

Настройка сети вручную

Содержание

- [Настройка сети вручную](#)
 - [Краткое описание руководства](#)
 - [Требования к системе](#)
 - [Настройка проводной сети](#)
 - [Настройка IP-адреса, шлюза по умолчанию, маски подсети](#)
 - [Временная настройка IP-адреса и маски подсети](#)
 - [Настройка DNS](#)
 - [Настройка соединений rrr](#)
 - [Соединение PPPoE](#)
 - [Соединение PPTP](#)
 - [Настройка DIAL-UP подключения](#)
 - [Автоматическое подключение](#)
 - [Ручная настройка роутинга](#)
 - [Установка MTU и TTL](#)
 - [Настройка WiFi](#)
 - [Настройка Wi-Fi при помощи wpa-suplicant и /etc/network/interfaces](#)
 - [Другие способы работы Wi-Fi оборудования](#)
 - [Решение проблем](#)
 - [Перезапуск сети](#)
 - [FAQ по сетям](#)
 - [Ссылки](#)

Краткое описание руководства

В данном руководстве объясняется, как подключить компьютер к сети при помощи конфигурационных файлов и консольных утилит. Основная цель - рассказать о различных способах подключения к интернету без использования GUI (графического интерфейса). Руководство не затрагивает таких тем, как настройка сетевых фильтров или, например, собственных точек доступа Wi-Fi. Предполагается, что существует некий, предоставленный провайдером, способ подключения к интернету, для использования которого и необходимо выполнить приведенные ниже действия.

В руководстве приведены примеры редактирования конфигурационных файлов с помощью текстовых редакторов «nano» и «gedit». Обратите внимание на то, что первый редактор запускается в терминале и может быть использован как при запуске Ubuntu с графическим интерфейсом, так и без него, а «gedit» можно использовать только при включенной графической среде.

Требования к системе

Для воспроизведения описанных в руководстве действий подходит любой вариант установки системы. Наличие графического пользовательского интерфейса не обязательно. Все действия необходимо выполнять в [консоли](#). Предполагается, что команды, начинающиеся с символа \$ - необходимо выполнять от пользователя, а начинающиеся с # - от суперпользователя (root).

Прежде чем Вы начнете, убедитесь, что:

- Различные сетевые утилиты, предназначенные для автоматического конфигурирования сети выключены. Например, [tut](#) Вы можете прочитать,

как отключить установленный по умолчанию в Ubuntu сетевой помощник Network Manager.

- Различные сетевые фильтры (например iptables), и утилиты их конфигурирования (например, Firestarter) отключены/правильно настроены и не вмешиваются в работу сети.
- У Вас есть все необходимые параметры для подключения в Вашей сети (например, IP-адрес, маска подсети и шлюз по умолчанию для соединения с использованием статического IP).
- Устройства сети осуществляющие фильтрацию по MAC-адресу правильно настроены и «знают» Ваш сетевой интерфейс.
- Драйвер Вашего сетевого устройства корректно установлен, кабель (при проводном соединении) исправен и подсоединен. Команда

```
$ sudo lshw -C network
```

позволяет посмотреть подключенные сетевые устройства.
Пример вывода команды:

```
ubuntu@ubuntu:~$ sudo lshw -C network
```

```
*-network
```

```
description: Ethernet interface # Тип устройства
```

```
product: L2 100 Mbit Ethernet Adapter # Название адаптера
```

```
vendor: Attansic Technology Corp. # Производитель устройства
```

```
physical id: 0
```

```
bus info: pci@0000:03:00.0
```

```
logical name: eth0 # Имя сетевого интерфейса
```

```
version: a0
```

```
serial: 00:00:00:00:00:00 # Физический адрес устройства (mac-адрес)
```

```
size: 100MB/s
```

```
capacity: 100MB/s
```

```
width: 64 bits
```

```
clock: 33MHz
```

```
capabilities: pm msi pciexpress vpd bus_master cap_list
```

```
ethernet physical tp 10bt 10bt-fd 100bt 100bt-fd autonegotiation
```

```
configuration: autonegotiation=on broadcast=yes

driver=atl2 # Используемый драйвер

driverversion=2.2.3 # Версия драйвера

duplex=full firmware=L2 ip=192.168.0.5 latency=0

link=yes # Наличие линка

module=atl2 multicast=yes port=twisted pair

speed=100MB/s # Текущая скорость подключения.
```

Обратите внимание на пункт:

```
logical name: eth0 # Имя сетевого интерфейса
```

Имя `eth0` будет далее применяться для настройки именно данной сетевой карты. Где `eth` обозначает что используется [Ethernet](#) интерфейс, а `0` - номер устройства. Если у вас установлено несколько сетевых устройств, то, соответственно, им будут присвоены имена: `eth0`, `eth1`, `eth2` и т.д.

