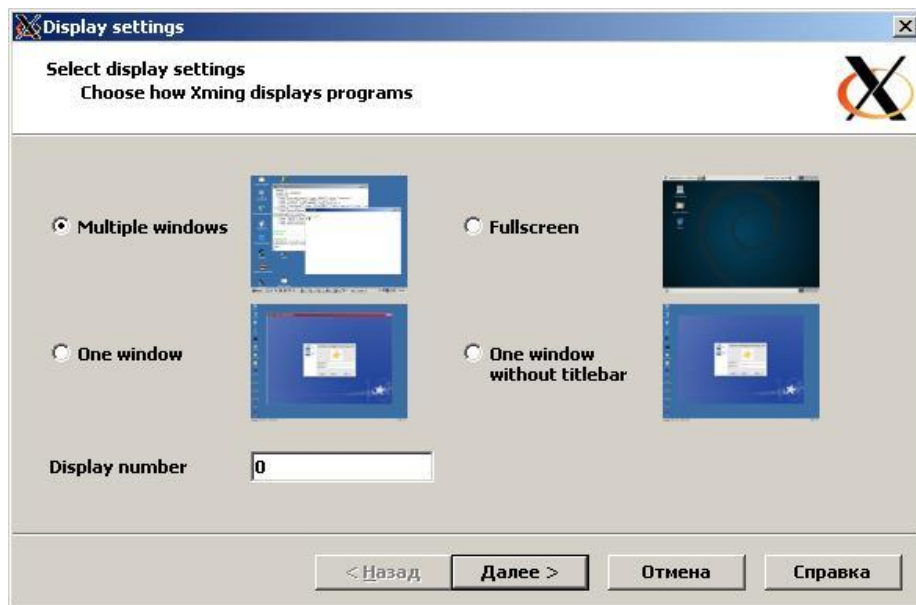


Рисунок 3 Настройки дисплея

При запуске Xming, с помощью мастера Xlaunch, задается номер дисплея (при необходимости) и выбирается многооконный режим Multiple windows, остальные параметры можно оставить по умолчанию. Фактически, номер дисплея определяет номер порта графического сервера на стороне Windows

– 0 соответствует порту 6000/TCP

- 1 – 6001/TCP

...

Как уже рассматривалось выше, настройки X11 Forwarding в секции SSH/X11 PuTTY, также определяют X-сервер, на который будет перенаправляться графический ввод-вывод:

localhost:0 - X-сервер, слушающий порт 6000/TCP

localhost:1 - X-сервер, слушающий порт 6001/TCP

...

Соответственно, если на одном и том же компьютере запускается несколько X-серверов Xming, то номера графических дисплеев для них должны быть разными и соответствовать номерам, задаваемым в настройках клиента SSH. Для проверки подключенных графических подсистем можно воспользоваться командой отправки сообщения графическому дисплею `xmessage $DISPLAY` - отобразить значение переменной `DISPLAY`

В результате выполнения команды получим:

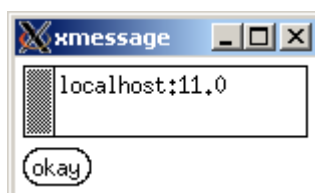


Рисунок 4 Значение переменной

Данное сообщение отображается на том графическом дисплее, которому соответствует перенаправление графического вывода SSH-клиента, в окне которого выполнялась команда `xmessage`.

После того, как Xming стартовал, с помощью ssh-клиента PuTTY подключаемся к ssh-серверу Linux Ubuntu, и в командной строке запускаем нужное графическое приложение, например, если запустить графический терминал `xterm`, то на компьютере с Windows появляется окно графического терминала Linux.

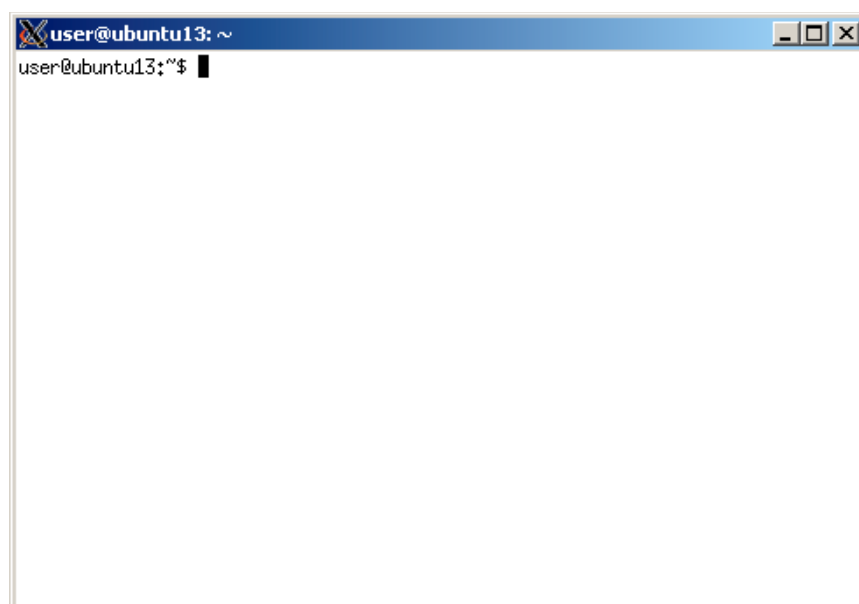


Рисунок 5 Использование граф. терм.

При запуске из сессии SSH-клиента PuTTY, или из окна уже запущенного терминала xterm, какого-либо графического приложения, например, обозревателя firefox на рабочем столе Windows отобразится его окно, в котором можно работать точно так же, как и на локальном компьютере с Linux Ubuntu.

Удаленное подключение к графической подсистеме из Linux.

Настройки демона sshd для удаленного доступа к графической подсистеме Linux выполняются точно так же, как и в случае перенаправления для X-сервера на стороне Windows. Графический вывод удаленной системы, в данной конфигурации, будет перенаправляться демоном sshd и разворачиваться графическим сервером на стороне подключившегося клиента. [14]

Для удаленного подключения к графической подсистеме с использованием перенаправления X11 Forwarding клиентов Linux-систем, можно воспользоваться стандартным SSH-клиентом:

```
ssh -X -l user 192.168.0.1
```

-X - использовать перенаправление графического вывода X11Forwarding.

-l user - имя пользователя для подключения к удаленному компьютеру.

192.168.0.1 - адрес удаленного компьютера

После регистрации в удаленной системе, пользователь **user** имеет возможность работать с графическими приложениями на удаленном компьютере 192.168.0.1

Для запуска конкретного приложения на удаленной системе, можно указать его имя:

```
ssh -X -l user 192.168.0.1 'xterm'
```

После авторизации, на локальном дисплее отобразится окно терминала xterm удаленного компьютера под управлением Linux.

Вместо параметра ssh -X желательно использовать параметр -Y, предотвращающий возможность взаимодействия удаленного клиента с

локальным графическим дисплеем системы, к которой выполняется подключение.

