ПРАКТИЧЕСКАЯ РАБОТА №4

Элементы безопасности в Linux

Цель работы: получить практические навыки работы с сетевой подсистемой в Linux, научится управлять пользователями, правами на файлы и каталоги, научиться настраивать сетевые интерфейсы, NAT и настраивать ssh.

Необходимо:

- ОС Linux Debian на виртуальной машине

Краткие теоретические сведения:

Linux сейчас является основной операционной системой для развертывания сервисов обработки данных. Доступ к Linux платформам осуществляется через сеть.

Для работы сети в общем случае нужно настроить на сетевом адаптере IP адрес, маску, адрес шлюза по умолчанию и IP адрес DNS сервера. Эти параметры можно настраивать вручную или при помощи специальной службы – DHCP. DHCP (Dynamic Host Configuration Protocol — протокол динамической конфигурации узла) — это сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP, запрашивая эти параметры с DHCP сервера.

Узнать текущие настройки можно с помощью команды ip address (или сокращённо ip a).

В Linux Debian для постоянного изменения конфигурации требуется изменить конфигурационные файлы, которые используются сетевой службой для их конфигурирования при запуске. Это файл /etc/network/interfaces для настройки ip, mask, gateway и файл /etc/resolv.conf где хранятся настройки DNS.

Вот пример файла /etc/network/interfaces.

```
auto eth0  # включать интерфейс (сетевой адаптер) при старте ОС
allow-hotplug eth0 # включать интерфейс (сетевой адаптер) если он появится
уже в запущенной ОС
iface eth0 inet dhcp # получать параметры работы с dhcp сервера в сети
auto eth1
allow-hotplug eth1
iface eth1 inet static # настроить интерфейс вручную
address 192.168.1.2/24 # установить на него IP = 192.168.1.2 и маску
255.255.255.0
gateway 192.168.1.1 # установить на него шлюз по умолчанию 192.168.1.1
dns-nameservers 8.8.8.8 # установить адрес DNS сервера 8.8.8.8
```

В примере используются имена интерфейсов eth1 и eth0. Они могут отличаться в разных системах и называться enp0s1 и ens1. Узнать, как называются интерфейсы в вашей системе можно с помощью команды ip link (или сокращенно ip l).

После изменения настроек нужно перезапустить или систему или (что правильнее) службу сети:

systemctl restart networking.service

Для диагностики соединений используется утилита ping.

ОС Linux содержит необходимые средства для организации защищенного удаленного доступа и организации Интернет-шлюза.

NAT (Network Address Translation) – технология стека TCP\IP. Она позволяет модифицировать заголовки пересылаемых через NAT IP-пакетов и TCP\UDP сообщений.

NAT в общем случае представляет собой компьютер или аппаратный маршрутизатор, подключенный одним интерфейсом к внешней сети, а другими к внутренней. Оба интерфейса имеют IP адреса в каждой из сетей. Типичным применением NAT является обеспечение доступа из локальной сети с приватными IP-адресами к ресурсам внешней сети с IP-адресами интернет. При передаче запроса от локального клиента к внешнему ресурсу подменяется сокет отправителя: IP адрес меняется на внешний IP адрес NAT, а порт на свободный порт на внешнем интерфейсе NAT. Когда приходит ответ от внешнего ресурса, происходит обратная замена сокета и пакет передается в локальную сеть получателю. Так же с помощью NAT можно публиковать локальные сокеты на реальном IP адресе и реальном порту. Например, для обеспечения доступа извне к Web серверу, расположенному в локальной сети. В этом случае на NAT делается статическое отображение внешнего сокета на внутренний.

Под межсетевым экраном или брандмауэром понимают фильтр IP пакетов предназначенный для формального ограничения соединений клиентов и серверов работающих «поверх» стека TCP\IP.