Настройка проводной сети

Настройка IP-адреса, шлюза по умолчанию, маски подсети

Отредактируйте файл конфигурации `/etc/network/interfaces`, например так:

```
$ sudo nano /etc/network/interfaces
```

И _____ напишите _____ в _____ него:
Для статического IP:

```
iface eth0 inet static

address 192.168.0.1

netmask 255.255.255.0

gateway 192.168.0.254

auto eth0
```

Где:

- `iface eth0 inet static` - указывает, что интерфейс (`iface eth0`) находится в диапазоне адресов [IPv4](#) (`inet`) со статическим `ip (static)`;
- `address 192.168.0.1` - указывает что IP адрес (`address`) нашей сетевой карты `192.168.0.1`;

- `netmask 255.255.255.0` - указывает что наша маска подсети (`netmask`) имеет значение `255.255.255.0`;
- `gateway 192.168.0.254` - адрес шлюза (`gateway`) по умолчанию `192.168.0.254`;
- `auto eth0` - указывает системе что интерфейс `eth0` необходимо включать автоматически при загрузке системы с вышеуказанными параметрами.

eth0 - имя подключаемого своего интерфейса. Список интерфейсов можно посмотреть набрав:

```
$ ifconfig -a
```

В итоге файл `/etc/network/interfaces` должен выглядеть примерно так: (для одного проводного соединения со статическим IP)

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# Моя проводная сеть.
iface eth0 inet static
address 192.168.0.1
netmask 255.255.255.0
gateway 192.168.0.254
auto eth0
```

Сохраните файл и закройте редактор. В данном примере (редактор `nano`) - нажмите `Ctrl+X`, затем `Y`, убедитесь, что «Имя файла для записи» - `/etc/network/interfaces` и нажмите `Enter`.

Более подробно про синтаксис файла `/etc/network/interfaces` можно прочитать в [документации](#).

Пример конфигурации для динамического IP:

```
iface eth0 inet dhcp
```

```
auto eth0
```

Временная настройка IP-адреса и маски подсети

При необходимости задать пробные настройки, выполните:

```
$ sudo ifconfig eth0 192.168.0.1 netmask 255.255.255.0 up
```

Где 192.168.0.1 - наш IP-адрес, 255.255.255.0 - наша маска подсети.
eth0 - подключаемый сетевой интерфейс.

Данные настройки пропадут после перезагрузки системы и не повлияют на файл */etc/network/interfaces*

Настройка DNS

За конфигурацию DNS отвечает утилита `resolvconf`, которая работает в паре с небольшим кеширующим DNS сервером `dnsmasq`. `resolvconf` позволяет осуществить настройку DNS на основе данных от разных подсистем. Одним из следствий этого полезного нововведения (переход на эту схему произошел в Ubuntu начиная с версии 12.04) является то, что теперь файл */etc/resolv.conf* генерируется автоматически, и не индивидуально каждой программой, которая хочет исправить (например Network Manager или DHCP клиент), а через общий программный интерфейс. Это значит, что внесенные «руками» изменения в */etc/resolv.conf* будут потеряны. Автоматически формируемый */etc/resolv.conf* содержит ссылку на DNS сервер на локальном интерфейсе (127.0.1.1), а там (на 53 порту) и сидит сервис `dnsmasq`, который отвечает за разрешение символьных имен в IP адреса. Нужно отметить, что этот порт (53) открыт в режиме LISTEN, но т.к. это локальный интерфейс, то из внешней сети этот порт не доступен. Информацию о DNS для статических интерфейсов теперь надо вносить в */etc/network/interfaces* в параметры `dns-nameservers`, `dns-search` и `dns-domain` (которые соответствуют параметрам `nameserver`, `search` и `domain` в */etc/resolv.conf*) Обратите внимание - в */etc/resolv.conf*, при записи нескольких серверов используется несколько ключей `nameserver`, а в */etc/network/interfaces* все адреса DNS серверов записываются в одну строку после ключа `dns-nameservers`, разделенные пробелами.

В итоге описание статического интерфейса в */etc/network/interfaces* должно выглядеть примерно так:

```
iface eth0 inet static

address 192.168.0.1

netmask 255.255.255.0

gateway 192.168.0.254
```

```
dns-nameservers 8.8.8.8 192.168.0.254
```

```
auto eth0
```

Ubuntu до версии 12.04

Если в более старых версиях ubuntu есть необходимость указать DNS сервера (если они не выдаются автоматически) выполните:

```
$ sudo gedit /etc/resolv.conf
```

и впишите туда:

```
nameserver 192.168.0.100
```

```
nameserver 192.168.0.200
```

Где 192.168.0.100 и 192.168.0.200 - адреса DNS серверов. Если нужно добавить больше адресов - каждый адрес нужно начинать с новой строки и с фразы `nameserver`

Настройка соединений ppp

За создание соединений типа «точка-точка» в Ubuntu отвечает демон *pppd*, более подробная информация о котором доступна в [документации](#). В рамках данного руководства будут рассмотрены примеры создания PPPoE подключения через DSL модем, подключения PPTP (VPN-подключения) и DIAL-UP подключения через обычный модем.

Соединение PPPoE

В стандартную установку Ubuntu входит утилита для настройки PPPoE соединений - *pppoeconf*, для ее запуска наберите:

```
$ sudo pppoeconf
```

Появится «псевдографическое» ²⁾ окно в терминале. Утилита произведет поиск сетевых устройств и выведет их на экран, далее она произведет поиск модема ³⁾ на этих устройствах. Если на этом этапе *pppoeconf* выдаст отрицательный результат - проверьте правильность подключения, питание модема. Следующий шаг - выбор «популярных параметров» - в большинстве случаев стоит согласиться. Далее утилита запросит Ваш логин, а затем - пароль. Теперь - выбор способа указания DNS серверов. Опять же, в большинстве случаев следует согласиться на получение адресов DNS серверов автоматически. Далее Вам предложат ограничить размер **MSS** до 1452-х байт - как правило, стоит согласиться. Следующий вопрос - устанавливать ли подключение автоматически при загрузке компьютера. Последний вопрос утилиты - установить ли соединение сейчас. *pppoeconf* по умолчанию создает для подключения имя *dsl-provider*. Управлять подключением Вы можете при помощи команд:

```
$ sudo pon dsl-provider # Для подключения
```

или

```
$ sudo poff dsl-provider # Для отключения
```

Если в Вашем случае опций, предоставляемых утилитой *pppoeconf* недостаточно - обратитесь к [документации по pppd](#) или [pppoeconf](#).

Замечание: при настройке соединения с помощью *pppoeconf* часть настроек записывается в `/etc/network/interfaces`, в результате чего Network Manager больше не может управлять сетью. Выход: либо использовать только NM, либо только консоль+конфиги. Вернуть управление Network Manager можно следующим образом. Приведите `/etc/network/interfaces` к следующему виду (лишнее не обязательно удалять, достаточно закомментировать):

```
# This file describes the network interfaces available on your system

# and how to activate them. For more information, see interfaces(5).

# The loopback network interface

auto lo

iface lo inet loopback
```

Перезапустите сеть:

```
$ sudo /etc/init.d/networking restart
```

Перезагрузитесь или перезапустите Network Manager:

```
$ sudo /etc/init.d/NetworkManager restart
```

Соединение PPTP

Для осуществления VPN-соединения при помощи *pppd* Вам потребуется [установить](#) пакет *pptp-linux*, который может быть найден на установочном диске Ubuntu. Далее создайте (от root'a) в папке `/etc/ppp/peers` файл с названием Вашего провайдера и отредактируйте его, например так:

```
$ sudo nano /etc/ppp/peers/my-provider
```

И добавьте туда опции подключения, например такие:

```
persist # При разрыве соединения - переподключаться снова.
```

```
maxfail 0 # Максимальное количество неудачных попыток подключения. 0 -
Бесконечно.

mtu 1476 # Значение MTU

name {логин} # Ваш логин.

#nodefaultroute # Не быть шлюзом по умолчанию

defaultroute # Быть шлюзом по умолчанию

replacedefaultroute # Заменить шлюз по умолчанию если он был

remotename {vpn} # Имя удаленного сервера (для нас), может быть любым.

pty "pptp {адрес_сервера} --nolaunchpppd" # Команда запуска pptp.

# Адрес сервера - может быть как IP адресом, так и доменным именем, например
vpn.foo.bar
```

Далее - отредактируйте файл `/etc/chap-secrets` [4\)](#) и добавьте туда:

```
{логин} {vpn} {пароль}
```

После перезагрузки системы Вы сможете управлять соединением при помощи команд:

```
$ sudo pon my-provider # Для подключения

или

$ sudo poff my-provider # Для отключения
```

Процесс настройки VPN-соединения может сильно облегчить [скрипт-помощник](#).

Настройка DIAL-UP подключения

Для настройки модемного соединения можно использовать встроенный конфигуратор `pppd` - `pppconfig` или специальную утилиту `wvdial`.

При помощи rrrconfig

Процесс настройки при помощи `pppconfig` во многом похож на утилиту `pppoeconfig`, Вам по очереди будут заданы вопросы о параметрах подключения, и будет предложено ввести номер телефона, логин и пароль, а также имя соединения. Запускать `rrpconfig` следует с правами суперпользователя. Например так:

```
$ sudo pppconfig
```

Управлять соединением можно так:


```
$ sudo pon my-provider # Для подключения
```

или

```
$ sudo poff my-provider # Для отключения
```

Где my-provider - имя, присвоенное Вами соединению при настройке.

При помощи wvdial

В некоторых случаях (например при подключении с использованием мобильного телефона), удобнее использовать *wvdial*. Для этого нужно его сначала установить. Например так:

```
$ sudo apt-get install wvdial
```

В состав пакета *wvdial* входит утилита автоматического конфигурирования - *wvdialconf*.

