Министерство науки и высшего образования Российской Федерации

Рязанский государственный радиотехнический университет им. В.Ф. Уткина

Сетевые возможности Linux

Методические указания к практическим занятиям

Рязань 2020

УДК 681.3.06

Сетевые возможности Linux: методические указания к практическим занятиям / Рязан. гос. радиотехн. ун-т.; сост. А.А. Митрошин, В.Г. Псоянц. – Рязань, 2020. – 16 с.

Содержат описание практического занятия, используемого в курсе «Операционная система Linux». Могут использоваться при изучении других курсов, связанных с операционной системой Linux.

Предназначены для студентов очной, заочной и очно-заочной форм обучения направления подготовки «Информатика и вычислительная техника». Могут использоваться для студентов других направлений подготовки.

Могут использоваться как методические указания к лабораторным работам в курсах, связанных с изучением операционной системы Linux и свободно распространяемого программного обеспечения.

Ил. --. Библиогр.: -- назв.

Операционная система Linux, сетевые возможности

Печатается по решению редакционно-издательского совета Рязанского государственного радиотехнического университета.

Рецензент: кафедра САПР вычислительных средств Рязанского государственного радиотехнического университета (зав. кафедрой засл. деят. науки и техники РФ В.П.Корячко)

Сетевые возможности Linux

Составители: Митрошин Александр Александрович

Псоянц Владимир Грикорович

Редактор \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Корректор \_\_\_\_\_\_\_\_\_\_\_\_\_

Подписано в печать \_\_\_\_\_\_\_\_. Формат бумаги 60×84 1/16.

Бумага газетная. Печать трафаретная. Усл. печ. л. 1,0.

Уч-изд. л. 1,0. Тираж 50 экз. Заказ

Рязанский государственный радиотехнический университет.

390005, Рязань, ул. Гагарина, 59/1.

Редакционно-издательский центр РГРТУ.

**Получение сведений о текущих сетевых настройках**

Утилита ifconfig (**i**nter**f**ace **conf**iguration) используется для настройки любых сетевых интерфейсов, установленных на сетевом узле. Помимо настройки интерфейса с ее помощью можно получить сведения об интерфейсе. На рис. 1 приводится пример вывода утилиты ifconfig –a. Опция –a позволяет получить сведения и о соединениях, которые не работают. Часть данных ifconfig получает при обращении с помощью системного вызова к открытому сетевому сокету, а некоторые считывает из /proc.

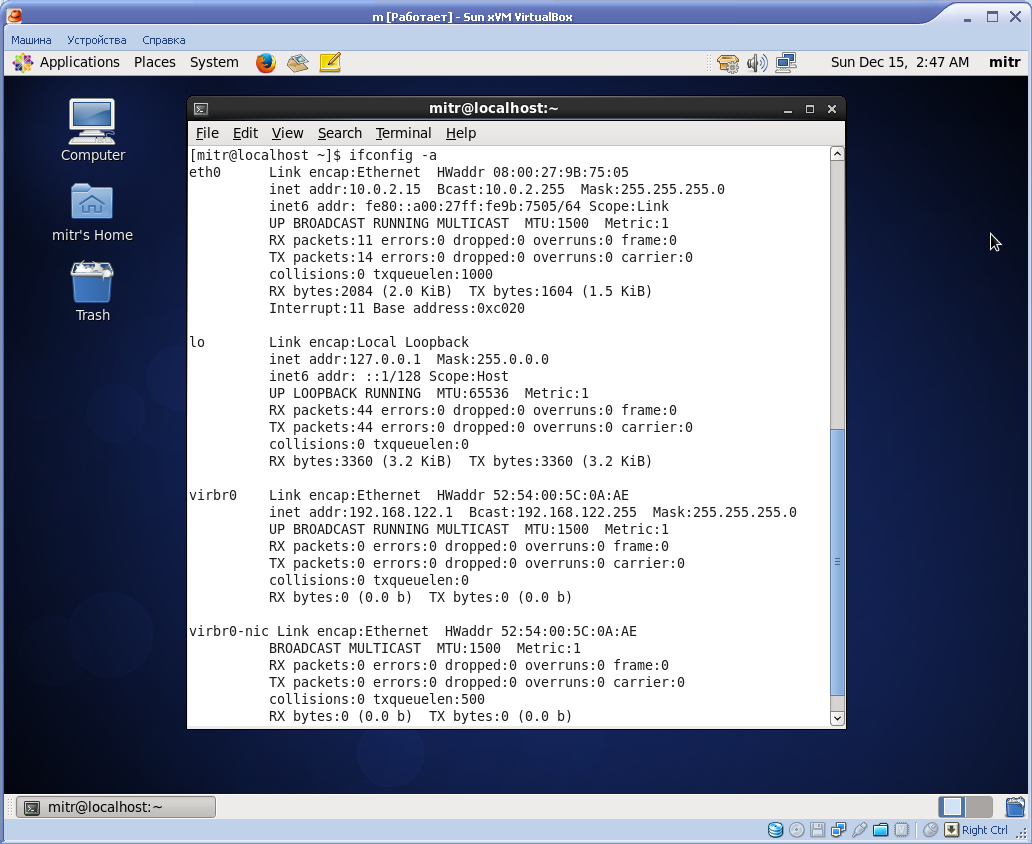


Рисунок 1. Пример вывода утилиты ifconfig

Сеть начинается в месте подключения к среде передачи данных, то есть на сетевом интерфейсе. Название сетевого интерфейса состоит из его типа и порядкового номера, который определяется тем, каким по счету его распознало ядро. Все сетевые интерфейсы Ethernet в Linux называются eth*номер*, начиная с eth0.

