**Шпаргалка по сетевым командам Linux**

**Введение**

Сетевые команды – обязательные инструменты для любого сетевого или системного администратора. С помощью этих команд они могут настраивать, диагностировать и устранять неполадки и управлять сетевыми подключениями системы Linux.

Поиск доступных команд и изучение документации может оказаться довольно утомительным занятием, а наличие единого ориентира имеет критически важное значение при работе над сетевыми задачами.

**В этой статье рассмотрены 20 основных сетевых команд Linux.**

Необходимые предварительные условия:

* Доступ к командной строке/терминалу
* Учетная запись администратора с привилегиями sudo

**20 системных команд Linux**

В Linux есть огромное количество полезных сетевых команд и инструментов. Как правило, эти команды отвечают за выполнение сложных сетевых задач, таких как мониторинг, настройка и устранение неполадок в сети Linux. Большая часть сетевых утилит – это часть более старого (устаревшего) пакета net-tools или более современного iproute2.

**Примечание**: команды net-tools и iproute2 можно найти в большинстве дистрибутивов Linux. Однако все же рекомендуется использовать инструменты iproute2 в силу их гибкости и скорости.

Несмотря на то, что net-tools считается устаревшим, он по-прежнему широко используется в устаревших сценариях и конфигурациях.

Точный синтаксис команды может отличаться в зависимости от ее версии. Вы можете перепроверить синтаксис с помощью вот такой команды:

man [command]

Команда man отображает страницу руководства для команды, которую вы указали в терминале.

Ниже приведен краткий обзор на 20 сетевых команд Linux.

**ip**

Команда ip – это единый сетевой инструмент для систем Linux. С помощью команды ip можно просматривать и настраивать маршрутизацию, интерфейсы, сетевые устройства и туннели.

Эта команда – часть пакета iproute2. Она заменяет многие старые сетевые инструменты, такие как команды route, ifconfig и netstat.

**Синтаксис**

Синтаксис команды ip следующий:

ip [options] object [command]

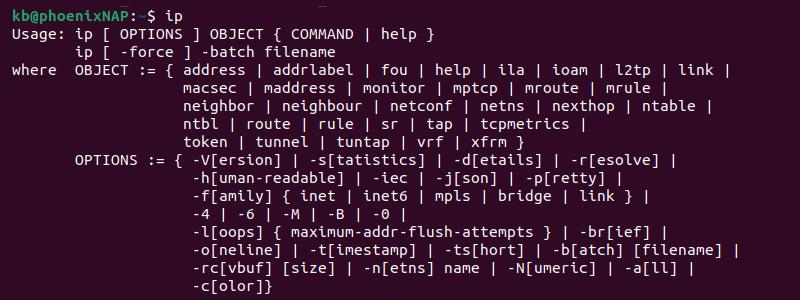
Каждая часть команды выполняет определенные действия:

* [options] – параметры командной строки, которые изменяют поведение команды
* object – доступные объекты для настройки
* [command] – подкоманда, то есть действие, которое выполняется над объектом. Они могут отличаться в зависимости от объекта

**Пример**

Команда ip без всяких параметров, объектов и подкоманд показывает справочное меню:

ip

[](https://legacy.merionet.ru/images/tmp/ad-4nxfjmkg-ahgd6-raixct4b1qwljykoeqepv4jodg-4gk3npnnebzppqyqixzzvbaubkkp07ndleaskn6-lv8un-90f-cbpwc012nllh0htudxjwoviq7uuovglapeimfapgzbmaub1mzxk05rfzoym9jfepackrmqlsrhipvyjjixrbxj-doqbi.)

Чтобы увидеть текущую версию, необходимо добавить параметр -v:

ip -V

[ip -V terminal output](https://legacy.merionet.ru/images/tmp/ad-4nxc5p1-ywpojoy9lw4bf6b0sukc59nk4knlet3ngkihurmvmiu-xns1g7v19s6-i7q10flyfvprfisrmemt7iktrok1bjalfuurmismwehbkfdx8n9oj2khkvzzdcel1qp2akk-mqfcv99fjytalkyelxk5rptxcl-gxtc6x4jmayo5su3vyifg.)

В результате мы получим версию пакета и библиотеки для утилиты ip.

**ip addr**

Команда ip addr управляет и отображает IP-адреса сетевых интерфейсов. Псевдонимами команды являются ip address или ip a.

**Синтаксис**

Синтаксис команды ip addr следующий:

ip addr [subcommand]

Список доступных подкоманд для объекта:

* add – добавляет новый адрес
* show – отображает адреса протоколов
* del – удаляет адрес
* flush – сбрасывает адреса по конкретным критериям

У каждой подкоманды есть свои дополнительные параметры и ключевые слова, которые позволяют выполнять определенные задачи по отношению к адресам сетевых интерфейсов.

**Пример**

Команда ip addr без каких-либо параметров отображает информацию о сетевом интерфейсе, в том числе связанные с ним IP-адреса:

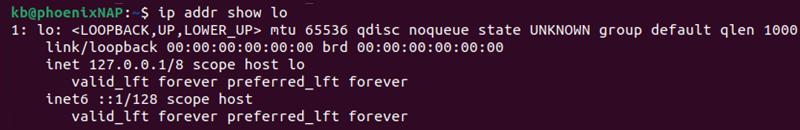
ip addr

[](https://legacy.merionet.ru/images/tmp/ad-4nxezzahfubyp7cd8jqaagy9azw0iveme5boins4sqtgfutwuu-mywddnk4skjxonfdoqlvchrtpx6rfvui6oyhnga3ijqjfbrrpt4tyl-twstieitj8x85nviojfcdwppih1dw6girqettjp-rs5yklybusavjr40xtc05lbxdfswsnfnlumbbs.)

Вывод команды ip addr show идентичен.