```
$ sudo wvdialconf
```

Вывод будет примерно следующим:

```
ubuntu@ubuntu:~$ sudo wvdialconf
```

```
[sudo] password for ubuntu:
```

```
Editing `/etc/wvdial.conf'.
```

```
Scanning your serial ports for a modem.
```

```
Modem Port Scan<*1>: S0    S1    S2    S3
```

```
WvModem<*1>: Cannot get information for serial port.
```

```
ttyACM0<*1>: ATQ0 V1 E1 -- OK
```

```
ttyACM0<*1>: ATQ0 V1 E1 Z -- OK
```

```
ttyACM0<*1>: ATQ0 V1 E1 S0=0 -- OK
```

```
ttyACM0<*1>: ATQ0 V1 E1 S0=0 &C1 -- OK
```

```
ttyACM0<*1>: ATQ0 V1 E1 S0=0 &C1 &D2 -- OK
```

```
ttyACM0<*1>: ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0 -- OK
```

```
ttyACM0<*1>: Modem Identifier: ATI -- Manufacturer: QUALCOMM INCORPORATED
```

```
ttyACM0<*1>: Speed 4800: AT -- OK
ttyACM0<*1>: Speed 9600: AT -- OK
ttyACM0<*1>: Speed 19200: AT -- OK
ttyACM0<*1>: Speed 38400: AT -- OK
ttyACM0<*1>: Speed 57600: AT -- OK
ttyACM0<*1>: Speed 115200: AT -- OK
ttyACM0<*1>: Speed 230400: AT -- OK
ttyACM0<*1>: Speed 460800: AT -- OK
ttyACM0<*1>: Max speed is 460800; that should be safe.
ttyACM0<*1>: ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0 -- OK
```

Found an USB modem on /dev/ttyACM0.

Modem configuration written to /etc/wvdial.conf.

```
ttyACM0<Info>: Speed 460800; init "ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0"
```

Теперь остается только отредактировать файл `/etc/wvdial.conf` и добавить в него номер телефона, логин и пароль.

```
$ sudo nano /etc/wvdial.conf
```

В данном примере я дополнительно добавил несколько опций. См. комментарии.

```
[Dialer Defaults]

Init1 = ATZ

Init2 = ATQ0 V1 E1 S0=0 &C1 &D2 +FCLASS=0

Modem Type = USB Modem

ISDN = 0

Idle Seconds = 0

New PPPD = yes

Dial Attempts = 0

Phone = #777
```

```
Modem = /dev/ttyACM0

Username = mobile

Password = internet

Baud = 460800

Idle Seconds = 0 # Время простоя соединения,

# после которого нужно будет его разорвать. Значение 0 - никогда.

Dial Attempts = 0 # Количество попыток дозвона. 0 - бесконечно.

[Dialer pulse]

Dial Command = ATDP # Команда набора номера (P - импульсный, T - тональный).
Имеет смысл для набора номера в импульсном режиме на старых АТС.
```

Файл `/etc/wvdial.conf` разбит на секции, в качестве разделителей которых выступают сами названия секций, предварённые словом `Dialer`, в квадратных скобках. Если исполнять команду без параметров, то в дело пойдут установки, перечисленные в секции `Defaults`. В противном случае дополнительно будут исполнены указанные в добавочных секциях команды.

Теперь, когда все настроено, соединение можно установить набрав:

```
$ sudo wvdial
```

Если потребуется запустить `wvdial` с набором номера в импульсном режиме, то это можно сделать командой

```
$ sudo wvdial pulse
```

Прервать соединение можно прервав выполнение команды `wvdial`, т.е. в том же терминале нужно нажать `Ctrl+C`.

Автоматическое подключение

Отредактируйте файл конфигурации `/etc/network/interfaces`, например так:

```
$ sudo nano /etc/network/interfaces
```

И допишите в него:
Для `pppoe`, `pptp`, и модемного подключения без использования `wvdial`:

```
iface ppp0 inet ppp

provider my-provider
```

```
auto ppp0
```

Где **my-provider** - название вашего соединения.
При использовании *wvdial*:

```
iface ppp0 inet wvdial  
  
provider wvdial  
  
auto ppp0
```

Теперь при перезапуске сетевых служб соединение будет автоматически установлено.

Ручная настройка роутинга

Если Вы не получаете адрес шлюза по-умолчанию от сервера, к которому подключаетесь, или по какой-либо иной причине Вам необходимо указать маршруты вручную - Вы можете создать свой скрипт в */etc/ppp/ip-up.d/*, либо по рекомендации официальной документации создать */etc/ppp/ip-up.local* например так:

```
$ sudo nano /etc/ppp/ip-up.local
```

или

```
$ sudo nano /etc/ppp/ip-up.d/routing
```

со следующим кодом:

```
#!/bin/sh  
  
#  
  
route del default  
  
route add default ppp0 # Имя ppp-подключения.  
  