Как видно из рис. 1, интерфейс eth0 является интерфейсом инкапсуляции линии связи (Link encap) типа Ethernet. Аппаратный адрес (HWaddr) – это МАС-адрес (**M**edia **A**ccess **C**ontrol – управление доступом к среде, состоит из 6 байтов, которые записывают в шестнадцатеричной системе счисления; каждая Ethernet-карта имеет собственный уникальный МАС-адрес, поэтому он используется для определения отправителя и получателя в рамках одной среды Ethernet. Если идентификатор получателя неизвестен, то используется аппаратный широковещательный адрес FF:FF:FF:FF:FF:FF. Сетевая карта, получив широковещательный фрейм или фрейм, МАС-адрес получателя в котором совпадает с ее МАС-адресом, отправляет его на обработку системе), inet addr – ip4-адрес, Bcast – широковещательный адрес, Mask – маска сети, inet6 – ip6-адрес. Параметр MTU (**M**aximum **T**ransfer **U**nit) определяет наибольший размер фрейма.

Большая часть остальных сведений об интерфейсе является статистической. Исключение составляет последняя строка, в которой указывается базовый аппаратный адрес и канал прерывания IRQ.

Интерфейс lo (**lo**opback, локальный сетевой узел, который используется для организации сетевых взаимодействий компьютера с самим собой) соответствует адресу localhost. Любая система обладает интерфейсом localhost, которому соответствует адрес 127.0.0.1.

Когда компьютер с некоторым IP-адресом решает отправить пакет другому компьютеру, он выясняет, принадлежит ли адресат той же локальной сети, что и отправитель. Для этого на IP-адрес получателя накладывается сетевая маска и вычисляется адрес сети, которой принадлежит получатель. Если этот адрес совпадает с адресом сети отправителя, значит оба находятся в одной локальной сети. Это в свою очередь означает, что аппаратный адрес получателя должен быть известен отправителю. МАС-адреса компьютеров локальной сети хранятся в специальной таблице ядра, называемой таблицей ARP. Просмотреть содержимое этой таблицы можно с помощью команды arp -a.

ARP-таблица отражает соответствие между IP- и MAC-адресами. Это динамическая таблица, устаревшие соответствия из нее удаляются. Для установления соответствия между адресами сетевого и интерфейсного уровня используется протокол ARP (**A**ddress **R**esolution **P**rotocol – протокол разрешения адресов). Для преобразования IP в МАС он работает следующим образом: отправляется широковещательный фрейм типа «ARP-запрос», содержащий IP-адрес получателя. Компьютер, адрес которого указан в запросе, возвращает отправителю пустой фрейм типа «ARP-ответ», в поле «отправитель» которого указан МАС-адрес отправителя. На основании полученных данных заполняется ARP-таблица.

В том случае, когда компьютер, которому надо отправить сообщение, не находится в локальной сети компьютера-отправителя, то пакет нужно отправить какому-то абоненту локальной сети, чтобы он перенаправил его дальше. Этот абонент, маршрутизатор, подключен к нескольким сетям и пересылает пакеты меду ними по определенным правилам.

Сведения о таблице маршрутизации могут быть получены с помощью утилиты route. По умолчанию route отображает информацию с указанием символьных имен, однако в большинстве случаев IP-адреса оказываются более информативными, поэтому часто route используется с опцией –n. Вывод команды route без опций и с опцией –n показан на рис. 2. Это вывод для компьютера, который не выполняет функции маршрутизатора. Аналогичный вывод маршрутизатора был бы значительно сложнее.