Для того, чтобы отобразить какой-то конкретный сетевой интерфейс, вам понадобиться команда ip addr show, после которой идет имя интерфейса. Например,

ip addr show [interface]

[](https://legacy.merionet.ru/images/tmp/ad-4nxcivbnutkqotunbkb1tu9o-618qp-mfmxw6oowx7xtv9vhjaeny3cv1o63ohqrw17q9dsx0-lll-gu039y6vhtx3maekcfaehgno7j5cte-tkqz9cjg-zsrkxf6ouf864roo7yfsajdbrlut8aew8kdvnav4kvclojbtscqhhlqnlo5t53g2-c.)

Эта команда фильтрует вывод команды ip addr и отображает информацию, которая относится только к указанному интерфейсу.

**ip link**

Команда ip link управляет и отображает информацию о сетевом интерфейсе. С ее помощью можно просматривать, изменять, включать и отключать сетевые интерфейсы.

**Синтаксис**

Синтаксис команды ip link следующий:

ip link [subcommand] [options] [interfaces]

У этой команды есть несколько подкоманд:

* show – печатает информацию о сетевом адресе
* set – меняет или добавляет информацию сетевого интерфейса
* add – добавляет новый сетевой интерфейс
* del – удаляет сетевой интерфейс

Подкоманды имеют свои дополнительные параметры, позволяющие обращаться к определённым интерфейсам.

**Пример**

Команда ip link без каких-либо дополнительных параметров отображает информацию обо всех сетевых интерфейсах.

ip link

[](https://legacy.merionet.ru/images/tmp/ad-4nxcpgbmbyheuer6rtuuhf6jo9xxlfbwwpqvwuyevysvvhwoiyprtxc4s98yl9liw8fiidohrhozr5qddc0vq3zpfzn7iia-eoy-j04m5b6vlmavyjk1b1l69qqs3nwbg3gh42n8nxmkzh3uu21y6sck3l1bqfcrawxy1hb1fxjpj7nfuefhi7rw.)

Команда ip link show дает такой же результат.

Чтобы отключить интерфейс, вам понадобятся права суперпользователя и следующий синтаксис:

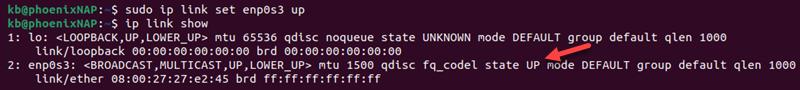
sudo ip link set [interface] down

[](https://legacy.merionet.ru/images/tmp/ad-4nxfdelf53mmmkjnd4gnpxsvxl0erlqgo48ir1elzoe9azxjlpb809rr0k6vdbzu52bgr2jzcyhjhmgffblijdf-h18vgvhgjwnosv-aaplr0khr86o60w0tnzdd0iqcma5nux-ttceppgethvpynys5pny5boywhcqpdneal21xq5fjpi3zx.)

После того, как команда будет выполнена, у интерфейса отобразится статус DOWN.

Аналогично, чтобы включить интерфейс, примените ключевое слово up:

sudo ip link set [interface] up

[](https://legacy.merionet.ru/images/tmp/ad-4nxceir9nhfzadlapvpkromqxalg6n2dnm0jgded7x5ujzsjjvr8iupi3ip8maylovar-opjstsmeqmtzuac5-ixx-qubooa-986f3yxgl-swfbw5aojx02s2xri4hjyiglxhzbxdiewey-bqndb31spp527lh5wna4dl4-ttrzxx6alzsyblvni.)

Статус интерфейса сменится на UP.

**ip route**

Команда ip route отображает и настраивает таблицу IP-маршрутизации. С ее помощью пользователи могут корректировать таблицу маршрутизации, а также выполнять прочие важные сетевые задачи.

**Синтаксис**

У команды есть определенный синтаксис:

ip route [subcommand] [options] [destination]

А с помощью подкоманд вы можете выполнять следующие действия:

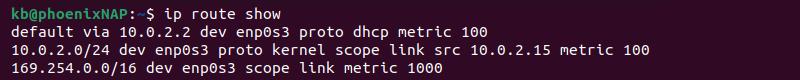
* show – отображает таблицу маршрутизации
* add – добавляет новый маршрут в таблицу
* del – удаляет маршрут из таблицы
* change – изменяет существующий маршрут

Параметр [destination] позволяет определить, куда будет направляться сетевой трафик. Дополнительные параметры помогают управлять трафиком.

**Пример**

Для того, чтобы просмотреть таблицу маршрутизации, запустите следующую команду:

ip route show

[](https://legacy.merionet.ru/images/tmp/ad-4nxd46ywzibfjcugoycokwly67afnay9oqrobcb53ong4xoqheg1hrhu3z4tbklz-k06-foubrok7-zqfe1lezgf4qck0akcvaomyxzzuaqzssuxxrhhvwismuepg05q-fsaa8vd02qwuxxeuui1n8ndzx1fkwrvssio0frkz0w8yavefh1lq-g.)

Каждая строка вывода – это отдельные маршруты в таблице.

**ifconfig**

Команда ifconfig (конфигурация интерфейса) управляет сетевыми интерфейсами системы и отображает информацию о них. Эта команда – часть пакета net-tools.

И хотя эта команда обладает меньшим функционалом в сравнении с командой ip, она по—прежнему широко используется для настройки сетевых интерфейсов.

**Синтаксис**

Синтаксис команды следующий:

ifconfig [interface] [options]

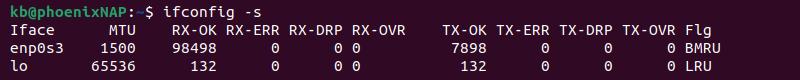
Здесь:

* [interface] – сетевой интерфейс, который необходимо настроить или информацию о котором необходимо отобразить. Параметр является необязательным. Если интерфейс не указан, отображается состояние всех активных интерфейсов.
* [options] – параметры командной строки, позволяющие выполнять конкретные действия или настраивать определенные параметры. Этот параметр также является необязательным.