1.2 Обзор существующих SSH технологий на серверах Linux и роутерах Cisco

- **Ubuntu Server**

Ubuntu является стабильным дистрибутивом. Эта система и ее варианты предоставляют прекрасные возможности для пользователя. Система Ubuntu Server поставляется в двух версиях — LTS и в виде плавающего релиза. Версия LTS Ubuntu Server, как утверждается, имеет пятилетний период поддержки. [5]

- **CentOS**

Операционная система CentOS предоставляет устойчивую рабочую среду. Это вариант операционной системы Red Hat Enterprise Linux (RHEL) с открытым исходным кодом. В связи с этим CentOS обеспечивает работу сервера уровня предприятия. В состав CentOS входит менеджер пакетов RPM.

CentOS хорошо работает и на мейнфреймах. Для пользователей, предпочитающих GUI, в системе доступны KDE и GNOME. Система CentOS может использоваться в качестве непосредственной операционной системы для настольных компьютеров.

- **Debian**

Debian представляет собой одну из лучших доступных операционных систем Linux для серверов. Поскольку система Debian была выпущена в 1993 году, а ее первый стабильный выпуск появился в 1996 году, эта система является невероятно защищенной. Многие дистрибутивы Linux, включая и Ubuntu, основаны на Debian.

Debian часто используется на серверах. В состав системы входит менеджер проектов, инструменты APT, и различные средства внешнего представления, такие как GDebi. Debian характеризуется впечатляющей совместимостью приложений, надежностью и стабильностью. [11]

- **Cisco IOS**

Для доступа к Cisco IOS используется метод удаленного доступа к интерфейсу командной строки (CLI) по сети и предусматривает более высокий уровень безопасности. Обеспечивает более надёжную аутентификацию с использованием пароля и использует шифрование при передаче данных.

1.3. Вопросы развития технологий SSH

SSH может работать с технологией IPv6 которая сегодня уже поддерживается на серверах и телекоммуникационном оборудовании. Для этого служит директива ListenAddress в файле sshd_config, которая указывает на адреса, которые должен слушать sshd. У данной директивы следующий синтаксис:

ListenAddress host

ListenAddress IPv4_addr: port

ListenAddress [IPv6_addr]: port

Отроем файл /etc/ssh/sshd_config:

vi /etc/ssh/sshd_config

Для привязки sshd к любому IPv4 и IPv6 адресу на вашем сервере введите:

ListenAddress 0.0.0.0

ListenAddress::

Для привязки sshd к определенному адресу, например IPv6 2607:f0d0:1002:11::2, введите следующую команду:

ListenAddress [2607:f0d0:1002:11::2]

Для привязки к определенному адресу 2607:f0d0:1002:11::2 и порту 311, введите:

ListenAddress [2607:f0d0:1002:11::2]:311

Сохраните и закройте файл. Если вы изменили порт, обновите конфигурацию iptables или pf. И наконец, перезапустите sshd:

service sshd reload

Разработчики надеются, что протокол будет развиваться, и некоторые организации захотят использовать собственные методы шифрования, аутентификации или обмена ключей. Централизованная регистрация всех расширений затруднена, особенно для обеспечения экспериментальных или классифицированных методов. С другой стороны, отсутствие централизованной регистрации приводит к конфликтам в методах идентификации, затрудняя интероперабельность. [12]

Для идентификации алгоритмов, методов, форматов и протоколов расширения были выбраны текстовые имена определенного формата. Имена DNS используются для создания локального пространства имен, в котором могут быть определены экспериментальные или классифицированные расширения без риска конфликта с другими реализациями.

Одна из целей разработки состоит в том, чтобы, с одной стороны, максимально упростить базовый протокол, а с другой стороны, иметь возможность поддерживать несколько алгоритмов. Тем не менее, все реализации должны поддерживать минимальный набор алгоритмов для обеспечения интероперабельности. [6]

Выводы

Область применения протокола SSH практически неограничена. Исходя из его основной функции - удаленного входа в операционную систему, протокол используют:

- системные администраторы для удаленной настройки компьютеров локальной сети;
- для настройки почтовых служб (повышает безопасность данных);
- для скрытого обмена внутри сети массивными файлами;
- для интернет-игр.

Функции надежного шифрования, сжатия, аутентификации и системы контроля данных, реализуемые протоколом SSH-2, позволяют использовать его для сокрытия и защиты от внешних атак передаваемой информации, сократить количество используемого трафика, снизить нагрузку на

центральный процессор за счет создания многопоточного соединения через порт SSH. [13]

ГЛАВА 2. Практическая часть

2.1. Развёртывание технологии SSH на предприятии

Предприятие имеет несколько филиалов, находящихся на большом расстоянии друг от друга, которые необходимо соединить в единую файловую систему. Для решения данной задачи была разработана инструкция по установке и настройке ssh-сервера и ssh-клиента.

2.1.1. Установка OpenSSH на Windows server и Windows 10

Клиент OpenSSH и сервер OpenSSH являются отдельными устанавливаемыми компонентами в Windows Server 2019 и Windows 10 1809.

Чтобы установить OpenSSH, откройте раздел *Параметры* и последовательно выберите *Приложения* > *Приложения* и *возможности* > *Управление дополнительными компонентами*. [8]

Просмотрите этот список и выясните, установлен ли клиент OpenSSH. Если нет, то выберите пункт *добавить компонент* в верхней части страницы, а затем:

- чтобы установить клиент OpenSSH, найдите элемент *Клиент OpenSSH* и щелкните *установить*;
- чтобы установить сервер OpenSSH, найдите элемент *Сервер OpenSSH* и щелкните *установить*.

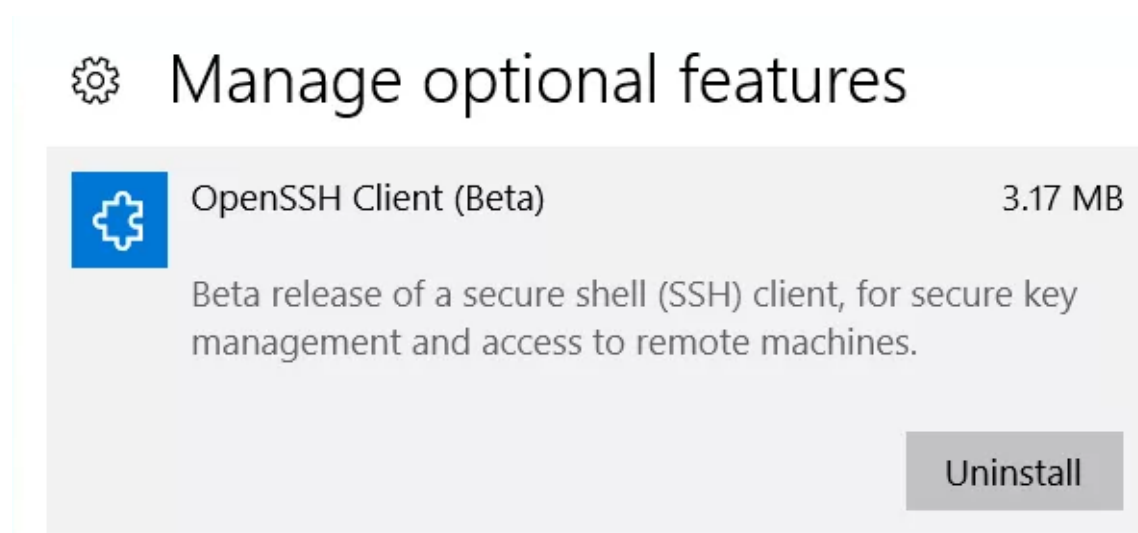


Рисунок 6 Установка ssh

После завершения установки вернитесь в раздел *Приложения> Приложения и возможности> Управление дополнительными компонентами*, где теперь должны появиться компоненты OpenSSH.