В основу работы классического firewall положен контроль формальных признаков. В общем случае фильтрация осуществляется по:

- IP адресам отправителя и получателя в заголовке IP пакета
- номерам портов приложения-получателя и приложения-отправителя
- инкапсулированным в IP протоколам транспортного (TCP, UDP) и сетевого уровней (ICMP).

Правила фильтрации формируются в виде списка. Все проходящие пакеты проверяются по списку последовательно, до первого срабатывания. Последующие правила к пакету не применяются.

Для управления шлюзом используются различные инструменты управления брандмауэром Linux, такие как iptables, nftables и firewalld. Однако, все еще самым распространенным является **iptables**.

Важно отметить, что для того, чтобы Linux начал пересылать пакеты из интерфейса в интерфейс надо чтобы в параметре ядра net.ipv4.conf.all.ip_forward = 1. Установить его можно с помощью утилиты sysctl (файл /etc/sysctl.conf), или записью в конфигурационный файл в каталоге /proc.

В Linux для удаленного доступа к серверам используется протокол SSH (secure shell). Он создает шифрованное соединение между клиентом и сервером. Благодаря этой технологии может осуществляться удаленное управление компьютером.

Сервер ssh (openssh-server) устанавливается по умолчанию и выполняется службой sshd. Конфигурация сервера осуществляется в конфигурационном файле /etc/ssh/sshd_config.

С помощью ssh можно не только подключаться к удаленным хостам, но и получать доступ к другим сервисам и сетям через эти хосты. Например, можно опубликовать на локальном сокете любой удаленный сокет, доступный с ssh хоста, к которому осуществляется подключение.

ssh -L [LOCAL_IP:]LOCAL_PORT:DESTINATION:DESTINATION_PORT [USER@]SSH_SERVER

где:

- [LOCAL_IP:]LOCAL_PORT — IP-адрес и номер порта локального компьютера,
- DESTINATION:DESTINATION_PORT — IP или имя хоста и порт конечного компьютера,
- [USER@]SERVER_IP — удаленный пользователь SSH и IP-адрес сервера.

Среди прочих утилит в комплект ssh входят утилиты для защищенной передачи файлов **scp**.

Сейчас безопасным считается использование ключей, а не паролей для аутентификации. Для генерации ключа используется утилита ssh-keygen. Ключи обычно хранятся в каталоге .ssh в домашнем каталоге пользователя.

Для управления запуском и просмотра состояния сервиса используется системная утилита systemctl. Вот основные приемы ее использования:

systemctl enable ИмяСервиса # разрешение запуска сервиса

systemctl start ИмяСервиса # запуск сервиса

systemctl stop ИмяСервиса # остановка сервиса

systemctl restart ИмяСервиса # перезапуск сервиса при котором (в зависимости от конкретного сервиса) будут оборваны соединения

systemctl reload ИмяСервиса # перезапуск сервиса при котором (в зависимости от конкретного сервиса) не будут оборваны соединения

systemctl status ИмяСервиса # вывод информации о состоянии сервиса.

Для того, чтобы посмотреть «слушает» ли сетевой сервис сокет в ожидании подключений, можно воспользоваться командой ss на локальном хосте, то есть на ОС с сервером, а для того, чтобы проверить это снаружи, например с другого компьютера, можно воспользоваться командой nmap.

В этой работе вы освоите основные приёмы работы с учетными записями пользователей и назначения прав.

Для управления пользователями используются команды:

adduser – для создания пользователя или включения пользователя в группу.

userdel, usermod – для удаления или изменения пользователя,

passwd – для изменения пароля.

Пользователи могут входить в группы. Для управления группами используются утилиты: groupadd, groupdel, groupmod (добавление, удаление и изменение групп).

На файл (или каталог) можно назначать три тройки прав: Read Write Execute. Первая тройка прав rwx для пользователя, который является владельцем файла, вторая тройка rwx - для группы, которая является группой-владельцем файла, третья тройка rwx для всех остальных пользователей.