**Пример**

Для того, чтобы отобразить краткий отчет по всем активным сетевым интерфейсам, запустите следующую команду:

ifconfig -s

[](https://legacy.merionet.ru/images/tmp/ad-4nxcpkkbndmyfxdwqthkydpquayyr31y6fnckj-y9qwqxat9brlylntnes8sb9znboe3lsluxdeucd0pux2ybp4c9d-m9fecolt3flu-faqsddji5gumke2kxetiehjlpi109ohvhlgtsujnokmkw7vwenrylwzdl8w6i88ip4av5rti9d-rqvp0.)

Команда выведет короткий список со всей важной информацией об активных интерфейсах.

**dig**

Команда dig запрашивает системы доменных имен (DNS - Domain Name Systems) и находит информацию о DNS-записях. Команда собирает информацию о доменном имени и соответствующих записях.

Команду dig можно использовать для выявления и устранения проблем с DNS и проверки конфигурации DNS в системе Linux. Она также подходит для создания сценариев и автоматизации задач, связанных с выявлением и устранением неполадок в сети. Эту команду используют настолько часто, что у нее появилась версия для Windows.

**Синтаксис**

Синтаксис команды следующий:

dig [options] [domain] [record type] [DNS server]

Эта команда состоит из:

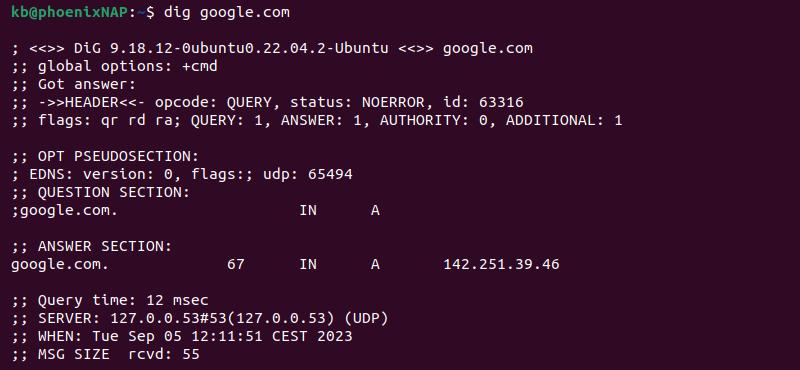
* [options] – параметры, позволяющие менять поведение команды
* [domain] – имя домена для запроса
* [record type] – тип DNS-записи для запроса. По умолчанию используется тип А.
* [DNS server] – конкретный DNS-сервер для запроса

Все параметры являются необязательными. Команда без дополнительных параметров отображает информацию о стандартном DNS-преобразователе и статистику запросов.

**Пример**

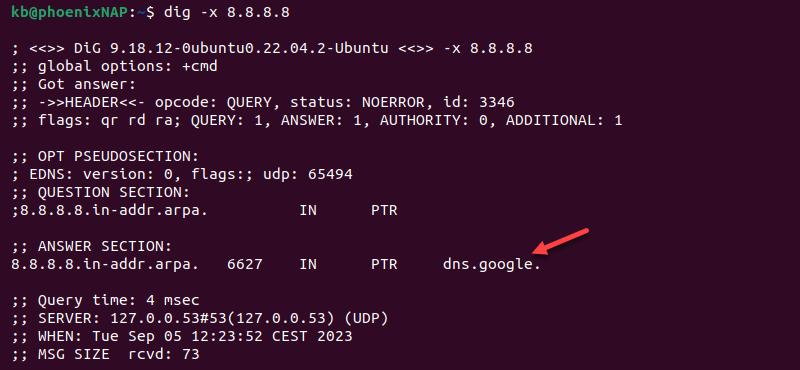
Для того, чтобы выполнить простой DNS-поиск, запустите команду с именем домена:

dig google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxck1jjb-rlpt2j1fzt9umzgualjbbv3zkxgt6vvzngdehnqpw3tydiekyqvvdfjo-diszicboj61wb6agw4sm2ehfjfe0im-bibx1crzs-dfzt0vpws-hq6hqxg3sqrm9vcx-5eqovn3grrch2sjhw6toul9k9c8ghuedxmhmx-xhgxhz9wpco.)

Или укажите IP-адрес и параметр -x для того, чтобы выполнить обратный DNS-поиск. Например,

dig -x 8.8.8.8

[](https://legacy.merionet.ru/images/tmp/ad-4nxceodrk-z3qsc1yk4ltlluqi4fdjph1ujyhbkstfek-ddtxygslt2k963-3fpfcuvgzufoewn1gzyemvwqkhqx-hvdubbl-nknktlt5ufialqxneszkqpsk1uhrdjo3wwwqyafzusrjghcvzrgz-uiy28q6xzlbmng-oah-6vaxiyzw7vk7hw.)

В разделе ANSWER SECTIONвы увидите запрашиваемое имя домена.

**Примечание**: частые DNS-запросы влияют на производительность веб-сайта. Уменьшив количество запросов, можно снизить нагрузку на сервер и задержку сети.

**nslookup**

Команда nslookup похожа на команду dig. Основное различие между этими двумя командами заключается в том, что у nslookup есть интерактивный режим. С ее помощью можно диагностировать и отправлять запросы на DNS-серверы. Это помогает в выявлении и устранении неполадок в сети и выполнении задач DNS.

Эта команда доступна в большинстве Unix-подобных операционных системах и системах Windows.

**Синтаксис**

Общий синтаксис команды nslookup выглядит так:

nslookup [domain] [DNS server]

Команда состоит из:

* [domain] – имя домена для поиска. Если имя не указано, запросы можно отправлять к нескольким доменам в интерактивном режиме.
* [DNS server] - DNS-сервер, который используется для поиска. Если сервер не указан, по умолчанию используется DNS-сервер системы.