Установка сервера OpenSSH создаст и включит правило брандмауэра с именем OpenSSH-Server-in-TCP. Правило разрешает входящий трафик SSH через порт 22.

Установка OpenSSH с помощью PowerShell

Чтобы установить OpenSSH с помощью PowerShell, запустите PowerShell от имени администратора. Убедитесь, что функции OpenSSH доступны для установки, выполнив следующие действия.

```
Get-WindowsCapability -Online | ? Name -like 'OpenSSH*'
```

This should return the following output:

```
Name : OpenSSH.Client~~~~0.0.1.0
```

```
State : NotPresent
```

```
Name : OpenSSH.Server~~~~0.0.1.0
```

```
State : NotPresent
```

Затем установите компонент сервера и (или) клиента.

Install the OpenSSH Client

```
Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

Install the OpenSSH Server

```
Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
```

Both of these should return the following output:

```
Path :
```

```
Online : True
```

```
RestartNeeded : False
```

Удаление OpenSSH

Чтобы удалить OpenSSH через раздел *Параметры* в ОС Windows, откройте этот раздел и последовательно выберите *Приложения> Приложения и возможности> Управление дополнительными компонентами*. В списке

установленных компонентов выберите компонент *Клиент OpenSSH* или *Сервер OpenSSH* и щелкните *Удалить*. [8]

Чтобы удалить OpenSSH с помощью PowerShell, выполните одну из следующих команд:

Uninstall the OpenSSH Client

Remove-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0

Uninstall the OpenSSH Server

Remove-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0

После удаления OpenSSH может потребоваться перезагрузка Windows, если служба использовалась в момент удаления.

Начальная настройка сервера SSH

Чтобы настроить только что установленный сервер OpenSSH для использования в ОС Windows, запустите PowerShell от имени администратора и выполните следующие команды, чтобы запустить службу SSHD:

Start-Service sshd

OPTIONAL but recommended:

Set-Service -Name sshd -StartupType 'Automatic'

Confirm the Firewall rule is configured. It should be created automatically by setup.

*Get-NetFirewallRule -Name *ssh**

There should be a firewall rule named "OpenSSH-Server-In-TCP", which should be enabled

If the firewall does not exist, create one

New-NetFirewallRule -Name sshd -DisplayName 'OpenSSH Server (sshd)' -Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22

Начальное использование SSH

После установки сервера OpenSSH в Windows вы можете быстро проверить его работу с помощью PowerShell на любом устройстве Windows, где установлен клиент SSH. В PowerShell запустите следующую команду:

Ssh username@servername

					09.02.02 -3KC11-3	Лист
						22
Изм.	Лист	№ докум.	Подпись	Дата		

Первое подключение к любому серверу сопровождается сообщением примерно такого содержания:

The authenticity of host 'servername (10.00.00.001)' can't be established.

ECDSA key fingerprint is SHA256:(<a large string>).

Are you sure you want to continue connecting (yes/no)?

