ОС UNIX. Лекция 3
Управление сетью и инструменты обеспечения безопасности

Михаил Пожидаев

18 сентября 2023 г.

Понятие сетевого интерфейса

Сетевой интерфейс — запись в ядре операционной системы, получающая IP-адрес и используемая для обмена данными с удалёнными компьютерами.

- Каждая сетевая карта имеет свой сетевой интерфейс с некоторым уникальным именем.
- Различается два состояния интерфейса: включен и выключен.
- Интерфейсы позволяют обрабатывать статистику передачи данных.
- Пример: `lo` — интерфейс обратной петли, который возвращает все полученные пакеты обратно на локальный компьютер.

Основные сетевые утилиты

1. `ifconfig` и `route` — управление интерфейсами и таблицей маршрутизации.
2. `ping` и `traceroute` — отладка сетевых маршрутов.
3. `telnet` и `ssh` — получение терминала на удалённой системе.
4. `nmap` — сканирование портов.
5. `iptables` — утилита управления фильтрацией пакетов.
6. `rsync` — передача файлов между узлами сети.
7. `pppd` — демон протокола PPP.

Существуют интерактивные сервисы управления сетью, такие как `network-manager`.

Таблица маршрутизации

```
      Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         router          0.0.0.0         UG    0      0        0 enp2s0
192.168.1.0     *               255.255.255.0   U     0      0        0 enp2s0
```

Управление беспроводными соединениями

1. `iwconfig` — управление параметрами интерфейса для беспроводных подключений.
2. `iwlist` — сканер беспроводных соединений.
3. `wpa_supplicant` — демон для поддержки протоколов WEP, WPA и WPA2.

Базовые инструменты отладки

1. `ping` — отправление пакета echo request с целью получения echo reply.
2. `traceroute` — последовательное отправление ICMP-пакетов до некоторого узла с увеличением TTL от единицы и до тех пор, пока не придёт ответ от целевого адреса. Получение ответов о превышении TTL позволяет установить фактический маршрут до узла.

Настройка фильтрации пакетов

Набор правил для фильтрации IP-пакетов в системах GNU/Linux можно редактировать при помощи утилиты `iptables`. Основные цепочки пакетов:

1. `INPUT` — правила для пакетов, адресованных системе непосредственно.
2. `FORWARD` — правила для пакетов, выбранных для дальнейшей передачи.
3. `OUTPUT` — правила для исходящих пакетов.

Применение правил фильтрации

1. Обрабатывается таблица маршрутизации:
   - если пакет для локальной системы, применяется правила `INPUT`;
   - если пакет для дальнейшей передачи, применяется цепочка `FORWARD`.
2. Цепочка `OUTPUT` применяется отдельно для всех исходящих пакетов.

Критерии для анализа пакетов

1. Интерфейс.
2. Протокол.
3. Исходящий или целевой IP-адрес.
4. Исходящий или целевой порты для TCP или UDP.
5. Набор флагов установки соединения.

Пример конфигурирования iptables

```
/sbin/iptables -P INPUT DROP
/sbin/iptables -P FORWARD DROP
/sbin/iptables -P OUTPUT ACCEPT
#lo interface activity
/sbin/iptables -A INPUT -i lo -j ACCEPT
/sbin/iptables -A OUTPUT -o lo -j ACCEPT
# Permitting packets of already established connections
/sbin/iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Использование ssh

Сервер ssh (Secure SHell) позволяет открывать пользовательские сессии на удалённом компьютере с полным шифрованием всех пересылаемых данных.

При помощи ssh можно:

1. Открывать полноценные интерактивные сессии на удалённом компьютере.
2. Удалённо вызывать различные команды с перенаправлением стандартных потоков ввода/вывода.
3. Использовать ssh в качестве транспорта для других приложений.

Аутентификация SSH с открытым ключом

1. Клиент заявляет желание подключиться к серверу и называет желаемое имя пользователя.
2. Сервер генерирует случайную строку.
3. Сгенерированная строка шифруется открытым ключом, ассоциированным с заявленным пользователем, и пересылается клиенту.
4. Если у клиента есть парный секретный ключ, он дешифрует строку и высылает обратно серверу.
5. Сервер сверяет исходную и полученную от клиента строки и предоставляет доступ в случае их совпадения.

Утилита rsync

1. Использует ssh в качестве транспорта.
2. Содержит механизмы определения необходимого для пересылки набора данных.

Пример вызова:

```
rsync -vaP /music/horowitz/
laptop:music/horowitz/
```

Установление соединения

```
struct hostent* host_addr = NULL;
int fd;
struct sockaddr_in saddr;
fd = socket(AF_INET, SOCK_STREAM, 0);
bzero(&saddr, sizeof(struct sockaddr_in));
saddr.sin_family = AF_INET;
saddr.sin_port = htons(the_port_you_want);
host_addr = gethostbyname(the_host_you_want);
memcpy(&saddr.sin_addr, host_addr->h_addr, 4);
connect(fd, (struct sockaddr*)&saddr,
  sizeof(struct sockaddr_in));
```

Утилиты подсчёта хэш-сумм

- `md5sum`
- `sha1sum`
- `sha256sum`
- `sha512sum`

GNU Privacy Guard

GNU Privacy Guard — свободная реализация утилит обеспечения безопасности, совместимая со стандартом Open PGP. Её функции включают симметричное и асимметричное шифрование, а также инструменты работы с цифровой подписью.

Пример команды симметричного шифрования:

```
gpg -c --cipher-algo aes --passphrase-fd 0 input.txt < key.txt
```

Пример дешифровки:

```
gpg --decrypt-file --passphrase-fd 0 input.txt.gpg < key
```

Утилита cryptsetup

Linux Unified Key Setup (LUKS) позволяет создать прослойку между двумя блочными устройствами с функционалом прозрачного шифрования.

Форматирование раздела:
`sudo cryptsetup luksFormat /dev/sdb1`
Подключение шифрованного раздела /dev/sdb1 для отображения в /dev/mapper/private:
`sudo cryptsetup luksOpen /dev/sdb1 private`
Отключение шифрованного раздела:
`sudo cryptsetup luksClose private`

Спасибо за внимание!

Всё о курсе: https://marigostra.ru/materials/unix.html
E-mail: msp@luwrain.org
Канал в Телеграм: https://t.me/MarigostraRu