По умолчанию запрос выполняет поиск записи типа А.

**Пример**

Следующий пример демонстрирует, как производится DNS-поиск домена:

nslookup google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxeu1tnxexjt4jo-xm2xahlac-f0bg9i72o8pm1cnq-rr5krqioemosebrdxbzhakmazg1sci7zu7tbj8lincwsid7ym16ap42xae56t01inkwcxajqhmvsanorxvexzmzqrnenjpavfq4nmlhxeiuq-mzcwnybfvqtvamxyizye0e9g-guvse.)

В результате отображается информация о разрешении DNS для указанного домена.

**netstart**

Команда netstat (статистика сети) — это сетевая утилита, отображающая различные статистические данные сети. Команда предоставляет статистику сетевых портов и отображает доступные порты.

Эта команда – часть пакета net-tools и считается неактуальной. Рекомендуется заменить ее командой ss, которая является частью пакета iproute2. Дополнительные функции команды netstat доступны при ее использовании вместе с командой ip.

**Синтаксис**

Синтаксис команды netstat довольно прост:

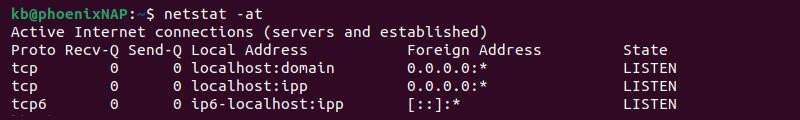
netstat [options]

Команда позволяет комбинировать различные параметры для того, чтобы настроить вывод и отображение сетевой информации. Команда без параметров выводит список открытых сокетов для всех настроенных семейств адресов.

**Пример**

Например, для того, чтобы перечислить все TCP-порты с помощью команды netstat, вам понадобиться параметр -at:

netstat -at

[](https://legacy.merionet.ru/images/tmp/ad-4nxdyey6ldginqkayv-7mpexkcphn8ndjxjhbvb3kryidjgzdig2b8oxzoipmlmy0bittgpag7s4kz9m7cqq0zqppriat88iymk5uip196g7jy4ezawygubgbfaifyk5s6ga9alhctipavkk8tctafvatc95sjsryckxprod1gcvpbucuaxnzzq.)

В результате вы получите список всех активных TCP-соединений в системе.

**traceroute**

Команда traceroute – это инструмент диагностики сети, которые есть в Linux, macOS и Windows. Команда позволяет отслеживать маршрут, по которому пакеты достигают пункта назначения в сети TCP/IP.

Эту команду можно использовать для того, чтобы обнаруживать проблемы маршрутизации и узкие места за счет отображения перемещения пакета в процессе его пути от исходной точки до пункта назначения.

По умолчанию для пакета размером 60 байт (для IPv4) (или 80 байт (для IPv6)) трассировка содержит 30 переходов.

**Синтаксис**

Синтаксис команды traceroute следующий:

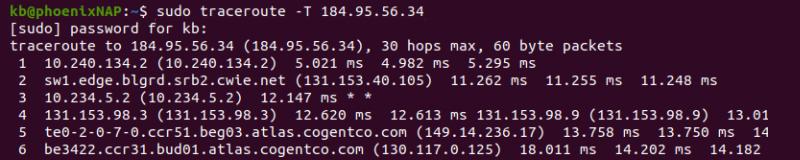
traceroute [options] [hostname/IP]

Параметр [hostname/IP] является обязательным, а вот в качестве дополнительного параметра можно указать DNS-поиск, параметр TTL и тип пакета.

**Пример**

Для того, чтобы выполнить трассировку маршрута пакета по протоколу TCP, запустите от имени администратора команду traceroute с параметром –T. Например,

sudo traceroute -T 184.95.56.34

[](https://legacy.merionet.ru/images/tmp/ad-4nxexnzj3pvrvcf5cazzmmqazgqvdcypv-nvs2va2agemc-9b2i-uwjgbad0zrl3y4ip-99ykkseccylnv1smidrxmvvyclupbmmdntnpcy8r14n17tnqj1mzix-ddae6bcofyteq8gpdb3gpfsas-aiyo8xekjyvjpdfcgh93btzpo2gblsqptq.)

В результате вы получите последовательный маршрут от исходной точки до пункта назначения.

**tracepath**

Команда tracepath похожа на команду traceroute. Команда определяет пути от исходной точки до пункта назначения и их задержки, сопоставляя маршруты и переходы по сети.

И хотя traceroute – хорошо известная команда с большим количеством параметров, команда tracepath – это простой инструмент топологии сети, который есть в большинстве систем Linux.

**Синтаксис**

Синтаксис команды tracepath следующий:

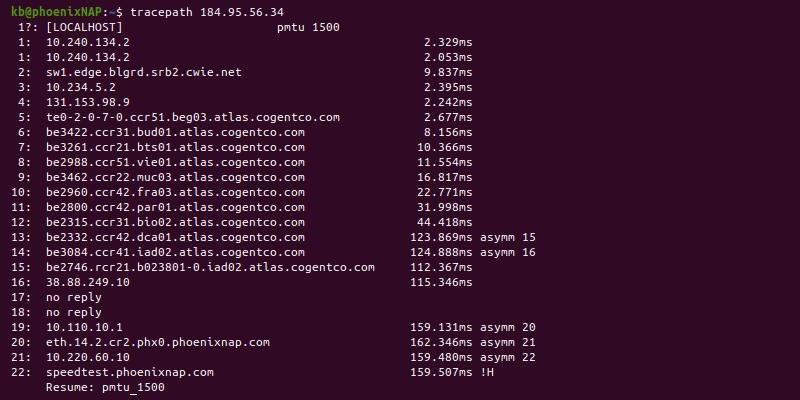
tracepath [options] [hostname/IP]

Дополнительные параметры позволяют управлять поведением запроса, например, количеством переходов и необходимостью выполнения обратного DNS-поиска адресов. Параметр [hostname/IP] является обязательным. Это пункт назначения.