Для файла:

- r (read) - чтение файла разрешено
- w (write) - запись файла разрешена, то есть можно его редактировать, переименовывать, удалять.
- x (execute) - исполнение файла разрешено.

Для каталога:

- r (read) - разрешено просматривать содержимое каталога, то есть можно воспользоваться командой ls и посмотреть какие файлы и каталоги содержаться в данном каталоге.
- w (write) - используется совместно с атрибутом x (execute). Позволяет удалять и переименовывать файлы в каталоге.
- x (execute) - при использовании совместно атрибутом r (read) позволяет увидеть атрибуты файла, то есть его размер, дату модификации, права доступа. Одним словом, позволяет полноценно воспользоваться командой ls -l. При использовании совместно с атрибутом w (write) позволяет перейти в каталог командой cd, удалять и переименовывать файлы.

Для управления правами служат утилиты chmod для смены прав и chown для смены владельца.

Посмотреть текущие права можно утилитой ls.

Для безопасной работы в Linux рекомендуется использовать утилиту sudo. Она временно повышает привилегии до суперпользователя root или дот заданного в конфигурации пользователя. Обычно для использования sudo достаточно поставить пакет sudo и включить пользователя в группу sudo (в Debian).

Существует файл sudoers для более тонкой настройки sudo. Для его редактирования используется редактор visudo, запускаемый от имени root. Ниже приведен пример, в котором для пользователя User1 дано разрешение запускать от root редактор nano, а для пользователя User2 ограничения по утилитам отсутствует.

```
User1 ALL=(ALL) NOPASSWD: /usr/bin/nano
User2 ALL=(ALL:ALL) ALL
```

Инструментальные средства:

Утилиты:	sysctl systemctl ip useradd ss iptables iptables-save iptables-restore ls adduser passwd chmod chown who ssh-keygen scp sodo visudo
Файлы:	/etc/ssh/sshd_config
Утилиты работы с текстом:	echo, grep, sed
Редакторы:	vi, nano

Порядок выполнения работы:

Часть 1. Подготовка конфигурации

В VirtualBox:

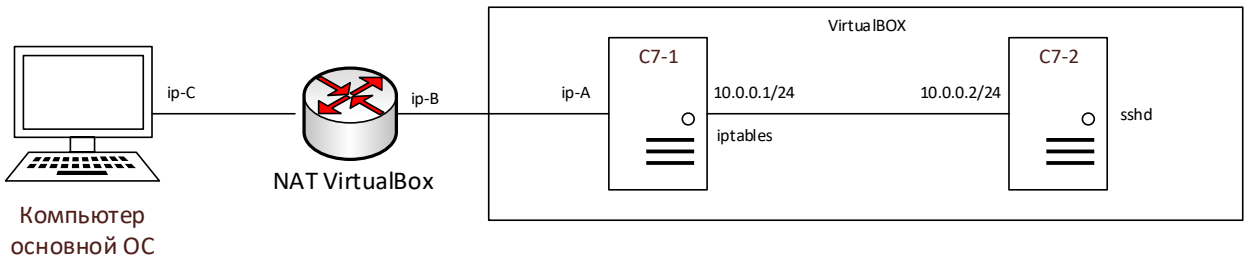
1. Вам понадобится две виртуальных машины. Создайте дополнительную машину клонированием.
2. В первой виртуальной машине Linux Debian добавьте дополнительный сетевой интерфейс. В VirtualBox один сетевой интерфейс настройте в режим «NAT», второй в режим «Внутренняя сеть» с именем intnet.
3. Во второй виртуальной машине с Linux Debian сетевой интерфейс настройте в режим «Внутренняя сеть» с именем intnet.
4. Запустите машины. Назовите первый хост Debian – c7-1, а второй – c7-2

Переименовать компьютер можно с помощью утилиты hostnamectl:

```
hostnamectl set-hostname ИМЯ-ХОСТА
```