# тут - необходимые маршруты, например:  
  
route add -net 192.168.0.0 netmask 255.255.255.0 gw 192.168.0.1 dev eth0
```

Далее - сделайте этот скрипт исполняемым, например так:

```
$ sudo chmod ug+x /etc/ppp/ip-up.local
```

или

```
$ sudo chmod ug+x /etc/ppp/ip-up.d/routing
```

Теперь маршруты будут автоматически подключаться при установлении ррр-соединения.

Установка MTU и TTL

MTU (Maximum Transfer Unit) - параметр определяет величину максимальной единицы передачи. Это максимальное количество октетов (байт), которое интерфейс способен поддерживать за одну операцию приема/передачи. Для Ethernet это значение по умолчанию составляет 1500 (максимальный размер пакета Ethernet).

TTL (Time To Live) - время жизни ip-пакета в секундах. Нужен чтобы избежать перегрузки сети пакетами. Обычно каждый роутер, через которого прошел пакет, уменьшает TTL на единицу. Если TTL=0, пакет из системы удаляется. Изначально TTL=128 (для Windows) и TTL=64 (для Ubuntu). Для DNS-записей TTL определяет время актуальности данных при кешировании запросов.

Для изменения величины MTU, отредактируем файл конфигурации `/etc/network/interfaces`, например так:

```
auto eth0

iface eth0 inet static

address 192.168.1.5

netmask 255.255.255.0

mtu 600
```

Для изменения величины TTL наберите:

```
$ sudo su

затем

# echo "128" > /proc/sys/net/ipv4/ip_default_ttl
```

Значение TTL меняется только с правами администратора, для выхода из аккаунта администратора введите `exit`

Настройка WiFi

Настройка Wi-Fi при помощи wpa-supplіcant и /etc/network/interfaces

В этой главе пойдет речь о настройке подключения к существующей Wi-Fi сети с использованием наиболее безопасного на сегодняшний день стандарта

шифрования и аутентификации WPA2. Дополнительно приведены примеры настроек для менее безопасных соединений.

Если Вы можете повлиять на настройку точки доступа, например, в случае, если это Ваш домашний Wi-Fi роутер - постарайтесь настроить авторизацию с использованием WPA2, т.к. это наиболее безопасный протокол аутентификации в беспроводных сетях на настоящий момент.

Замечания

Дополнительно к [требованиям, указанным выше](#) Вам следует учесть, что:

1. После установки всех необходимых пакетов, кабель проводной сети лучше отсоединить.
2. Некоторые устройства (или их драйвера, например Madwifi) не поддерживают WPA2 (AES). Если соединение WPA2 установить не удастся, можно попробовать WPA1 (TKIP).
3. Если у Вас RTxxx (Ralink) с драйверами Serialmonkey - этот способ Вам не поможет. Вам следует либо установить пакет *ndiswrapper*, заменяющий Serialmonkey, либо попробовать другой способ.

Подготовка

Установите пакеты *wpa-supPLICant* и *wireless-tools*
Например так:

```
$ sudo apt-get install wpasupplicant wireless-tools
```

Теперь убедитесь в том, что Ваше беспроводное устройство (в данном примере это «wlan0») работает и «видит» беспроводные сети. Команда

```
$ iwconfig
```

должна выдать примерно такой результат:

```
ubuntu@ubuntu:~$ iwconfig

lo          no wireless extensions.

eth0        no wireless extensions.

wmaster0    no wireless extensions.

wlan0       IEEE 802.11bg  ESSID:""
```

```
Mode:Managed  Frequency:2.412 GHz  Access Point: Not-Associated  
  
Tx-Power=27 dBm  
  
Retry min limit:7  RTS thr:off  Fragment thr=2352 B  
  
Power Management:off  
  
Link Quality:0  Signal level:0  Noise level:0  
  
Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0  
  
Tx excessive retries:0  Invalid misc:0  Missed beacon:0
```

а доступные сети можно посмотреть командой

```
$ iwlist <имя интерфейса> scan
```

которая должна выдать примерно такой результат:

```
ubuntu@ubuntu:~$ iwlist wlan0 scan  
  
wlan0      Scan completed :  
  
Cell 01 - Address: 00:00:00:00:00:00  
  
ESSID:"ubuntuessid"  
  
Mode:Master  
  
Channel:8  
  
Frequency:2.447 GHz (Channel 8)  
  
Quality=7/100  Signal level:-99 dBm  Noise level=-104 dBm  
  
Encryption key:on  
  
Bit Rates:1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 18 Mb/s  
           24 Mb/s; 36 Mb/s; 54 Mb/s; 6 Mb/s; 9 Mb/s  
           12 Mb/s; 48 Mb/s  
  