В качестве ответа принимаются значения yes (да) или no (нет). Ответ "Да" приведет к добавлению этого сервера в список известных узлов SSH в локальной системе.

После этого появится запрос на ввод пароля. В целях безопасности пароль не будет отображаться по мере ввода.

После успешного подключения вы увидите командную оболочку, которая выглядит примерно так:

domain\username@SERVERNAME C:\Users\username>

2.1.2. Установка SSH в UBUNTU

Установить ssh на Ubuntu будет очень просто, программа считается стандартной и используется почти везде. Для установки откройте терминал и выполните команду:

sudo apt install openssh-server

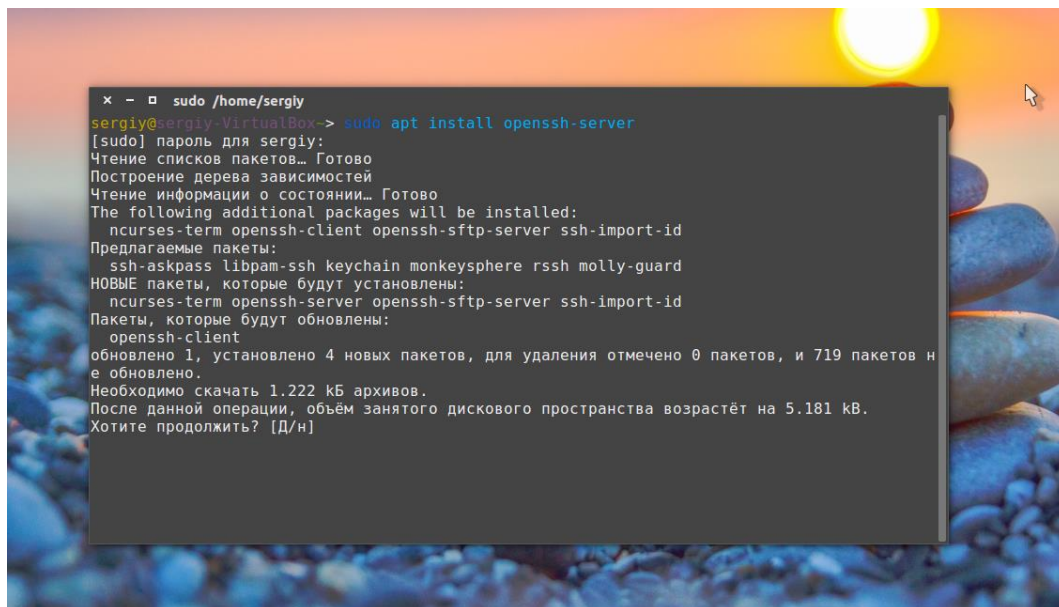


Рисунок 7 Установка openssh

Будет загружено несколько пакетов, а когда установка ssh сервера Ubuntu завершится, программа будет готова к работе. Если вы хотите, чтобы служба запускалась автоматически нужно добавить его в автозагрузку. Поэтому чтобы включить ssh Ubuntu 16.04 выполните:

sudo systemctl enable sshd

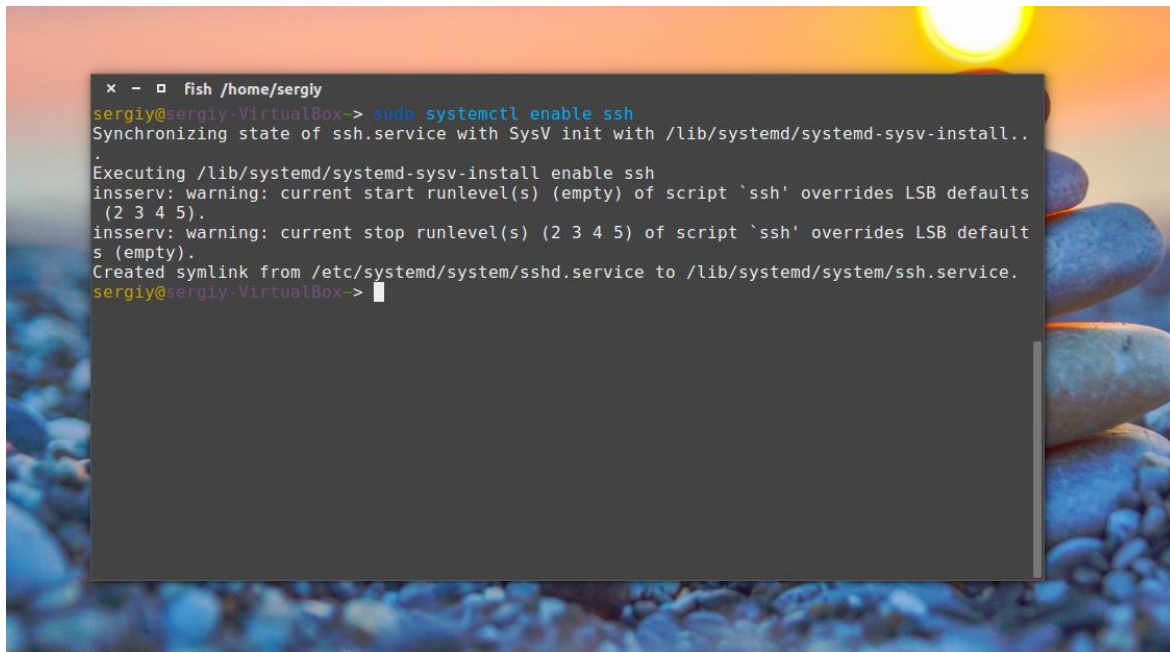


Рисунок 8 Enable sshd

Если затем вы захотите удалить службу из автозагрузки, используйте команду disable:

sudo systemctl disable sshd

Что касается клиента ssh, то он уже установлен в системе по умолчанию. Сейчас вы можете попробовать подключиться к локальному ssh серверу просто набрав:

ssh localhost

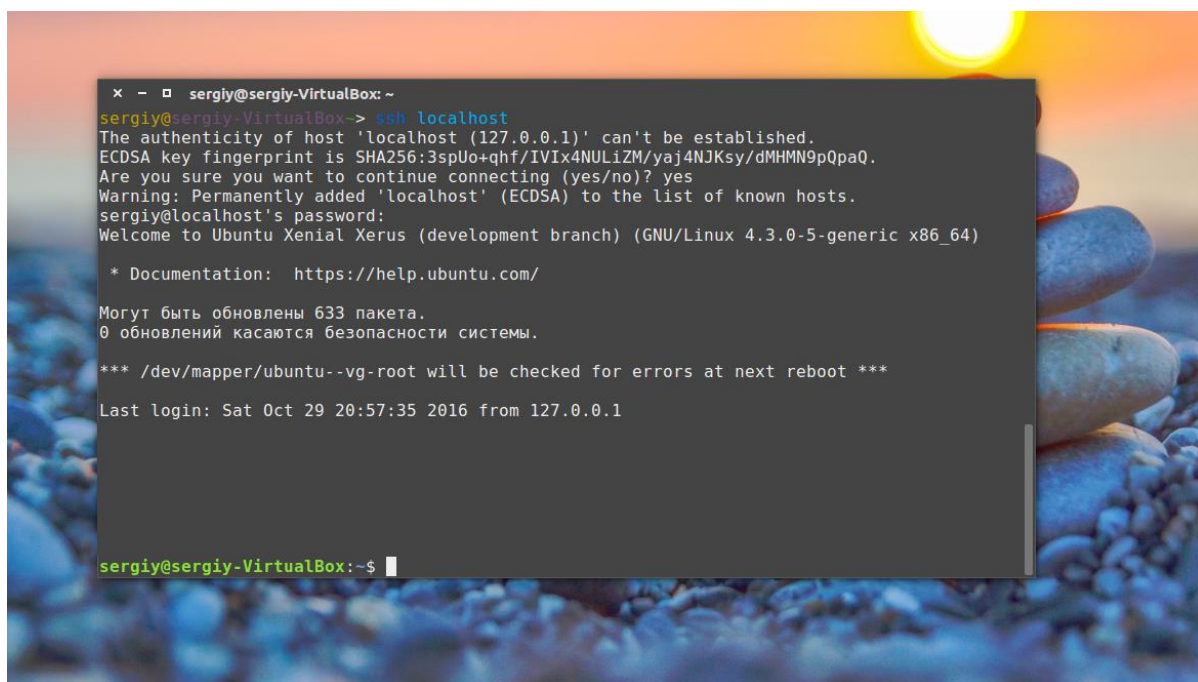


Рисунок 8 Подключение к локальному серверу

Точно таким способом вы можете получить ssh доступ ubuntu к любому другому компьютеру из сети. Для этого достаточно указать вместо localhost его ip адрес и имя пользователя в таком формате:

ssh имя_пользователя@ip_адрес

Настройка SSH в UBUNTU

С параметрами по умолчанию сервер SSH не очень безопасен поэтому перед тем, как программа будет готова к полноценному использованию ее нужно немного настроить. Все настройки сервера SSH хранятся в конфигурационном файле `sshd_config`, который находится в папке `/etc/ssh`. [9]

Перед тем как вносить изменения в этот конфигурационный файл рекомендуется сделать его резервную копию, для этого можете использовать такую команду:

sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.factory-defaults

Дальше вы можете перейти к настройке конфигурационного файла:

sudo vi /etc/ssh/sshd_config

Первым делом желательно сменить порт, на котором работает ssh, возможный злоумышленник не знал включен ли у вас этот сервис. Найдите в

конфигурационном файле строчку Port и замените ее значение на любое число, например, Port 2222:

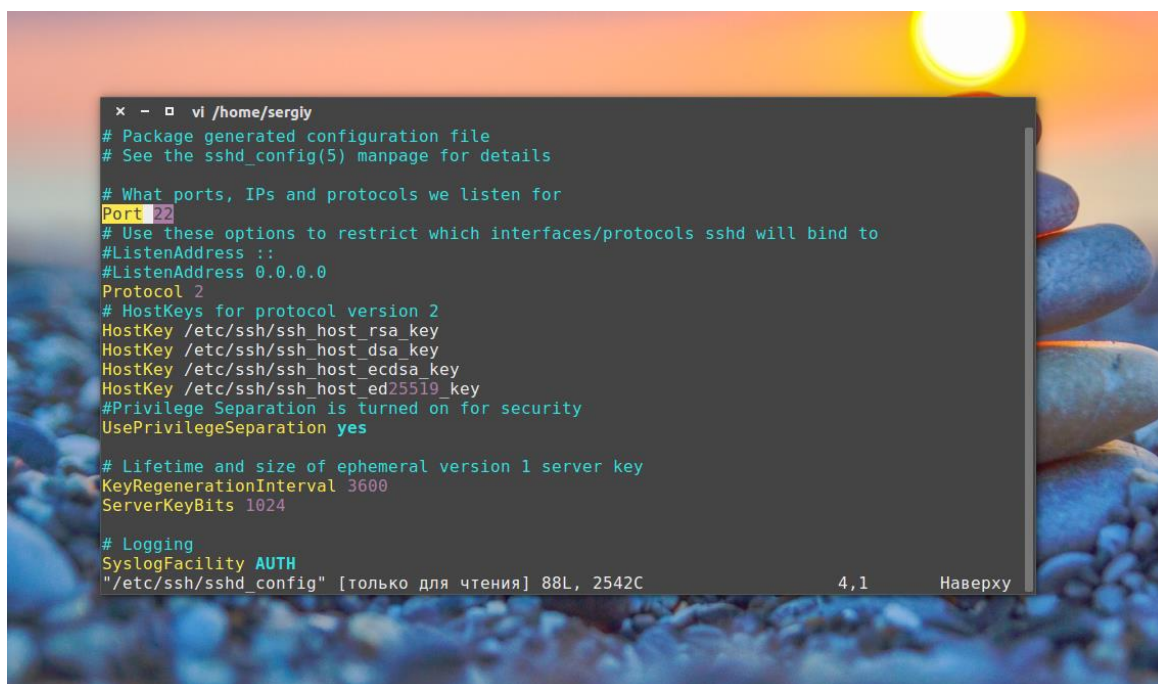


Рисунок 9 Смена значения Port

По умолчанию вход от имени root пользователя включен, рекомендуется отключить такую возможность. Для этого найдите строчку **PermitRootLogin** и замените ее значение на no

Чтобы разрешить аутентификацию по ключу, а не по паролю найдите строчку **PubkeyAuthentication** и убедитесь, что ее значение yes

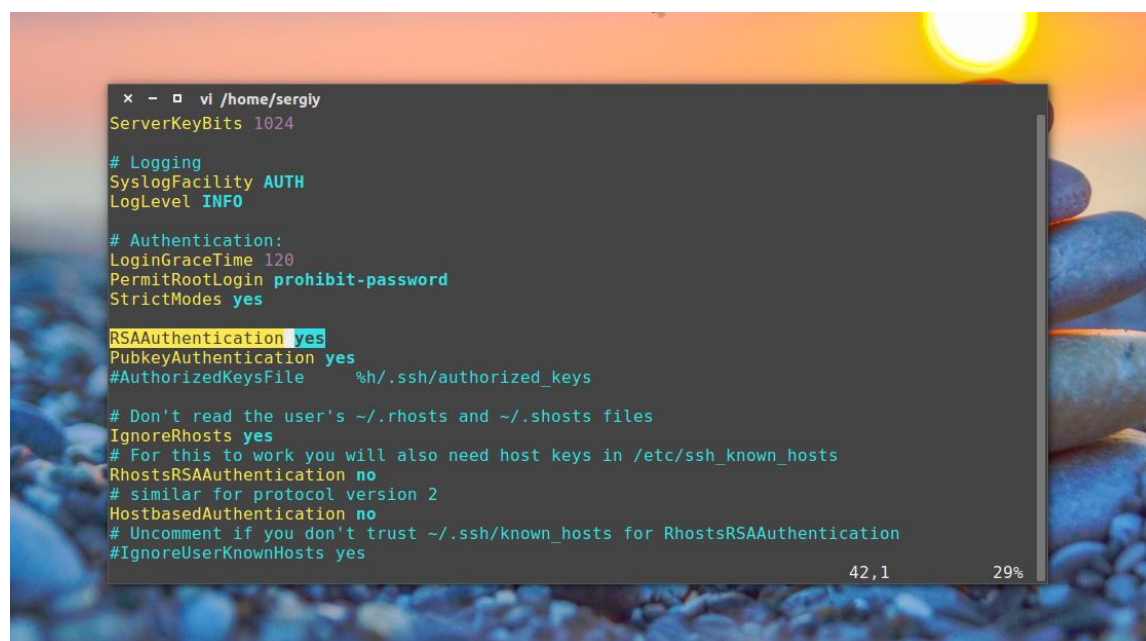


Рисунок 10 Аутентификация по ключу

После того как все настройки будут завершены, сохраните изменения нажав: `w` и перезапустите службу `ssh`:

```
sudo systemctl restart ssh
```

Если вы изменили порт, то при подключении в клиенте тоже нужно указать новый порт, так как по умолчанию будет использоваться 22, например:

```
ssh -p 2222 localhost
```

К тому же, если на компьютере установлен и настроен брандмауэр, то в нем тоже нужно разрешить доступ к новому порту `ssh`, для этого выполните:

```
sudo ufw allow 2222
```

Даже если служба `ssh` находится на порту по умолчанию, его тоже нужно открыть в брандмауэре если вы собираетесь подключаться к компьютеру через интернет:

```
sudo ufw allow 22
```

Теперь, когда установка `ssh` Ubuntu 16.04 завершена, вы можете получить удаленный доступ к своему компьютеру через интернет и быть уверенными что он находится в безопасности. [9]

2.1.3. Установка и настройка `ssh` на Debian

Установка `ssh` сервера

Сначала, обновим списки пакетов с помощью команды:

```
sudo apt-get update
```

Чтобы установить OpenSSH в Debian, выполните следующую команду:

```
sudo apt-get install openssh-server
```

В Debian по умолчанию сервер OpenSSH работает так, что он запускается автоматически после установки. Вы также можете проверить работу с помощью следующей команды:

```
sudo systemctl status ssh
```

Если `ssh`-сервер не работает, вы можете использовать следующую команду, чтобы запустить его. [10]

```
sudo systemctl start ssh
```

Root доступ через ssh

Если вы хотите получить root доступ к серверу, то вы можете попробовать войти так:

```
ssh root@192.168.10.82
```

Но root доступ в большинстве операционных систем отключен по умолчанию. Это также относится и к Debian. Существует обходной путь без изменения конфигурации, просто войдите в систему как обычный пользователь и смените пользователя на root:

```
su -
```

Введите свой пароль от root, и вы должны войти в систему как root

Вы можете изменить конфигурацию вашего SSH-сервера, чтобы разрешить прямой вход в систему как root. [10]

Для этого откройте файл конфигурации «/etc/ssh/sshd_config» с помощью «nano»:

```
sudo nano /etc/ssh/sshd_config
```

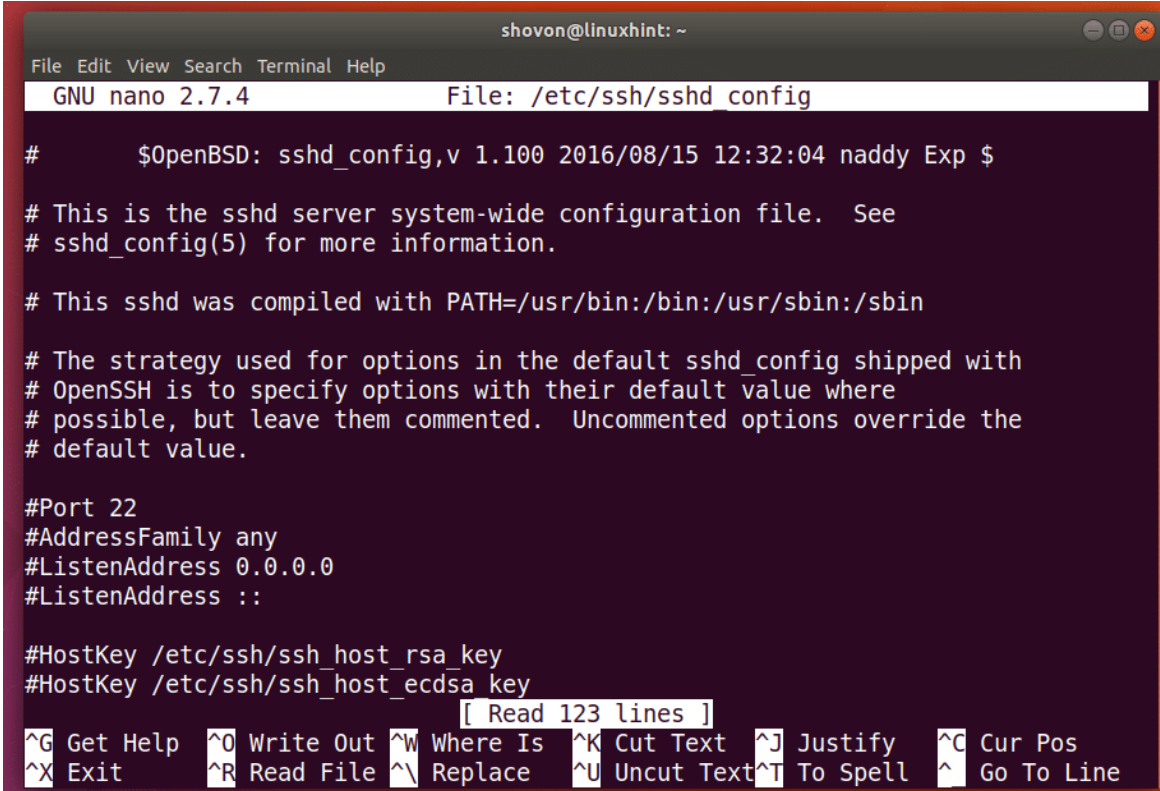


Рисунок 11 Содержимое файла sshd_config

Удалите # перед PermitRootLogin и измените «prohibit-password» на «yes». Как только вы закончите, нажмите Ctrl + X, нажмите «y», а затем нажмите <Enter>, чтобы сохранить файл.

```

shovon@linuxhint: ~
File Edit View Search Terminal Help
GNU nano 2.7.4 File: /etc/ssh/sshd config

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File ^\ Replace  ^U Uncut Text ^T To Spell  ^_ Go To Line
  
```

```

shovon@linuxhint: ~
File Edit View Search Terminal Help
GNU nano 2.7.4 File: /etc/ssh/sshd config Modified

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File ^\ Replace  ^U Uncut Text ^T To Spell  ^_ Go To Line
  
```

Рисунок 12 Изменение Root пользователя

Перезапустим ssh-сервер, чтобы изменения вступили в силу.

```
sudo systemctl restart ssh
```

Теперь вы сможете напрямую подключиться как пользователь root

Подключение по ssh

Перед тем как подключиться к серверу, необходимо знать его ip адрес.

Чтобы узнать ip компьютера, на котором запущен ssh-сервер, выполните следующую команду с этого компьютера. [10]

```
ip a
```

Теперь, подключимся к этому серверу с другого компьютера:

```
ssh username@host/ip_addr
```

Я подключусь к ssh-серверу с адресом 192.168.10.82 как пользователь «user».

```
ssh user@192.168.10.82
```

После вы должны увидеть следующее приглашение, если вы впервые подключаетесь к серверу. Просто введите «yes» и нажмите <Enter>. Затем вам будет предложено ввести пароль от пользователя, под которым вы заходите. Введите пароль и нажмите <Enter>. После подключения имя хоста изменилось с «linuxhint-pc» на «linuxhint». Далее вы можете проверить, что вы подключены к удаленному серверу с помощью следующей команды:

```
ip a
```

Ip адрес должен быть 192.168.10.82

Вы можете запускать здесь любые команды и управлять удаленным сервером по SSH.

Когда вы закончите настройку, просто выполните следующую команду, чтобы закрыть соединение.

```
Exit
```

2.1.4. Настройка сервера ssh на роутерах Cisco



Рисунок 13 Пример топологии

Обеспечим сетевую связность и настроим интерфейс vlan 1 на коммутаторе, для этого введите следующие команды:

En

conf t

interface vlan 1

ip address 192.168.1.1 255.255.255.0

no shutdown

Далее, настроим сетевую карту компьютера – укажем сетевой адрес в настройках FastEthernet0: 192.168.1.2. По умолчанию все новые компьютеры будут находиться в vlan 1. [14]

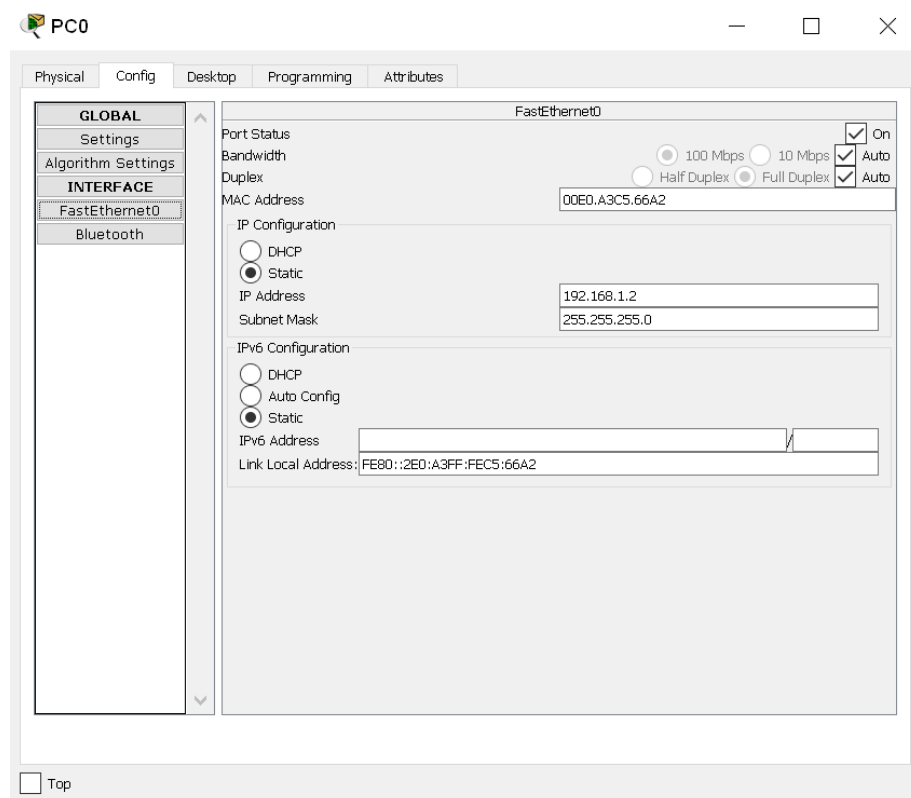


Рисунок 14 Настройка PC0

Теперь попробуем пингануть коммутатор и зайти на него по протоколу telnet с нашего ПК на коммутатор – соединение будет отклонено по причине того, что мы еще не настроили аутентификацию на коммутаторе.