5. Для внутренней сети задайте для машин c7-1 и c7-2 адреса 10.0.0.1 и 10.0.0.2 с маской 255.255.255.0. В качестве адреса DNS сервера на c7-2 указать адрес 8.8.8.8 и 77.88.8.1. На c7-2 в качестве шлюза по умолчанию задайте адрес c7-1. Используйте для настройки файл interfaces
6. Для исходного интерфейса c7-1 оставьте получение адреса автоматически от dhcp сервера VirtualBox
7. Проверьте доступность хостов по внутренней сети и доступность внешней сети на хосте c7-1.
8. На машине c7-1 установите параметры ядра так, чтобы ядро передавало сетевые пакеты между сетевыми интерфейсами. Для этого можно отредактировать файл /etc/sysctl.conf, установив параметр net.ipv4.ip_forward=1.

Должна получиться следующая схема:



В первой работе упоминались видео которые могут помочь и в этот раз: основы работы с Virtual Box (https://vk.com/wall-211496571_25) и настройка сети в VirtualBox (https://vk.com/wall-211496571_26).

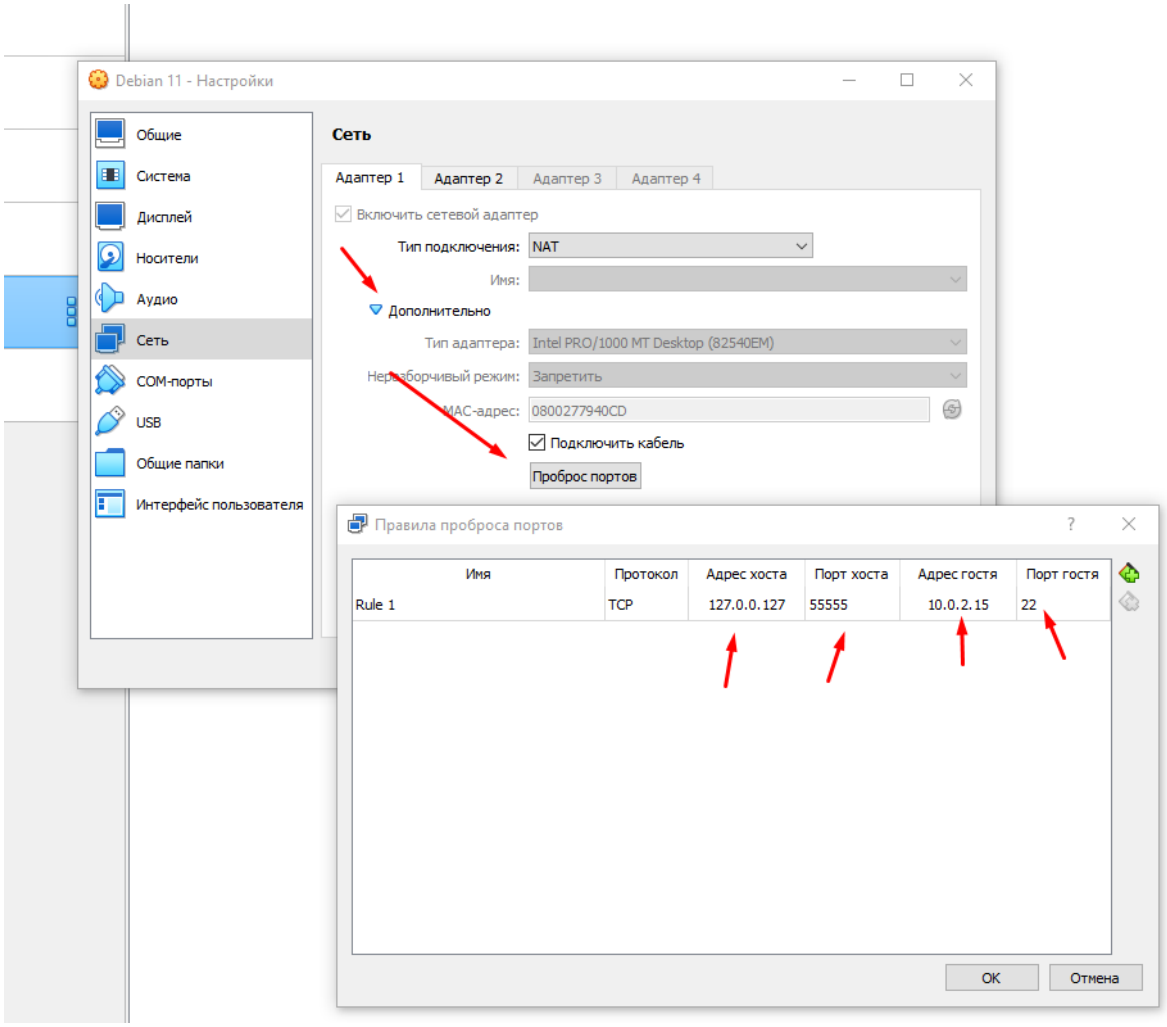
Часть 2. Создание пользователей и настройка OpenSSH Server (sshd).