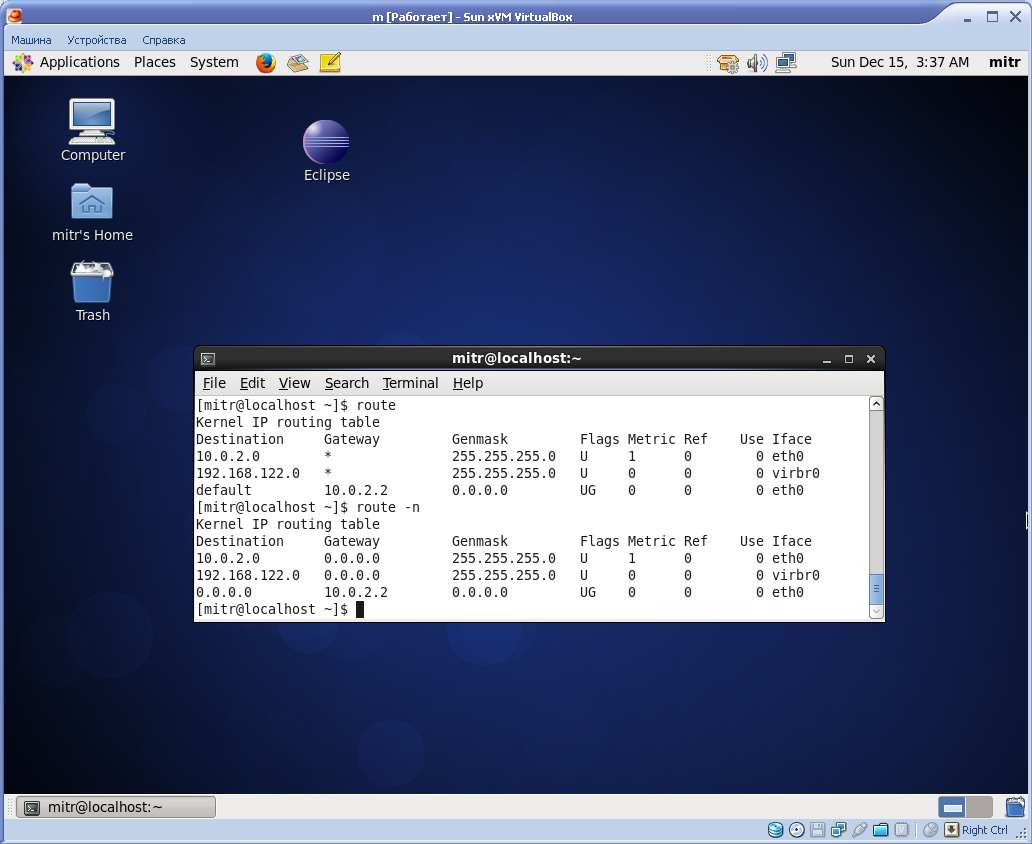


Рисунок 2. Вывод команды route без опций и с опцией –n

Поле «Destination» - адрес сети назначения.

Поле «Gateway» - адрес следующего маршрутизатора.

Поле «Flags» описывает состояние маршрута. Для описания состояния используются следующие символы:

- U – маршрут активен и работоспособен (Up);

- H – признак специфического маршрута к определенному хосту. Маршрут ко всей сети, к которой принадлежит данный хост, может отличаться от данного маршрута;

- G – означает, что маршрут пакета проходит через промежуточный маршрутизатор (gateway). Отсутствие этого флага отмечает непосредственно подключенную сеть;

- D – означает, что маршрут получен из сообщения Redirect (перенаправление) протокола ICMP. Этот флаг может присутствовать только в таблице маршрутизации конечного узла. Признак означает, что конечный узел в какой-то предыдущей передаче пакета выбрал не самый рациональный следующий маршрутизатор на пути к данной сети, и этот маршрутизатор с помощью протокола ICMP сообщил, что все последующие пакеты к данной сети нужно отправлять через другой следующий маршрутизатор. Протокол ICMP может посылать сообщения только узлу-отправителю, поэтому у промежуточного маршрутизатора этот признак встретиться не может. Признак никак не влияет на процесс маршрутизации, он только указывает администратору на источник появления записи;

- ! (Reject - отказ) – обозначает маршрут, через который не проходят пакеты. Если посылается пакет узлу или сети, которые помечены !, то будет получено сообщение , что маршрут не активен (Down). Это не означает, что по этому маршруту пакеты не могут передаваться в обратном направлении.

Поле «Ref» показывает, сколько раз на данный маршрут ссылались при продвижении пакетов.

Поле «Use» отражает количество байтов, переданных по данному маршруту.

Сеть 0.0.0.0 – ни содержит ни одного бита на сетевую маску, поэтому ей принадлежат любые адреса. Такая запись в таблице маршрутизации называется «маршрут по умолчанию». Если маршрут по умолчанию не задан, то попытка связаться с удаленным компьютером может закончиться ошибкой «No route to host», поскольку система не сможет определить, кому переслать пакет.

Получив IP-пакет, система последовательно сравнивает поле назначение пакета с записями в таблице маршрутизации. Если сеть адресата совпадает с сетью из таблицы маршрутизации, то пакет пересылается по адресу, указанному в поле «Gateway». Этот адрес используется вместо поля адресата и поиск возобновляется с начала таблицы. Если поле «Gateway» - нулевое, значит абонент находится в локальной сети и пакет надо передать на уровень ниже. Если ни одна сеть не подходит, выдается сообщение об ошибке.

**Протокол ICMP**

Протокол ICMP (**I**nternet **C**ontrol **M**essage **P**rotocol) предназначен для передачи служебных сообщений. В отличие от протоколов TCP и UDP пакеты протокола ICMP не передаются через сеть в составе пакетов IP. Протокол ICMP обладает собственным заголовком пакета. Этим протоколом пользуются, в частности, известные утилиты ping и traceroute.

Пакеты ICMP могут быть нескольких типов, пакеты некоторых типов могут нести коды нескольких сообщений. В табл. 1 приведены некоторые типы ICMP пакетов и коды сообщений.

Таблица 1. Некоторые типы ICMP пакетов и коды сообщений

|  |  |  |
| --- | --- | --- |
| **Тип пакета ICMP** | **Описание** | **Коды сообщений** |
| 0 | Echo Reply Message (эхо-ответ) | 0 |
| 3 | Destination Unreachable Message  (Приемник недоступен) | 0 – сеть недоступна;  1- сетевой узел недоступен;  2 – протокол недоступен;  3 – порт недоступен;  4 – необходима фрагментация и флаг DF (**D**on’t **F**ragment – не фрагментировать) установлен;  5 – исходный маршрут не работает. |
| 4 | Source Quench Message  (сообщение об отключении источника при перегрузке с предварительным возвратом сообщения) | 0 |
| 5 | Redirect Message  (сообщение о перенаправлении) | 0 – перенаправление дейтаграммы для сети;  1 - перенаправление дейтаграммы для сетевого узла;  2 – перенаправление дейтаграммы для типа службы и сети;  3 - перенаправление дейтаграммы для типа службы и сетевого узла |
| 8 | Echo Massage  (эхо-запрос) | 0 |
| 11 | Time Exceeded Massage  (время истекло) | 0 – при передаче сообщения истекло время жизни (time to live);  1 – истекло время формирования сообщения из фрагментов (fragment reassembly time) |
| 12 | Parameter Problem Message  (проблема с параметром) | 0 – проблема, связанная с параметром |
| 13 | Timestamp Message  (отметка времени) | 0 |
| 14 | Timestamp Reply Message  (ответ на отметку времени) | 0 |
| 15 | Information Request Message  (запрос информации) | 0 |
| 16 | Information Reply Message  (ответ на запрос информации) | 0 |