**Пример**

Запустив команду tracepath без каких-либо параметров, вы выполните простую трассировку от пункта назначения до хоста:

tracepath [hostname/IP]

[](https://legacy.merionet.ru/images/tmp/ad-4nxdwc4gzw02prtvq242i-axtvryuvfr7kj0cyqzom5xmenznan91fskxbwifofgir3dtqmrvoag6-ye2pewaxha0frqwidz-pg8iez3j3tsxxvwvhyosksus6ml5sz8a7tg818dqdmlryijaa9qq-zrvtyb5xap6vszlvmpv4kcbwxao6v8fpkk.)

В результате вы получите номера переходов, IP-адреса или разрешенное имя хоста, а также время, затраченное на передачу и подтверждение (RTT - Round-Trip Time) для каждого перехода.

**host**

Команда host – это простой инструмент для выполнения DNS-поиска. Команда преобразует IP-адреса в доменные имена и наоборот.

Эту команду можно использовать для того, чтобы запрашивать DNS-записи, а также выявлять и устранять неполадки, связанные с DNS.

**Синтаксис**

Синтаксис команды host выглядит так:

host [options] [hostname/IP]

Параметры команды помогают управлять ее поведением, например, можно указать тип запроса или приоритетную запись (SOA - Start Of Authority) для предоставленного домена.

**Пример**

Для того, чтобы выполнить простой DNS-поиск, запустите команду host и укажите имя хоста или IP-адрес. Например,

host google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxcie-u3cx72rwxmx9-ypfeuqlk6vt-mpx8xivykxt4-l-vtjvxgsutkvzfxy8kgrjvdjlt4xn5ojv-q398im8mszpnddter0voqcnu8ezutz7wjkrujcihgpevz-omo44a1wbdfu-tsv-8vvranle6ofm6pr2xxbs64eu8qdihjo-vfwpcuza.)

В результате вы увидите разрешенные адреса IPv4 и IPv6 для указанного имени хоста.

**hostname**

Команда hostname отображает и изменяет домен и имя хоста системы, а также определяет устройства в сетевом окружении.

С помощью этой команды можно отображать, изменять и искать имена хостов.

**Синтаксис**

Синтаксис команды hostname следующий:

hostname [options] [name]

Параметр [options] управляет тем, что отображает команда, а параметр [name] временно устанавливает указанное имя в качестве имени хоста.

**Пример**

Для того, чтобы изменить имя хоста системы на какое-то время, запустите команду без параметров и укажите нужное имя:

sudo hostname [name]

У этой команды нет вывода. Чтобы проверить текущее имя хоста, выполните следующую команду:

hostname

[hostname terminal output](https://legacy.merionet.ru/images/tmp/ad-4nxdcxzghlcln-ofmkwcvjw-cn5m9tqzcejomi1cy24r-h66qn9zjvqs8yislbysmrjfitgh1ajfih82vo2siryrydw5wiamxgeda8eoz9iw70jj6cj-tnj3gfi3dfcbiqszwds4tsgjdcw5bufhmqlwmb9kzm6avj4ywvyohhyf5gdiavb4jag.)

На экран выведется текущее имя хоста.

**ping**

Команда ping — это сетевая утилита, позволяющая проверить доступность хоста. Команда отправляет ICMP-запросы на хост (компьютер или сервер) и измеряет RTT.

Эта команда помогает определить задержку сети между двумя узлами и доступность сети.

**Синтаксис**

Синтаксис команды ping следующий:

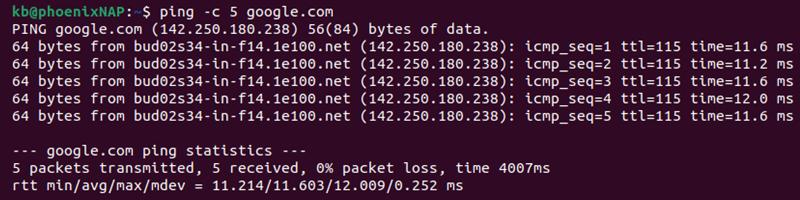
ping [options] [hostname/IP]

Для проверки доступности адресата укажите параметр [hostname/IP]. Также вы можете добавить параметры для управления поведением команды, например, номер ping-запроса, интервалы или размер пакета.

**Пример**

Пример запроса с помощью команды ping выглядит так:

ping -c 5 google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxelgxneamcdm4lh6bvoyccwxtwmpg7m1cwneuijmlotne9orxq3mwbms0mp2mdczgmprdfpslchzvs6ehp9frvd7mng-g2nxdnc5rcb1wsbfy1tpx2x9y8bi7xhcw2tlmshuzhk0akuj5o8spsocr97dht52n4cospmpswo8fsfcle6jzek-o.)

Команда отправляет пять ICMP-пакетов на хост, который вы указали и печатает статистику.

**ss**

Команда ss — это инструмент командной строки, который позволяет отобразить статистику сети. Этот инструмент является частью пакета iproute2. Кроме того, это более быстрая альтернатива команды netstat.

Эту команду можно использовать для проверки сетевых сокетов и просмотра различных сетевых данных.

**Синтаксис**

Базовый синтаксис команды следующий:

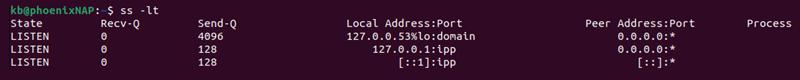
ss [options] [filter]

Параметр [options] позволяет фильтровать сокеты по протоколу, а параметр [filter] помогает ставить сокеты в очередь по состоянию. Это необходимо для того, чтобы сократить количество отображаемых данных.