1. На хостах c7-1 и c7-2 создайте пользователя с именем FIO-c7-N, где FIO – ваши инициалы, а N номер хоста, например adb-c7-1. Сделать это можно с помощью команды adduser.
2. Редактируя файл /etc/ssh/sshd_config, настройте ssh сервер так, чтобы:
    a. Пользователю root можно было бы входить по ssh (PermitRootLogin yes)
    b. Максимальное количество неудачных авторизаций в сессии = 2 (MaxAuthTries 2)
    c. Отключить определение имен хостов по DNS (UseDNS no)

    Часто достаточно убрать символ комментария #.
3. После изменения конфигурации перезапустите сервис sshd. Убедитесь, что после перезапуска сервис работает и готов принимать соединения. Консольный вывод команд сохраните для отчета.
4. Проверьте возможность входа с машины c7-1 на c7-2 по ssh, с использования новой учетной записи.

Часть 3. Подключение к виртуальной машине c7-1 по ssh через NAT VirtualBox

1. В VirtualBox в свойствах сетевого соединения, работающего через режим NAT добавьте публикацию порта ssh (Настройки-Сеть-Проброс портов). Используйте порт 2221 и адрес 127.0.0.10. Пример настройки показан на рис. Обратите внимание, что адрес 10.0.2.15 принадлежит тому сетевому интерфейсу c7-1 который настроен на работу через режим NAT. Он может быть другим!



2. С реального компьютера подключитесь один раз к машине c7-1 с помощью утилиты ssh под

пользователем root и второй раз под созданным пользователем.

5. С помощью команды ss определите адреса и номера портов, с которого и на который осуществлено подключение. Консольный вывод команд сохраните для отчета.
6. С помощью команды who определите номера виртуальных терминалов пользователей. Консольный вывод команд сохраните для отчета.
7. Далее команды можно вводить в терминале на вашем основном компьютере и использовать copy\paste!
8. С помощью команды scp скопируйте произвольный файл на машину c7-1 и любой файл с машины c7-2. Команды сохраните для отчета.

Часть 4. Установка и настройка NAT в iptables

1. На хосте c7-1 установите iptables
2. Настройте на хосте клиентский NAT (действие SNAT или MASQUERADE), так чтобы внешняя сеть стала доступна из внутренней сети.

```
iptables -t nat -A POSTROUTING -o ИМЯ-СЕТЕВОГО-ИНТЕРФЕЙСА-NAT -s 10.0.0.0/24 -j MASQUERADE
```

или

```
iptables -t nat -A POSTROUTING -o ИМЯ-СЕТЕВОГО-ИНТЕРФЕЙСА-NAT -s 10.0.0.0/24 -j SNAT --to-source МОЙ-ВНЕШНИЙ-iP
```

3. Добавьте разрешения для прохождения сквозь c7-1 трафика во внутреннюю сеть:

```
iptables -A FORWARD -d 10.0.0.0/24 -j ACCEPT
iptables -A FORWARD -s 10.0.0.0/24 -j ACCEPT
```

В реальности так не поступают! Это учебный пример.
В реальности ограничивают подключение по направлению и доступным портам.