Для проверки доступности сетевого ресурса можно использовать команду ping, имеющую следующий синтаксис:

**ping [опции] адрес\_назначения**

Все опции команды ping можно просмотреть в справочной системе. Наиболее часто используется опция –a. Пример использования ping приведен на рис. 3.

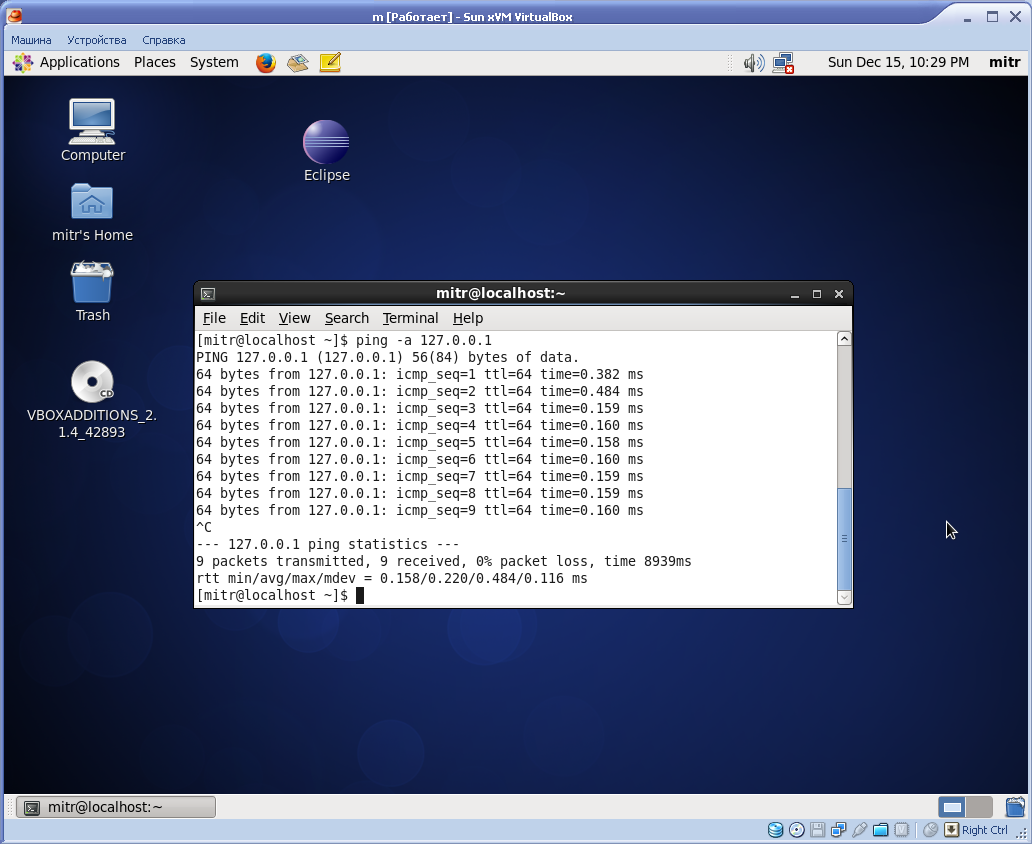


Рисунок 3. Использование ping

Утилита **traceroute** выводит список узлов, через которые проходит пакет по пути к адресату. Это приблизительный список. Дело в том, что первым трем пакетам (по умолчанию rtaceroure шлет пакеты по три) в поле TTL (**T**ime **T**o **L**ive – время жизни) устанавливается значение 1. Каждый маршрутизатор должен уменьшать это значение на 1, и если оно обнулиться, то передать отправителю ICMP-пакет о том, что время жизни закончилось, а адресат пакета еще не найден. Таким образом, на первую серию пакетов отреагирует первый маршрутизатор, и traceroute выдаст первую строку маршрута. Второй пакет (вторая серия из трех пакетов) посылается с TTL=2 и, если за две пересылки адресат не достигнут, об этом сообщит второй маршрутизатор. Процесс продолжается до тех пор, пока очередной пакет не достигнет места назначения. Строго говоря, неизвестно, каким маршрутом шла очередная группа пакетов, потому, что с тех пор, как посылалась предыдущая группа, какой-нибудь из промежуточных маршрутизаторов мог послать пакеты и другим путем.

На рис. 4 показан пример использования traceroute.