**Пример**

Например, для того, чтобы отобразить все прослушивающие TCP-сокеты с помощью команды ss, добавьте параметр –lt:

ss -lt

[](https://legacy.merionet.ru/images/tmp/ad-4nxe-lwc0kd-0nyl9gg-q2864ignig90b4j68uesnznyxwt-tazzo-gwjejhz9enztud2b1aqosnihw-ppymqwdzm2x2edshdr-jzhb42uj6g7ikit3tlawipiqbufiogzr5qax3e84m72-umqtzsmalvyoftycndzadkkzeqbqohaslbm5-q5o.)

В результате вы получите список всех TCP-сокетов в состоянии LISTEN, которые находятся в ожидании входящих соединений.

**route**

Команда route в Linux – это специальная команда для отображения и настройки таблицы маршрутизации. Команда позволяет изменять таблицы IP-маршрутизации ядра и помогает настраивать статические маршруты к определенным хостам или сетям.

Эту команду необходимо использовать уже после того, как вы настроите сетевой интерфейс с помощью специального инструмента, например, команды ifconfig.

**Примечание**: предпочтительной альтернативой команде route является команда ip route.

**Синтаксис**

Синтаксис команды route следующий:

route [options] [subcommand] [arguments]

Она содержит:

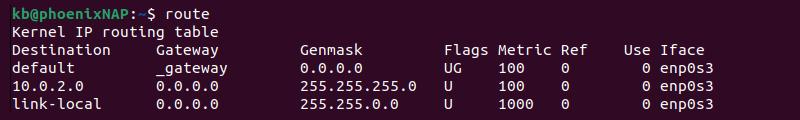
* [options] – дополнительные параметры командной строки, с помощью которых можно управлять следующим: конечным представлением данных, семейством адресов и IP-протоколом
* [subcommand] – действие которое необходимо выполнить, например, add или delete
* [arguments] –

дополнительные аргументы, которые могут отличаться в зависимости от выбранной подкоманды

**Пример**

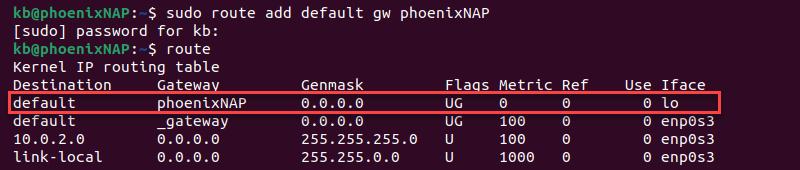
Для того, чтобы просмотреть текущую таблицу маршрутизации, введите команду route без параметров:

route

[](https://legacy.merionet.ru/images/tmp/ad-4nxeyxye-oxjfvady6oyfh4pjnu9f-tsmkpd1ztqukblsh-irirj1agj4-fyeodpqw8lvt8dar-k3qwyvet7c92ost6cn-1myon1bjgmjxk30oxhytgiw6emt3qknddoepsn-h-k1pxsz5anibezic5-8wa22v3eun94itl7jp4gnvvqi2h9s3ta.)

Чтобы добавить шлюз, используемый по умолчанию, выполните следующую команду:

sudo route add default gw [gateway]

[](https://legacy.merionet.ru/images/tmp/ad-4nxcifulzvbr8r5rnlxrpw5jjt2tmeettvv-mdjbdg9bqzcpb99x3naekjvejfz2nxwv3si-fehhwqq914-m6cvtisn26mxl-s-rpd0tvxhn-txfkzdlv8hdnmu7wzbfww0kj-tcv80ic02f3ba5glxizu27zhgqusuvptxgre4zom6xhprsbxu4.)

Команда добавляет маршрут по умолчанию, который будет использоваться, когда остальные маршруты не совпадают. Указанный шлюз должен являться маршрутом, доступным напрямую.

**arp**

Команда arp отображает и настраивает кэш протокола ARP (Address Resolution Protocol – протокол разрешения адресов). Протокол ARP сопоставляет IP-адреса с физическими MAC-адресами (Media Access Control – управление доступом к среде передачи данных) в локальной сети. Эта схема, составленная для всех устройств в локальной сети, хранится в кэше.

**Синтаксис**

Синтаксис команды arp выглядит так:

arp [options] [hostname/IP]

* Параметр [options] позволяет изменять поведение команды, например, настраивать и удалять действия или управлять выводом.
* Параметр [hostname/IP] – это необязательный идентификатор удаленной системы, для которой необходимо разрешить MAC-адрес. В случае, если этот параметр не указан, команда проверяет локальный кэш ARP.

**Пример**

Для того, чтобы отобразить кэш ARP, запустите команду arp без дополнительных параметров:

arp

[arp terminal output](https://legacy.merionet.ru/images/tmp/ad-4nxeiu7brydsbrnwqkahid6q0xya5c6zcmxoj400vnss6b3nqdcowcu34ex9z10kgdx4bxel1mbshljcclv-wplk9hkyxxka2ioplirligvieam-ngkhmh6-bcxj6zf8pfuot9nykhy-mow3wuujtcgyfcrz7via6ywxxtnw85-dks-ngq0afaa.)

В результате вы увидите таблицу кэша ARP (IP и MAC-адреса).

**iwconfig**

Команда iwconfig отображает и настраивает информацию об интерфейсе беспроводной сети. Команда может оказаться полезной при выявлении и устранении проблем с беспроводной сетью.

Эту команду можно использовать для просмотра или изменения имени беспроводной сети, настроек управления электропитанием и прочих параметров беспроводной сети.

**Синтаксис**

Синтаксис команды iwconfig выглядит так:

iwconfig [interface] [options]

Параметр [interface] фильтрует интерфейсы беспроводной сети по имени, а параметр [options] отвечает за различные настройки, например, режим работы, ограничения скорости и ключ шифрования беспроводной сети.