4. Проверьте с хоста c7-2 доступность любого хоста в Интернет.

5. Настройте публикацию порта tcp\22 на хосте c7-2 на порту tcp\55022 на внешнем сетевом интерфейсе c7-1.

```
iptables -t nat -A PREROUTING -i ИМЯ-СЕТЕВОГО-ИНТЕРФЕЙСА-NAT -p tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22
```

6. По подобию Части 3 настройте в VirtualBox публикацию порта 55022 на tcp порт 2222 адреса 127.0.0.10
7. С реального хоста подключитесь по ssh к хосту c7-2.
8. Выведите на консоль текущие правила iptables. Консольный вывод сохраните для отчета.
9. На хосте c7-1 установите пакет iptables-persistent. После его установки появится диалог, в котором надо дать согласие. После этого будет создан файл /etc/iptables/rules.v4 в который будут перенесены созданные вами правила. Далее при перезагрузке системы правила будут загружаться из этого файла. Без этого при перезагрузке правила «пропадут».
10. Далее все вносимые изменения надо сохранять утилитой iptables-save:
    `iptables-save > /etc/iptables/rules.v4`
11. Правила из файла можно восстановить так:
    `iptables-restore /etc/iptables/rules.v4`
    В принципе можно редактировать файл, а потом использовать эту утилиту.
12. Убедитесь, что правила сохраняются при перезагрузке ОС.

Часть 5. Настройка прав на файлы и каталоги

Задания этой части выполняются на хосте c7-2

1. Напишите скрипт, который создает пользователей с именем uN и паролем DerParolN, где N – порядковый номер пользователя. Количество создаваемых пользователей и начальный номер следует передать через параметры скрипта (например ./script.sh 8 5).
2. С помощью скрипта создайте 5 пользователей.
3. Создайте группу с произвольным именем.
4. Создайте каталог /DATA. Сделайте так, чтобы все члены группы могли писать, удалять любые файлы в этом каталоге, а все остальные пользователи системы могли бы только читать данные. Включите одного из созданных пользователей в группу. Проверьте работу прав.
5. Создайте каталог /DATA/sec1 в который любой пользователь системы сможет записать данные, но удалить сможет только те файлы, которые записал сам.
6. Создайте каталог /DATA/sec2 в который сделает так, чтобы пользователь созданный в части 2 п.1 смог бы читать, изменять и удалять файлы в каталоге, пользователи, созданные в Части 5 п 1 смогли бы только читать файлы, а все остальные пользователи системы не могли даже просматривать содержимое каталога (можно создать дополнительно необходимые группы).
7. Создайте каталог /DATA/sec3, в который скопируйте исполняемый файл редактора nano и сделайте так, чтобы любой пользователь смог изменять с его помощью файлы в каталоге /DATA/sec2
8. Выведите на экран все права на файлы и каталоги, назначенные в этой части. Сохраните данные для отчета.

Часть 6. Настройка аутентификации по ключу

1. На реальном компьютере создайте пару ssh ключ для аутентификации (реально создается пара ключей – открытый и закрытый ключ, но об этом будет отдельная лекция позже).
2. Передайте открытый ключ (он с расширением pub). В unix-подобных ОС есть утилита ssh-copy-id. В windows ее нет. Придется добавлять текст файла открытого ключа в файл .ssh/authorized_keys . Можно через ssh или скопировав файл через scp.
3. Отредактируйте конфигурацию sshd так, чтобы пользователю root нельзя было подключаться по ssh, была включена аутентификация по публичным ключам, для поиска ключей использовался файл .ssh/authorized_keys.
4. Проверьте подключение без ввода пароля.
5. Скопируйте с виртуальной машины файлы скрипта из части 5 на реальный хост, используя scp и аутентификацию по ключу. Команду сохраните для отчета.