```
C:\>
C:\>
C:\>
C:\>
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=2ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 1ms

C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

[Connection to 192.168.1.1 closed by foreign host]
C:\>
```

Рисунок 15 Пинг коммутатора

Перейдем к настройке аутентификации. Можно усилить общую безопасность с помощью валидации запросов на авторизацию на устройстве. Перейдём обратно в режим общей конфигурации на коммутаторе с помощью команды exit и введите следующие команды:

```

line vty 0 15
password cisco
login local
transport input all
end

```

Пароль cisco, используемый в статье, является крайне небезопасным и служит исключительно для демонстрационных целей. Если вы оставите такой пароль на настоящем оборудовании, шансы, что вас взломают будут стремиться к бесконечности. [14]

Теперь снова попробуем зайти по Telnet на свитч. Однако, при попытке перейти к настройке и выполнению команды enable вы увидите, что это невозможно, по причине того, что не установлен пароль на глобальный режим enable.

Чтобы исправить это, введите следующие команды:

```

conf t
enable password cisco

```

```

Command Prompt
Minimum = 0ms, Maximum = 2ms, Average = 1ms

C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

[Connection to 192.168.1.1 closed by foreign host]
C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

User Access Verification

Password:
merionetSw1>en
% No password set.
merionetSw1>
% Connection timed out; remote host not responding
C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

User Access Verification

Password:
merionetSw1>en
Password:
merionetSw1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
merionetSw1(config)#hostname merionSwitch1
merionSwitch1(config)#

```

Рисунок 16 Подключение по Telnet

Теперь настроим SSH на коммутаторе – для этого обязательно нужно указать хостнейм, доменное имя и сгенерировать ключ шифрования.

Вводим следующие команды:

```
hostname merionet_sw1
```

```
ip domain name merionet
```

```
crypto key generate rsa
```

Выбираем длину ключа — по умолчанию значение стоит равным 512 битам, для SSH версии 2 минимальная длина составляет 768 бит.

После генерации ключа продолжим настройку коммутатора:

```
ip ssh version 2
```

```
line vty 0 15
```

```
transport input ssh
```

Теперь зайти по протоколу Telnet уже не выйдет, так как мы заменили его на SSH. Попробуем зайти по ssh, используя логин по умолчанию — admin:

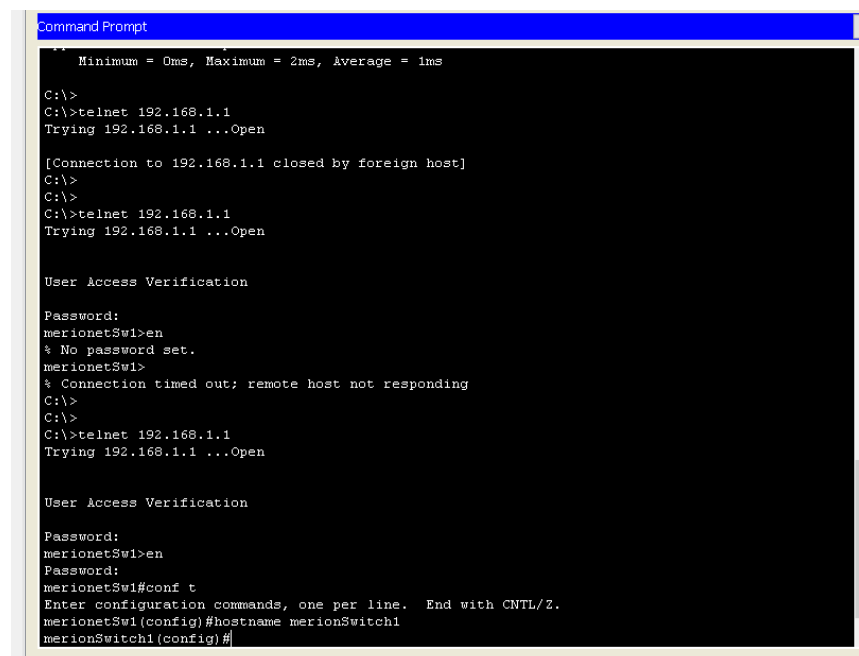
```
username admin secret cisco
```

```
line vty 0 15
```

```
login local
```

```
do wr
```

Заходим с рабочей станции на коммутатор и проверяем настройки.



```
Command Prompt
Minimum = 0ms, Maximum = 2ms, Average = 1ms
C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

[Connection to 192.168.1.1 closed by foreign host]
C:\>
C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

User Access Verification
Password:
merionetSw1>en
% No password set.
merionetSw1>
% Connection timed out; remote host not responding
C:\>
C:\>
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