**Пример**

Для того, чтобы просмотреть доступные беспроводные интерфейсы в системе и текущие настройки, запустите команду iwconfig без параметров:

iwconfig

[](https://legacy.merionet.ru/images/tmp/ad-4nxf9ujkeenvyaz7ggkodufpp4aocxwydwl9g2bjrxi81nf6qhih-yuxi7u2latx1y7onx0kgfmj-unbunmtaf8mmbsfcl4pkoicmucezaj8-wny2dzb2tmxbmb5aqgjlrrs8tcf4mxdujsr5lswg7qeyyo0uvpjuwi6g1olf6jopeelzp8caurk.)

Команда покажет всю информацию, касающуюся беспроводных интерфейсов системы.

**curl** **или** **wget**

Команды wget и curl — это инструменты командной строки, предназначенные для загрузки файлов из Интернета. Они похожи, но все же есть небольшие различия в том, как они работают, и в их параметрах:

* Команда wget загружает файлы из Интернета по протоколам HTTP, HTTPS или FTP. Этот инструмент довольно прост с точки зрения загрузки файлов.
* Команда curl является универсальной и поддерживает различные сетевые протоколы, такие как SCP, IMAP, POP3, SMTP и т. д. С помощью этого инструмента также можно отправлять HTTP-запросы и взаимодействовать с веб-службами.

Эти команды можно использовать для проверки скорости загрузки по сети.

**Синтаксис**

Синтаксис для этих двух команд одинаковый:

wget [options] [URL]

curl [options] [URL]

Параметр [options] управляет различными параметрами загрузки и вывода, а параметр [URL] – это URL-адрес, откуда необходимо загрузить файл. В отличие от команды wget, у команды curl есть множество различных параметров и шаблонов.

**Пример**

Для того, чтобы загрузить файл с помощью команды wget, запустите следующую команду:

wget -O [file name] [URL]

А для того, чтобы выполнить ту же задачу, но с помощью команды curl, вот эту:

curl -o [file name] [URL]

Файл загружается с указанного URL-адреса, а его содержимое сохраняется в указанном файле.

**mtr**

Команда mtr (**m**y **tr**aceroute) – это инструмент диагностики, который сочетает в себе элементы команд ping и traceroute. Команда отправляет аналитическую информацию о качестве сети в режиме реального времени. Это делает ее отличным инструментом для выявления и устранения неполадок, связанных с большой задержкой и потерей пакетов.

**Синтаксис**

Синтаксис команды mtr следующий:

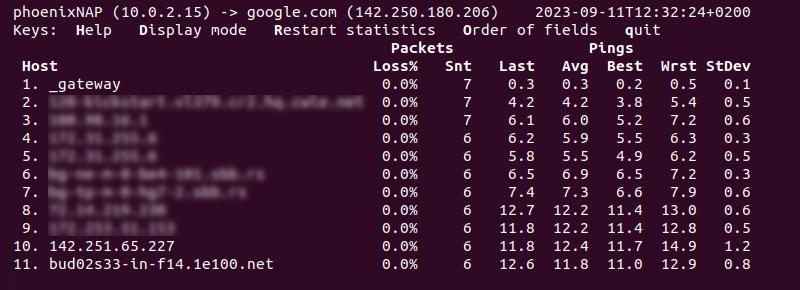
mtr [options] [hostname/IP]

Параметр [options] позволяет управлять количеством и размером пакетов, а параметр [hostname/IP] - это пункт назначения.

**Пример**

Команда mtr без параметров запускает сеанс трассировки до указанного хоста. Например,

mtr google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxeiqqhwpd2pdtbr-zymjw8lhorscbcla-irf9lix1t5oxuo6olfwjqlzs-0m0s1inc-lstnw8qdzhrl3sxrmdryqkomnh8i1tbqkz0eret0u-rxinzccupkvzlw7xk-auxiostxxn9hlgirirfng6ytpk4zpnewrlecwutzligitdswtw6fea.)

Чтобы выйти, нажмите **q**.

**whois**

Команда whois запрашивает информацию о доменных именах, IP-адресах и прочие сетевые данные. Эту команду можно использовать для получения сведений о владельце домена, например, дату регистрации и дату истечения срока действия.

**Синтаксис**

Синтаксис команды следующий:

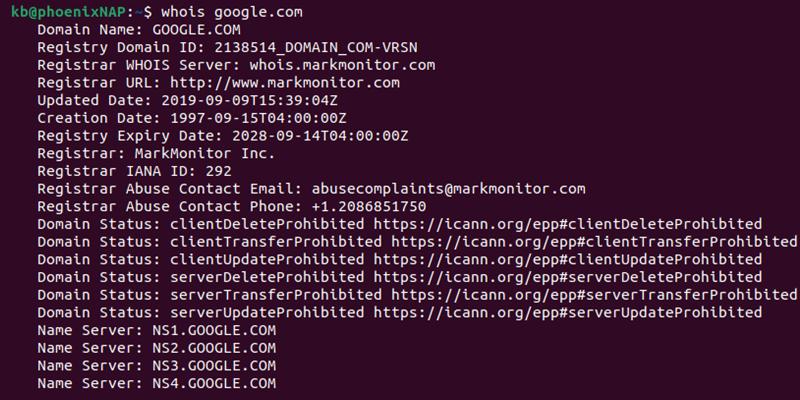
whois [options] [query]

* Параметр [options] позволяет установить конкретный сервер WHOIS для запроса, изменить протокол и добавить дополнительные параметры запроса.
* Параметр [query] – это имя домена, IP-адрес или номер в автономной системе (ASN - Autonomous System Number) для поиска.

**Пример**

Чтобы выполнить простой запрос для определённого доменного имени, необходимо запустить команду без параметров. Например,

whois google.com

[](https://legacy.merionet.ru/images/tmp/ad-4nxdlo9c4-gl4mf0tkxrhgwrwd8s5wmxc8uxdnmlnihdw99-6kewt60abrhtilbous6u6isptniowkjvasvsl2pdkbmqn5ofujgi4rkvu4uzgps1fvywlm25id0ckblzvfv2znxotn-m6gevqmyh98j13w-ukximbizfvxaoyrka5ucoxyxrbmte.)