Extra:tsf=00000001d57e98bf6  
  
Extra: Last beacon: 388ms ago
```

Бывает, что Ваше устройство Wi-Fi не включено по умолчанию, тогда команда вместо рапорта, приведенного выше выдаст такое сообщение:

```
wlan0    Failed to read scan data : Network is down
```

Ничего страшного, просто введите команду

```
sudo ifconfig wlan0 up
```

соответственно выключить устройство можно командой

```
sudo ifconfig wlan0 down
```

Настройка

Редактируем /etc/network/interfaces, например так:

```
$ sudo gedit /etc/network/interfaces
```

Удаляем (или комментируем) все упоминания нашего беспроводного интерфейса и добавляем свои:

```
iface wlan0 inet dhcp

wpa-driver wext

wpa-ssid <имя_точки_доступа>

wpa-ap-scan 1

wpa-proto RSN

wpa-pairwise CCMP

wpa-group CCMP

wpa-key-mgmt WPA-PSK

wpa-psk <hex_ключ> [ВАЖНО - см. Генерация ключей]

auto wlan0
```

Где:

Параметр	Описание
iface wlan0 inet dhcp	конфигурация для DHCP (для статического IP см. пример ниже) «iface wlan0» - Имя нашего интерфейса (wlan0, eth1, rausb0, ra0, итд.)
wpa-driver	Это wpa-driver для нашей карты (‘wext’ - общий драйвер для linux, работающий с «ndiswrapper»).

	<p>Рекомендуется оставить как есть.</p> <p>Другие варианты:</p> <p>hostap = Host AP driver (Intersil Prism2/2.5/3)</p> <p>atmel = ATMEL AT76C5XXx (USB, PCMCIA)</p> <p>wext = Linux wireless extensions (generic)</p> <p>madwifi = Atheros</p> <p>wired = wpa_supplicant wired Ethernet driver</p>
wpa-ssid	Имя нашей точки доступа (ESSID). Без кавычек.
wpa-ap-scan	<p>Параметр вещания точкой доступа ее имени.</p> <p>«1» = ESSID вещается всем.</p> <p>«2» = ESSID скрыт.</p>
wpa-proto	<p>Протокол</p> <p>«RSN» = WPA(2)</p> <p>«WPA» = WPA(1)</p>
wpa-pairwise & wpa-group	<p>«CCMP» = AES-шифрователь (как часть стандарта WPA(2))</p> <p>«TKIP» = TKIP-шифрователь (как часть стандарта WPA(1))</p>
wpa-key-mgmt	<p>«WPA-PSK» = Аутентификация через ключевую фразу (см. 'Создание ключей')</p> <p>«WPA-EAP» = Аутентификация при помощи сервера аутентификаций.</p>
auto wlan0	Автоматическое включение wlan0 при загрузке/перезагрузке сети.

Генерация ключей

Теперь нам нужно сконвертировать нашу ключевую фразу (WPA ASCII) в hex-ключ:

```
$ wpa_passphrase <имя_точки_доступа> <ascii_ключ>
```

Результат будет примерно таким:

```
network={
    ssid="test"

    #psk="12345678"

    psk=fe727aa8b64ac9b3f54c72432da14faed933ea511ecab1 5bbc6c52e7522f709a

}
```

hex-ключ это все символы после «psk=».

Нужно его скопировать в буфер обмена и вставить в файл /etc/network/interfaces в поле *wpa-psk*.

Теперь можно сохранить файл и перезагрузить сеть. Должно установиться соединение. Однако иногда этого сразу не происходит. Если это так - перезагружаем машину.

Дополнительно

Отключаем чтение файла /etc/network/interfaces для others во избежания попадания пароля от сети к третьим лицам.

```
$ sudo chmod o=-r /etc/network/interfaces
```

Примеры конфигураций

WPA2 + статический IP, скрытый ESSID.

```
iface wlan0 inet static
address 192.168.168.40
gateway 192.168.168.230
dns-nameservers 192.168.168.230
netmask 255.255.255.0
wpa-driver wext
wpa-ssid <имя_точки_доступа>
wpa-ap-scan 2
wpa-proto RSN
wpa-pairwise CCMP
wpa-group CCMP
wpa-key-mgmt WPA-PSK
wpa-psk <hex_ключ> [ВАЖНО - см. Генерация ключа]
auto wlan0
```

WPA1 + DHCP, ESSID виден всем.

```
iface wlan0 inet dhcp
wpa-driver wext
wpa-ssid <имя_точки_доступа>
wpa-ap-scan 1
wpa-proto WPA
wpa-pairwise TKIP
wpa-group TKIP
```

```
wpa-key-mgmt WPA-PSK  
  
wpa-psk <hex_ключ> [ВАЖНО - см. Генерация ключа]  
  
auto wlan0
```

Смешанный режим (WPA1, WPA2) + DHCP, ESSID виден всем.