User Access Verification
Password:
merionetSw1>en
Password:
merionetSw1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
merionetSw1(config)#hostname merionSwitch1
merionSwitch1(config)#
```

Рисунок 17 Проверка настроек

2.2. Экономическая часть

В данной части выпускной квалификационной работы будут проанализированы экономические аспекты владения сервером.

Для исследования рассчитаем выгоду от покупки виртуального частного сервера (VPS) и от покупки физического сервера.

VPS (virtual private server) - услуга предоставления в аренду так называемого виртуального выделенного сервера. В плане управления операционной системой, по большей части она соответствует физическому выделенному серверу.

В частности, root-доступ, собственные IP-адреса, порты, правила фильтрации и таблицы маршрутизации. Внутри виртуального сервера можно создавать собственные версии системных библиотек или изменять существующие, владелец VPS может удалять, добавлять, изменять любые файлы, включая файлы в корневом и других служебных каталогах, а также устанавливать собственные приложения или настраивать/изменять любое доступное ему прикладное программное обеспечение.

Виртуальный выделенный сервер эмулирует работу отдельного физического сервера. На одной машине может быть запущено множество виртуальных серверов. Помимо некоторых очевидных ограничений, каждый виртуальный сервер предоставляет полный и независимый контроль и управление, как предоставляет его обычный выделенный сервер. Каждый виртуальный сервер имеет свои процессы, ресурсы, конфигурацию и отдельное администрирование. Администратор-владелец виртуального сервера может устанавливать любые приложения, работать с файлами и выполнять любые другие задачи, возможные на отдельной машине. [15]

Виртуальные серверы без поддержки предоставляются по низким (от нескольких долларов в месяц) ценам. Неподдерживаемый хостинг хорошо подходит для специалистов и энтузиастов. Поддерживаемые виртуальные серверы варьируются в широких пределах и подходят тем, кто заинтересован

направить все усилия на развитие сайта, а не на технические детали его содержания.

Покупка услуги хостинга будет рассмотрена на примере сайта «mail.ru cloud solutions». Для заказа данной услуги, необходимо пройти несложную регистрацию. После чего выбрав соответствующий пункт меню, мы попадаем на страницу, на которой нам предлагают рассчитать стоимость, путем выбора подходящей для нас конфигурации будущего сервера. [10]

The screenshot shows a configuration interface for 'cloud servers'. It includes sliders for CPU (2), RAM (8 GB), and Hard disk (40 GB). There are buttons for OS (Windows/Linux) and Disk type (SSD/HDD/hSSD). A summary bar at the bottom shows '2700 Р / мес' and '0,063 Р / мин'.

Рисунок 18 Цена аренды сервера

Также доступна возможность приобрести одну из уже готовых, популярных конфигураций серверов:

STANDART-2-8-50	STANDART-4-8-80	STANDART-4-16-50	ADVANCED-8-16-100	ADVANCED-8-32-50
2 740 Р / месяц 0,064 Р / мин	3 880 Р / месяц 0,09 Р / мин	5 280 Р / месяц 0,123 Р / мин	7 520 Р / месяц 0,175 Р / мин	10 360 Р / месяц 0,24 Р / мин
2 vCPU	4 vCPU	4 vCPU	8 vCPU	8 vCPU
8 GB RAM	8 GB RAM	16 GB RAM	16 GB RAM	32 GB RAM
50 GB	80 GB	50 GB	100 GB	50 GB
Тип диска SSD HDD	Тип диска SSD HDD	Тип диска SSD HDD	Тип диска SSD HDD	Тип диска SSD HDD
Операционная система Windows Linux	Операционная система Windows Linux	Операционная система Windows Linux	Операционная система Windows Linux	Операционная система Windows Linux





Рисунок 19 Готовые VPS


В итоге, в нашем случае, при заказе собственной конфигурации для виртуального сервера, стоимость хостинга составит 2700 рублей в месяц, а итоговая стоимость годового пользования услугой будет 32400 рублей. Для объективного сравнения, покупка физического сервера производится со схожими характеристиками. На сайте «SoftMagazin»[6]:

Сервер HPE ProLiant ML350 Gen10 877619-421

Артикул – 877619-421

В наличии



ЦЕНА СЕГОДНЯ:

121 589.00 руб.

РЕКОМЕНДОВАННАЯ РОЗНИЧНАЯ ЦЕНА:

136 179.00 руб. (-11%)

Доставим за 1-2 дня

Купить

Рисунок 20 Покупка сервера

Данный сервер хорошо подходит для растущих малых и средних предприятий и имеет следующие характеристики [7]

Характеристики	Описание
Разработчик:	Hewlett-Packard
Форм фактор:	Tower
Установленный процессор:	Intel Xeon Bronze 3104
Количество установленных процессоров:	2
Количество сокетов:	4
Тип оперативной памяти:	DDR4
Объем установленной оперативной памяти, Гб:	8
Слоты для оперативной памяти:	24
RAID:	да
Встроенный сетевой интерфейс:	4xGE
Тип жесткого диска:	3.5"
Максимальное количество жестких дисков:	12
Горячая замена жесткого диска:	да
Максимальное количество БП:	1
Тип сокета:	FCLGA3647
Количество установленных БП:	1
Блок питания, Вт:	500

Рисунок 21 Характеристики сервера

Помимо цены за покупку, также необходимо учесть затраты на обслуживание физического сервера внутри предприятия. На себя эти обязанности берет системный администратор. По данным сайта «hh.ru»,

средняя зарплата на данной должности в Москве в 2020 году составляет 50 тыс. рублей. Системный администратор будет уделять примерно 1/4 своего времени на обслуживание сервера. Таким образом, в месяц на обслуживание сервера будет уходить 12500 рублей.

В итоге, покупка указанного сервера и его годовое обслуживание будет стоить: 271589 рублей (цена сервера – 121589 и зарплата системного администратора – 150000 (12500*12)). [12]

Выводы

В услуге VPS клиент сам выбирает себе операционную систему на сервере и затем может поставить на нее все что ему нужно. Любые программные сервера. В том числе ssh. Поэтому хостинг компаниям не нужно поддерживать ssh. При аренде linux VPS клиент всегда сразу получает ssh доступ. Выбирая хостинг обращайтесь внимание на наличие возможности подключиться через SSH.

Провайдер должен обеспечить надежность и быстроту обмена данными с его сервером, особенно если речь идет о больших объемах информации и ее высокой конфиденциальности. Также SSH может использоваться для удаленной работы по защищенному соединению с различными сервисами провайдера, такими как программное обеспечение, операционные системы.

Заключение

Целью данной работы был анализ развития технологий ssh, а также разработка инструкции для настройки этих технологий на предприятии. В первой главе работы рассмотрена история развития технологий ssh и изучены принципы их работы на роутерах cisco и серверах linux, а также изучение протоколов шифрования. Был рассмотрен вопрос о наиболее пригодном использовании сервера удаленного доступа. В конце первого пункта были изучены современные потребности на рынке серверов и соответствия им удаленного доступа по протоколу ssh

Во второй главе рассматривается установка OpenSSH на Windows Server 2019, Windows 10, Ubuntu и Debian. В последнем пункте второй главы имеется информация о экономических сравнениях подключения ssh, рассмотрен вариант покупки сервера, вариант аренды и хостинга. Были проанализированы количество необходимых затрат для внедрения в эксплуатацию и последующего обслуживания на предприятии сервера удаленного доступа по протоколу ssh и приведены цены для сравнения, и сделан общий экономический вывод.

Список литературы:

Электронные ресурсы:

1. Архитектура протокола SSH. [Электронный ресурс]. <http://rfc.com.ru/rfc4251.htm>;
2. Протокол SSH. [Электронный ресурс]. <https://netclo.ru/dostup-po-protokolu-ssh-k-oborudovaniyu-cisco/>;
3. Что такое SSH. [Электронный ресурс]. <https://firstvds.ru/technology/ssh-connection>;
4. Что такое SSH-технология. [Электронный ресурс]. <https://ru.hostings.info/termins/ssh.html>;
5. ОС Linux. [Электронный ресурс]. <https://8d9.ru/12-luchshix-operacionnyx-sistem-linux-dlya-serverov-i-dlya-kogo-oni-prednaznacheny>
6. Возможности дальнейшего развития SSH. [Электронный ресурс]. <https://www.intuit.ru/studies/courses/59/59/lecture/1760>
7. An Overview of the Secure Shell (SSH). [Электронный ресурс]. https://www.vandyke.com/solutions/ssh_overview/ssh_overview.pdf
8. Установка OpenSSH для Windows Server 2019 и Windows 10. [Электронный ресурс]. https://docs.microsoft.com/ru-ru/windows-server/administration/openssh/openssh_install_firstuse
9. Установка ssh Ubuntu 16.04. [Электронный ресурс]. <https://losst.ru/ustanovka-ssh-ubuntu-16-04>
10. Установка ssh и root доступа Debian. [Электронный ресурс]. <https://mordeniuss.ru/install-ssh/>
11. Виртуальная машина. [Электронный ресурс]. <https://mcs.mail.ru/cloud-servers/>

Книжные ресурсы:

12. АКИМОВ С.В. SSH The Secure Shell. - СПб.: СПбГУТ, 2017.;

13. Биячуев Т.А. OpenSSH. Учебное пособие / под ред. Л.Г.Осовецкого - СПб.: СПбГУ ИТМО, 2018г.;
14. Куроуз Д., Росс Т. Implementing SSH. - М.: Эксмо, 2019.;
15. Олифер В., Олифер Н. Step by Step SSH. - СПб.: Издательский Дом ПИТЕР, 2018.;

					09.02.02 -3КС11-3	Лист
Изм.	Лист	№ докум.	Подпись	Дата		41