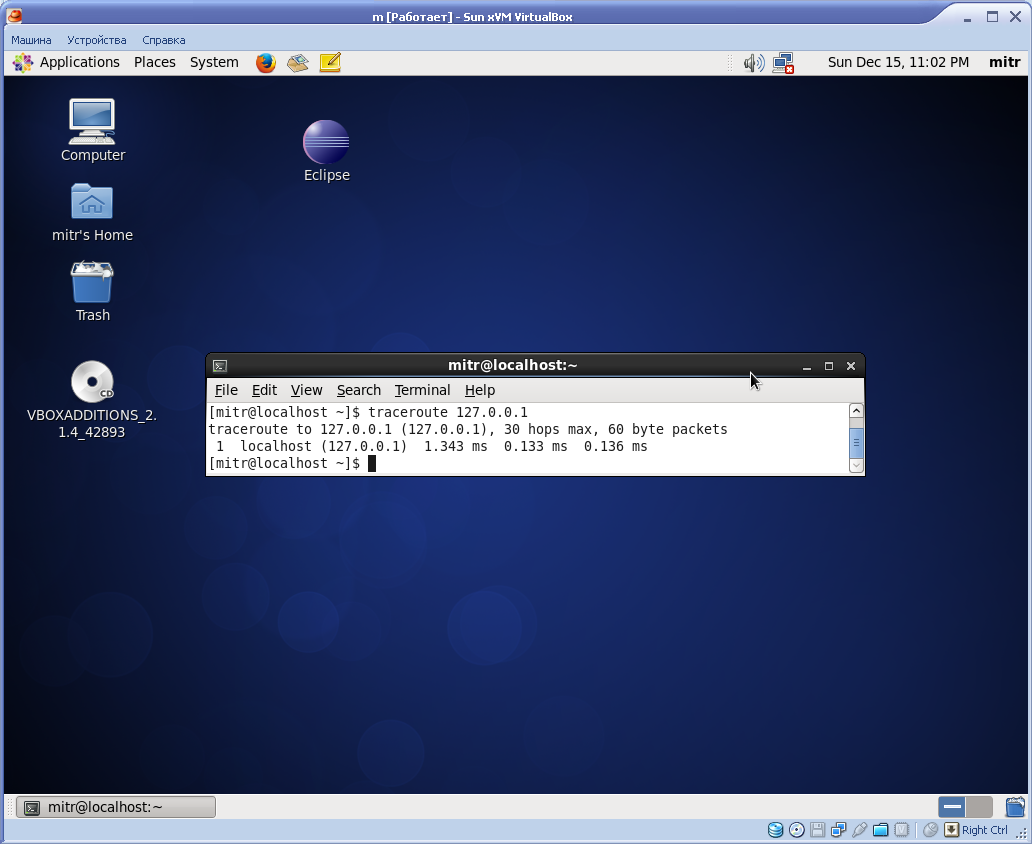


Рисунок 4. Использование traceroute

**Протоколы транспортного уровня**

Если задачей уровня межсетевого взаимодействия, к которому относится протокол IP, является передача данных между любой парой сетевых интерфейсов в составной сети, то задача транспортного уровня, которую решают протоколы TCP и UDP, заключается в передаче данных между любой парой прикладных процессов, выполняющихся в сети. После того, как пакет средствами протокола IP доставлен на сетевой интерфейс компьютера-получателя, данные необходимо направить конкретному процессу-получателю. Каждый компьютер может выполнять несколько процессов, а каждый прикладной процесс может иметь несколько точек входа, выступающих в качестве адресов назначения для пакетов данных.

Пакеты, поступающие на транспортный уровень, организуются операционной системой в виде множества очередей к точкам входа различных прикладных процессов. В терминологии TCP/IP такие очереди называются *портами*. Порт однозначно определяет приложение в пределах компьютера.

Существуют два способа присвоения порта приложению – централизованный и локальный. За каждым из способов закреплен свой диапазон номеров портов: для централизованного - от 0 до 1023, для локального – от 1024 до 65535. Если процессы представляют собой популярные общедоступные службы, такие как FTP, HTTP, DNS и др., то за ними закрепляются стандартные присвоенные номера, часто называемые *хорошо известными номерами*. Хорошо известные номера и портов и названия соответствующих служб хранятся в файле /etc/services (рис. 5)