```
iface wlan0 inet dhcp  
  
wpa-driver wext  
  
wpa-ssid <имя_точки_доступа>  
  
wpa-ap-scan 1  
  
wpa-proto WPA RSN  
  
wpa-pairwise TKIP CCMP  
  
wpa-group TKIP CCMP  
  
wpa-key-mgmt WPA-PSK  
  
wpa-psk <hex_ключ> [ВАЖНО - см. Генерация ключа]  
  
auto wlan0
```

LEAP, WEP, DHCP, ESSID виден всем.

```
iface wlan0 inet dhcp  
  
wpa-driver wext  
  
wpa-ssid <имя_точки_доступа>  
  
wpa-ap-scan 1  
  
wpa-eap LEAP  
  
wpa-key-mgmt IEEE8021X  
  
wpa-identity <имя_пользователя>  
  
wpa-password <пароль>  
  
auto wlan0
```

PEAP, AES, DHCP, ESSID виден всем.

```
iface wlan0 inet dhcp  
  
wpa-driver wext
```

```
wpa-ssid <имя_точки_доступа>

wpa-ap-scan 1

wpa-proto RSN

wpa-pairwise CCMP

wpa-group CCMP

wpa-eap PEAP

wpa-key-mgmt WPA-EAP

wpa-identity <наш_identity>

wpa-password <пароль>

auto wlan0
```

TTLS, WEP, DHCP, ESSID виден всем.

```
iface wlan0 inet dhcp

wpa-driver wext

wpa-ssid <имя_точки_доступа>

wpa-ap-scan 1

wpa-eap TTLS

wpa-key-mgmt IEEE8021X

wpa-anonymous-identity <anonymous_identity>

wpa-identity <наш_identity>

wpa-password <наш_password>

wpa-phase2 auth=PAP [Также: CHAP, MSCHAP, MSCHAPV2]

auto wlan0
```

Другие способы работы Wi-Fi оборудования

При помощи Wi-Fi адаптера также возможно установить децентрализованную сеть ad-хос или сделать из компьютера под управлением Ubuntu точку доступа. Поскольку описание данных способов Wi-Fi подключения выходит за рамки этого руководства - обратитесь к соответствующим разделам. Ссылки на эти разделы см. в разделе [Ссылки](#).

Решение проблем

Не устанавливается соединение по Wi-Fi/Ethernet с точкой доступа/маршрутизатором

Симптомы: сеть обычно изначально работает нормально, долго или недолго, а затем неожиданно пропадает и не появляется после перезагрузки. Эта проблема может быть непостоянной. Сеть «сама собой» начинает работать, а затем пропадает вновь. При перезапуске адаптера сети таким образом:

```
sudo ifdown wlan0
```

```
sudo ifup wlan0
```

будет выводиться в консоль похожий текст

```
Listening on LPF/wlan0/00-02-2A-E1-E0-6C
```

```
Sending on LPF/wlan0/00-02-2A-E1-E0-6C
```

```
Sending on Socket/fallback
```

```
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 8
```

```
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 8
```

```
DHCPDISCOVER on wlan0 to 255.255.255.255 port 67 interval 15
```

```
No DHCPOFFERS received.
```

```
No working leases in persistent database - sleeping.
```

Причиной проблемы может быть то, что материнская плата полностью не обесточивается при выключении компьютера. При этом вероятно не обесточивается и некоторое периферийное оборудование, в т.ч. могут не обесточиваться usb порты. Если вы используете, например, Wi-Fi usb-адаптер, то в таком случае можно заметить горящий на адаптере светодиод (если он им оборудован). Вероятно проблема возникает из-за того, что сетевое оборудование в этом режиме работает не совсем корректно.

Простое решение проблемы состоит в выключении компьютера и выдергивании кабеля питания из розетки с последующим подключением шнура питания и включением компьютера.

Сложное решение проблемы состоит в настройке параметров BIOS-а на полное обесточивание сетевого оборудования при выключении компьютера.

Иногда наглухо пропадает соединение по Wi-Fi с точкой доступа/маршрутизатором

Симптомы: сеть изначально работает, а затем после перезагрузки точки доступа/маршрутизатора неожиданно пропадает, и не появляется ни после перезагрузки, ни после танцев с бубном. При этом беспроводной адаптер в упор не видит точку доступа (хотя она может стоять рядом с компьютером), но

прекрасно видит все соседские сети. Затем после ~дцатой~ перезагрузки маршрутизатора сеть сама собой появляется вновь.

Причиной проблемы может быть то, что некоторые маршрутизаторы произвольно выбирают номер рабочего канала, игнорируя номер канала выбранный в настройках маршрутизатора. Если в файле `/etc/network/interfaces` номер канала для беспроводного интерфейса указан, то вероятно проблема состоит именно в этом. Номер б канала указывается в файле примерно так:

```
auto wlan0

...

wireless-channel 6
```

Простое решение проблемы состоит в комментировании этого параметра, чтобы адаптер не был ограничен только этим каналом, и перезапуске сети

```
auto wlan0

...