В итоге вы получите результаты поиска WHOIS для указанного доменного имени.

**iftop**

Команда iftop — это утилита для мониторинга сети. Эту команду можно использовать для просмотра сетевых подключений и загрузки сети в режиме реального времени.

**Синтаксис**

Синтаксис команды iftop следующий:

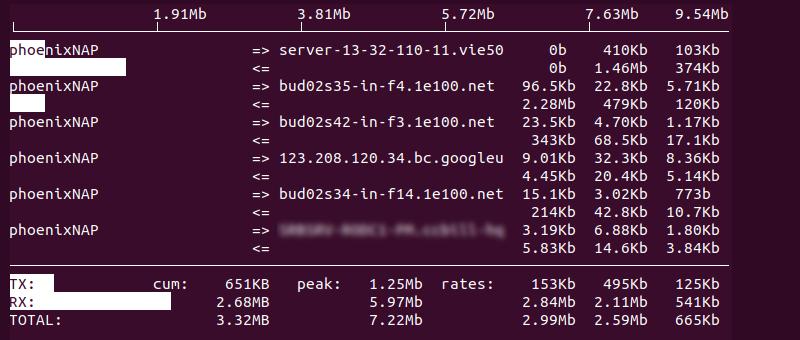
iftop [options]

Параметр [options] управляет отображаемой информацией. Кроме того, для мониторинга всего трафика сетевого интерфейса команда требует соответствующих прав доступа.

**Пример**

Как правило, эту команду используют без каких-либо параметров:

sudo iftop

[](https://legacy.merionet.ru/images/tmp/ad-4nxcp75q0y7ivmixwhwjmvrrrq8oovhnawcnrrrlagdjajendi-7r87kzy0xvn8tglgrztdteqly7wdxj9rzu7c6zc2ji3f3ucipggjzmr6-bfpionsnrerxs-amss9g-xkia6yj7gk8mfiwxgdipdw-j5fssaz2-xirurm4oizibjfnsjcy6p0g.)

Команда открывает экран мониторинга, где в режиме реального времени отображается процесс передачи данных через сетевой интерфейс.

С помощью интерфейса можно управлять отображением данных на этом экране, например, переключаться между отображениями исходной точки (**s**) и пункта назначения (**d**). Чтобы выйти, нажмите **q**.

**tcpdump**

Команда tcpdump — это перехватчик пакетов и инструмент сетевой безопасности. Он собирает информацию о сетевых пакетах, поступающую в режиме реального времени. Эту команду можно использовать для анализа трафика, выявления и устранения неполадок, а также мониторинга безопасности сети.

**Синтаксис**

Синтаксис команды tcpdump следующий:

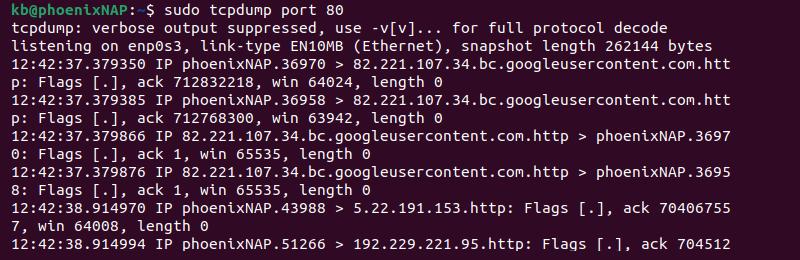
tcpdump [options] [filter]

Параметр [options] позволяет управлять отображением данных и количеством пакетов, а также работать с файлами. А для ввода критериев захвата пакетов вам понадобиться параметр [filter].

**Пример**

Для того, чтобы выполнить захват пакетов на каком-то определенном порту, выполните следующую команду:

sudo tcpdump port 80

[](https://legacy.merionet.ru/images/tmp/ad-4nxcodpdlzsccuehpprrrsxxgb45-rr11c-xg8nba-jg4x7n9arrpxzsofn7niozpw2qmdtsvaylpmfevjokatoinfmpdjpsstdbbafwi5u0gg-j3jechsi9gx-zmpjv6fwyqrd88bivhbqmiet8fuc6t7rvpus9-08cixnv8t8o4hzgjtwb61g.)

Теперь фильтр порта 80 захватывает на нем пакеты для мониторинга HTTP-трафика.

**ifplugstatus**

Команда ifplugstatus — это простая утилита, предназначенная для проверки состояния сетевого интерфейса. С помощью этой команды можно определить, подключен ли Ethernet-кабель к интерфейсу.

Эта команда используется для проверки физического соединения сети, особенно после того, как в сетевой интерфейс были внесены изменения.

**Синтаксис**

Синтаксис команды ifplugstatus следующий:

ifplugstatus [options] [interface]

Параметр [options] позволяет установить конкретный файл конфигурации или запустить сценарии в пакетном режиме. Указав параметр [interface], вы можете проверить состояние определенного интерфейса.

**Пример**

Для того, чтобы просмотреть состояния всех сетевых интерфейсов, запустите команду без параметров:

ifplugstatus

[ifplugstatus terminal output](https://legacy.merionet.ru/images/tmp/ad-4nxf1lobkjqgivkt09fv10s-vije9fbdb8a60zma-jw4ee05g6ubo5pyy3ldy4y9s73ad7hzeztaiayh3dbhopuq8utkm7ynto4-rgwu6b6hdeawundqu0cju4g87shopnar715z99m5fzmx2c-mjeunnygarxzrphhyegvrcb8qxetgt-dnfsao.)

Если в результате вы видите сообщение «link beat detected», это значит, что у интерфейса есть активное физическое соединение.