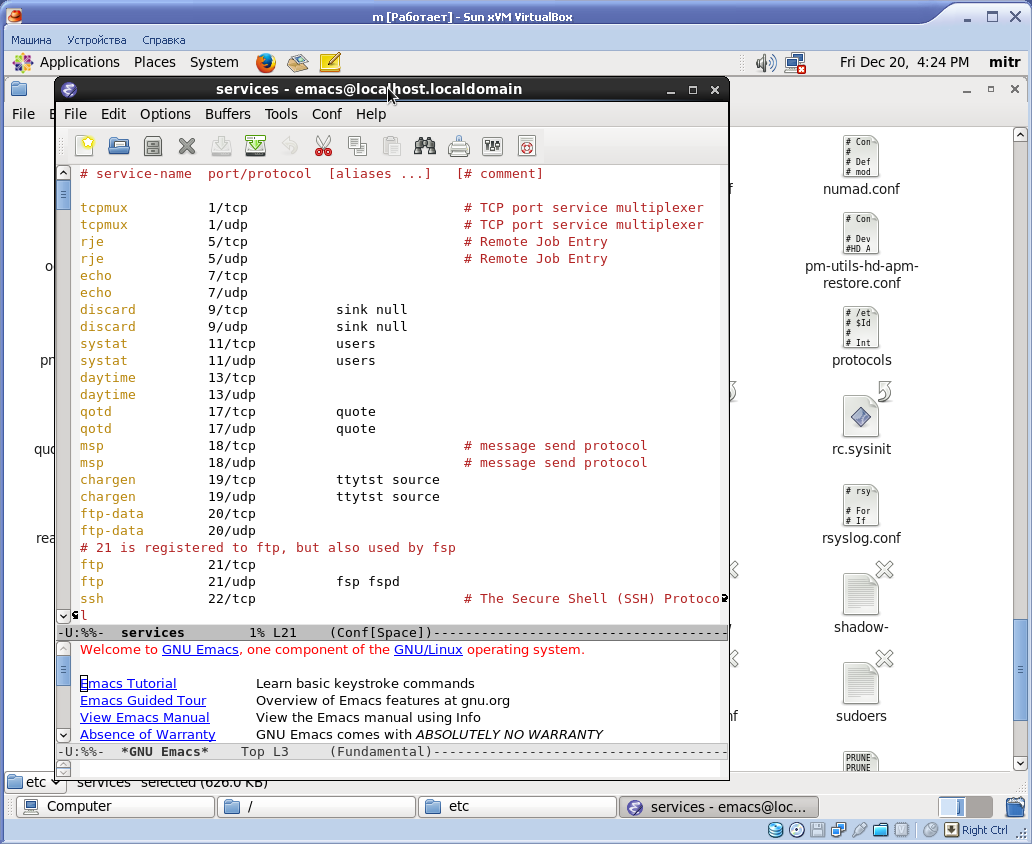


Рисунок 5. Фрагмент файла /etc/services

В файл /etc/services можно добавлять новые службы и уникальные номера портов, однако не следует в этот файл добавлять службы, которые в нем уже существуют. Относительно номеров портов в Linux нужно сделать важное замечание – порты с номерами ниже 1024 рассматриваются как привилегированные. Только root может выполнять связывание служб с этими номерами.

Получить список установленных на компьютере соединений, а также служб-обработчиков можно с помощью команды netstat (рис. 6).

По умолчанию netstat выводит только список открытых соединений. Если при вызове добавить опцию –a, то будут показаны все сондинения.

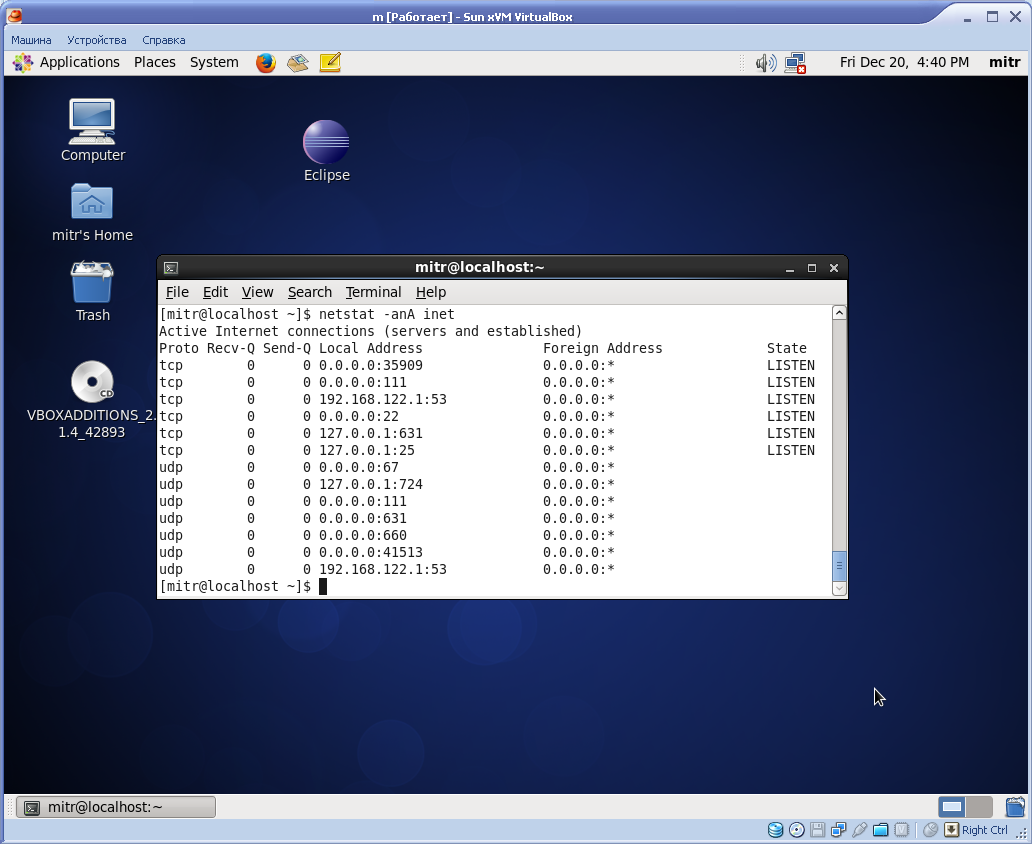


Рисунок 6. Вывод команды netstat –anA inet

Первая строка, отображаемая утилитой netstat, сообшает о том, какие сведения отображаются в данный момент утилитой. В приведенном примере в первой строке указано «***Active Internet connections (servers and established)***», то есть активные соединения с Internet. Термин «server» в данном случае означает программы, открывшие порт и ожидающие поступления запросов на соединение. Такие программы *слушают в очереди ожидания*, однако они еще не обмениваются данными с клиентом.

Вторая строка – заголовок для раздела вывода netsat. «Proto» - протокол Часто в этом столбце отображаются tcp и udp, однако могут быть т другие протоколы, например raw.