#wireless-channel 6
```

Сложное решение проблемы состоит в регистрации бага на сайте производителя маршрутизатора (прошивки для него) и обновление прошивки маршрутизатора после (в случае) его исправления.

Перезапуск сети

Теперь, когда все необходимые действия выполнены, можно перезапустить сеть и проверить соединение. Для этого:

```
$ sudo /etc/init.d/networking restart
```

Теперь, при запуске команды `ifconfig` должно отобразиться подключение `eth0` с выставленными параметрами. Если подключение отображается, но параметры не такие, какие были указаны в файле `/etc/network/interfaces`, либо возникают любые другие ошибки, перепроверьте этот файл на наличие неточностей и опечаток и попробуйте ещё раз перезапустить сеть.

FAQ по сетям

Как зайти на мой компьютер извне (через интернет)?

Для начала надо узнать, какой IP-адрес даёт провайдер — серый или белый (не путать со статическим/динамическим). Если серый, то ничего не получится. Если белый, то возможны два варианта:

- Роутера нет или он работает в режиме бриджа (моста). В этом случае белый IP-адрес присваивается самому компьютеру. Вводим адрес — попадаем на комп, всё просто.

- Белый адрес присваивается роутеру. Соответственно, по этому адресу мы попадаем на роутер, а не на компьютер. Чтобы попасть на компьютер, на роутере нужно пробросить порты (см. ниже).

Мне кажется, у меня слишком медленно работает сеть!

Измерьте скорость сети между двумя компьютерами с помощью `iperf`. Можно воспользоваться [этой](#) инструкцией. В ней предлагают скомпилировать программу из исходников, но можно просто [установить её из репозитория](#). Если `iperf` покажет значение немного меньшее, чем ожидаемое, то с сетью всё в порядке, проблема может быть в железе (жёсткий диск/процессор не могут обеспечить большую скорость), в способе передачи (например, `scp` и `ftp` весьма неторопливы), в настройках (скорость может быть ограничена, например, настройками FTP-сервера) или в чём-то ещё. Если `iperf` показал величину, которая в разы меньше желаемой, то да - с сетью проблемы. Стоит посмотреть, в нужном ли режиме работает карта (например, с помощью `ethtool`), проверить наличие «errors» в выводе `ifconfig` и протестировать скорость подключения к какому-нибудь третьему компьютеру.

Как узнать, какие программы слушают порты на моём компьютере?

Чтобы посмотреть список открытых портов и названия слушающих их программ, используйте команду:

```
sudo netstat -nlpA inet,inet6
```

Для вывода информации о конкретном порте можно использовать `grep`. Например, для 80 порта:

```
sudo netstat -nlpA inet,inet6 | grep :80
```

Из вывода `netstat` не всегда понятно, о какой программе идёт речь (например, `2671/python`), подробнее о процессе расскажет `ps`:

```
ps aux | grep 2671
```

Как присвоить два IP-адреса одной сетевой карте?

Например, интерфейсу `eth0` нужно добавить адрес `192.168.1.1`. Кратковременно, до перезапуска сети:

```
sudo ifconfig eth0:1 192.168.1.1
```

Навсегда — добавить в `/etc/network/interfaces` следующее:

```
#исправить строку auto
auto eth0 eth0:1
```

```
# добавить алиас

iface eth0:1 inet static

    address 192.168.1.1

    netmask 255.255.255.0
```

Как пробросить порт?

Например, нужно пробросить порт 8081. Адрес, на который обращается клиент, назовём *внешний_ip*, а адрес, на который он должен попасть — *внутренний_ip*.

```
iptables -t nat -A PREROUTING -p tcp -d внешний_ip --dport 8081 -j DNAT --to-destination внутренний_ip:8081

iptables -t nat -A POSTROUTING -p tcp --dst внутренний_ip --dport 8081 -j SNAT --to-source внешний_ip
```

И обязательно нужно что-то вроде

```
iptables -t filter -A FORWARD -m conntrack --ctstate DNAT -j ACCEPT
```

Ссылки

Раздел				Администрирование	
Отключение		Network		Manager'a	
Скрипт-помощник		для	настройки	VPN	соединения
Программная		точка		доступа	
Раздел	часто	задаваемых	вопросов	на	форуме
Описание		стандарта		Wi-Fi	
Описание		протокола		WPA2	
Немного	интересного	про	безопасность	Wi-Fi	сетей
Описание		ad		hoc	
Статья	о	настройке	Wi-Fi	на	ubuntuforums.org
Ресурс,	посвященный	Wi-Fi	устройствам	в	Linux
Сайт "Убунтология". Настройка сети вручную.					