Часть 7. Sudo

1. На хосте c7-1 установите sudo.
2. Сделайте так, чтобы созданный пользователь из Части 2 п.1 смог бы повышать привилегии до root с помощью sudo.
3. Сделайте так, чтобы первый из созданных в Части 5 пользователей смог бы с помощью sudo и команды passwd менять пароли другим пользователям, но не смог бы использовать другие утилиты от имени root.
4. Проверьте работу прав.

Часть 8. Получение информации о пользователях

1. На хосте c7-2 выведите информацию по входах пользователей в систему за текущий месяц.

2. Одной командой получите информацию для созданного в Части 2 п.1 пользователя включая: его UID (User ID), GID (Group ID) идентификатор основной группы пользователя, все группы, в которые входит пользователь. Команду и консольный вывод сохраните для отчета.

Инструментальные средства:

Утилиты:	sysctl systemctl ip useradd ss iptables iptables-save iptables-restore ls adduser passwd chmod chown who ssh-keygen scp groupadd useradd groupdel userdel groupmod usermod last
newusers, passwd, chmod, chgrp, chown, chpasswd, groups, id.	
Файлы:	/etc/ssh/sshd_config
Утилиты работы с текстом:	echo, grep, sed
Редакторы:	vi, nano

Содержание отчета

Требуется подготовить отчеты в формате DOC\DOCX или PDF. Отчет содержит титульный лист, артефакты выполнения и ответы на вопросы и задания.

Артефакты:

1. Файлы interfaces с обоих хостов.
2. Консольный вывод из Части 2, п.3
3. Консольный вывод из Части 3, п 5,6,8
4. Консольный вывод команды из части 4 п.8
5. Скрипт из Части 5, п.1
6. Консольный вывод из Части 5, п.8
7. Команду из части 6 п.1 и п. 5.
8. Измененные параметры конфигурационного файла из Части 6 п.3
9. Измененные параметры конфигурационного файла из части 7 п.3
10. Консольный вывод из части 8 п.1 и п.2

Вопросы и задания:

1. В части 4 вы использовали готовые команды для настройки NAT. Поясните какие параметры передаются в ключах команды iptables.
2. При создании ключей ssh программа-генератор предлагает ввести пароль. Зачем он нужен и для чего используется?
3. При первом подключении по ssh к новому серверу вам выводится хэш и программа предлагает принять его или отклонить. Зачем это нужно?
4. Как на сервере ssh определить сколько подключений по ssh есть и от каких пользователей?
5. Если у двух пользователей в Linux будут одинаковые пароли, то сможем ли мы понять это по данным в файле /etc/shadow ? Почему?
6. Заполните таблицу, описывающую действие различных атрибутов прав (r, w, x) и атрибутов безопасности (suid, sgid, stiky bit) при назначении их файлу или каталогу. В таблице должны быть следующие столбцы:

атрибут	сокращенное название	значение действия для файла	значение действия для каталога

В таблице должно быть 6 строк, не считая заголовок.

7. В Linux существует расширенные права на файлы или каталоги. Работать с ними можно с

помощью утилит satfacl и getfacl. Приведите пример команды, с помощью которой мы можем дать конкретному пользователю все права на файл, не делая его владельцем и не добавляя его в группы.

Отчет выслать в течение 4-х недель на адрес edu-net@yandex.ru.

Поддержка работы

Дополнительные материалы по теме курса публикуются на Telegram-канале ITSMDao (t.me/itsmdao). Обсуждать работу и задавать вопросы можно в чате ITSMDaoChat (t.me/itsmdaochat).