В столбцах «Recv-Q» (Receive Queue – очередь приема) и «Send-Q» (Send Queue – очередь на передачу) отображается соответственно количество пакетов, принятых из сети, но не скопированных пользовательской программой, подключенной к сокету, и количество байтов, прием которых не подтвержден удаленной системой. После того, как сокет TCP закрывается, как правило, удаленный узел не подтверждает прием одного пакета.

В столбцах локальный (Local) и удаленный (Foreign) IP-адреса указываются адреса, между которыми устанавливается соединение. Адреса указываются в формате *адрес:порт*. Протокол TCP подключает сервер к клиенту таким образом, что данные посылаются по одному порту, а принимаются по другому. Порт возврата жестко не задан; это может быть любой незанятый порт с номером выше 1024.

В процессе первого соединения клиент сообщает серверу номер выбранного порта и ожидает поступления на этот порт пакета SYN-ASC от сервера. Если сервер не может использовать этот порт, потому что он занят, клиенту передается сообщение ICMP Type 3 Code 3 «Port Unreachable» - порт недоступен. После этого клиент случайным образом выбирает другой порт. Когда клиент через выбранный им порт принимает от сервера пакет SYN, он возвращает серверу пакет ACK. После этого соединение считается установленным и начинается передача данных.

Когда соединение разрывается, выполняется похожая последовательность процедур.

В столбце «State» (состояние) указывается состояние соединения. В столбе могут отображаться следующие состояния:

- ESTABLISHED;

- SYN\_SENT;

- SYN\_RECV;

- FIN\_WAIT1;

- FIN\_WAIT2;

- TIME\_WAIT;

- CLOSED;

- CLOSE\_WAIT;

- LAST\_ACK;

- LISTEN;

- CLOSING;

- UNKNOWN.

Понятие состояния для протокола UDP не имеет смысла,

Если к команде netstat добавить опцию –e, то к выводу команды добавляется столбец «User» (пользователь), в котором будет указываться UID пользователя, от имени которого запущен процесс, создавший соединение (рис. 7).

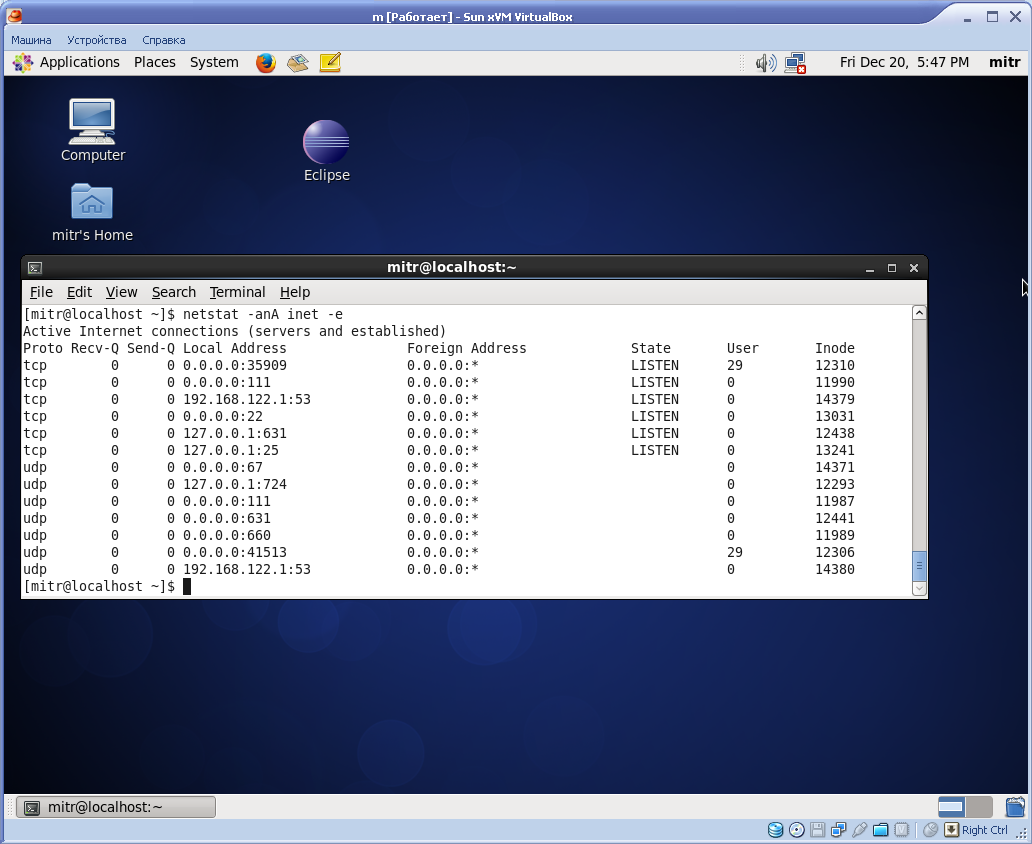


Рисунок 7. Вывод команды netstat –anA inet -e

Для того, чтобы просмотреть активные доменные сокеты UNIX, используется команда netstat –unix (рис. 8).

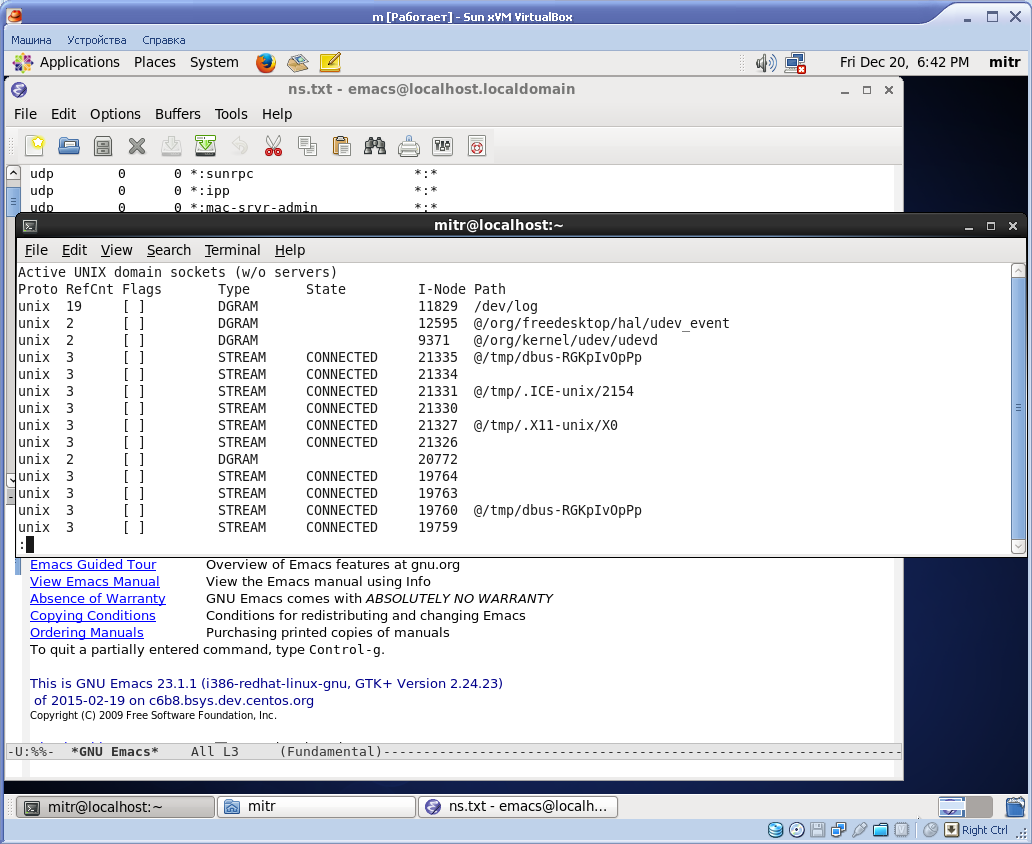


Рисунок 8. Вывод команды netstat - -unix

В выводе присутствуют заголовки:

- Proto – протокол; для сокетов Unix, единственным допустимым значением является unix.

- RefCnt – счетчик ссылок; указывается количество подсоединенных процессов.

- Flags – набор флагов; как правило, это поле пусто, однако, если поле RefCnt равно 0 и соответствующие процессы ожидают поступление запросов на соединение, поле флагов может быть равно ACC (SO\_ACCEPTION - принятие), что означает, что сокет готов к приему запросов на соединение. В некоторых ситуациях могут возникать и другие флаги, например W (S)\_WAITDATA – ожидание данных) и N (SO\_NOSPACE – нет места).

- Type – тип сокета. Как правило, стоит метка STREAM (сокет с созданием соединения), однако, могут быть и другие метки: DGRAM (сокет без создания соединения), RAW (сокет передачи данных без транспортного протокола), RDM (Reliably Delivered Message – надежно передаваемое сообщение), SEQPACKED (Sequential Packet – последовательно передаваемый пакет), PACKET (пакет простого доступа к интерфейсу), UNKNOWN (неизвестный тип сокета для будущих усовершенствований).

- State – состояние сокета; могут использоваться следующие метки: FREE (сокет свободен), LISTENING (ожидает поступления запроса), CONNECTING (соединение устанавливается), CONNECTED (соединение установлено), DISCONNECTING (соединение разрывается), UNKNOWN (неизвестное состояние). Поле может быть и пустым. При нормальном функционировании системы сокет не может находиться в состоянии UNKNOWN.

- I-Node – номер дискриптора I-Node, соответствующий соединению. I-Node существует в каталоге /proc только в том случае, если соединение используется.

- Path – процесс, подключенный к сокету.

**Порядок выполнения работы**

1) Изучите теоретический материал.

2) Выполните на своем компьютере приведенные команды и объясните их вывод.

3) Изучите по справочной системе или с помощью cети Интернет опции команд arp, ifconfig, route, ping, traceroute, netstat, tcpdump, ip. Используйте изученные опции на своем компьютере.

4) Ответьте на контрольные вопросы.

**Контрольные вопросы**

1) Прокомментируйте вывод команды ifconfig на вашем компьютере. Опишите опции ifсonfig.

2) Прокомментируйте вывод команды arp на вашем компьютере. Опишите опции arp.

3) Прокомментируйте вывод команды route на вашем компьютере. Опишите опции route.

4) Прокомментируйте вывод команды ping на вашем компьютере. В качестве удаленной системы используйте любой известный вам сайт. Опишите опции ping.

5) Прокомментируйте вывод команды traceroute на вашем компьютере. В качестве удаленной системы используйте любой известный вам сайт. Опишите опции traceroute. Почему вывод утилиты не точен?

6) Прокомментируйте вывод команды netstat на вашем компьютере. Опишите опции netstat.

7) Прокомментируйте вывод команды tcpdump на вашем компьютере. Опишите опции tcpdump.

8) Расскажите об утилите ip и ее опциях. Приведите примеры использования утилиты ip.

9) В каком файле хранятся известные номера и портов и названия соответствующих служб?