



# **Installation Guide For Linux Driver**



# Contents

1. Development Environment .....	1
2. Compile the Driver .....	1
2.1. Compilation tool and kernel sources .....	1
2.2. Compile the Driver .....	1
3. Load the Driver .....	2
4. Join the Wireless Network .....	2
4.1. Use the Graphical Interface .....	2
4.2. Use the Command .....	4
4.2.1. Identify the device .....	4
4.2.2. Create the interface .....	4
4.2.3. Change the interface status to <i>up</i> .....	4
4.2.4. Start wpa_supplicant in the background .....	4
4.2.5. Scan AP and See Results.....	5
4.2.6. Connect to the AP .....	5
4.2.7. Enable DHCP client .....	7

## 1. Development Environment

System version: Ubuntu 14.04.1

Kernel version: 3.16.0-30-generic

Gcc version: 4.8.2

## 2. Compile the Driver

Before you compile the driver, please make sure you have the correct compilation tool and kernel sources.

### 2.1. Compilation tool and kernel sources

A compilation tool (e.g. gcc) is required for your native compilation. If you have not installed a compilation tool, run the following command to confirm the tool version for the kernel sources and then install the same version tool to compile the driver.

```
cat /proc/version
```

Command

Confirm and install the compilation tool. The information in red indicates the tool version for your kernel sources.

```
Linux version 3.16.0-30-generic (buildd@phianna) (gcc version 4.8.2
(Ubuntu 4.8.2-19ubuntu1) ) #40~14.04.1-Ubuntu SMP Thu Jan 15
17:45:15 UTC 2015
```

Result

Run the following command if you have connected to the Internet.

```
apt-get install gcc-4.8.2
```

Confirm the kernel sources. If you have no kernel sources, run the following command to confirm the kernel source version and then download the corresponding kernel sources at <http://www.kernel.org/>

```
cat /proc/version or uname -a
```

After you download the file, extract it into /usr/src/.

*Note: We recommend that you do NOT use apt-get install linux-source to download the kernel sources.*

### 2.2. Compile the Driver

Edit the Make file, KSRC, ARCH and other fields in the driver.

```
ifeq ($(CONFIG_PLATFORM_I386_PC), y)
EXTRA_CFLAGS += -DCONFIG_LITTLE_ENDIAN
```

```

SUBARCH := $(shell uname -m | sed -e s/i.86/i386/)
ARCH ?= $(SUBARCH)
CROSS_COMPILE ?=
KVER := $(shell uname -r)
KSRC := /lib/modules/$(KVER)/build
MODDESTDIR := /lib/modules/$(KVER)/kernel/drivers/net/wireless/
INSTALL_PREFIX :=
endif

```

Explanation:

- KSRC is used to specify the kernel source path for driver compilation.
- CROSS\_COMPILE is used to specify the toolchain.
- ARCH is used to specify the target platform's CPU architectures such as arm, mips, i386 and so on.

If your Linux kernel does not support 802.11, close macro CONFIG\_IOCTL\_CFG80211

```

CONFIG_IOCTL_CFG80211=n
ifeq ($(strip &(CONFIG_IOCTL_CFG80211)),y)
EXTRA_CFLAGS += -DCONFIG_IOCTL_CFG80211 = 1
EXTRA_CFLAGS += -DRTW_USE_CFG80211_STA_EVENT = 1
endif

```

After all parameters are configured, go to the directory of the original driver file and run the command “make”.

### 3. Load the Driver

Here we shows the 8812au.ko wireless driver loading process as an example.

```

sudo cp 8812au.ko /lib/modules/[kernel version]/kernel/drivers/net/wireless/
sudo depmod -a
sudo modprobe 8812au

```

Or you can directly load the driver shown as below.

```

sudo insmod 8812au.ko

```

Run the following command to check if the driver is successfully loaded.

```

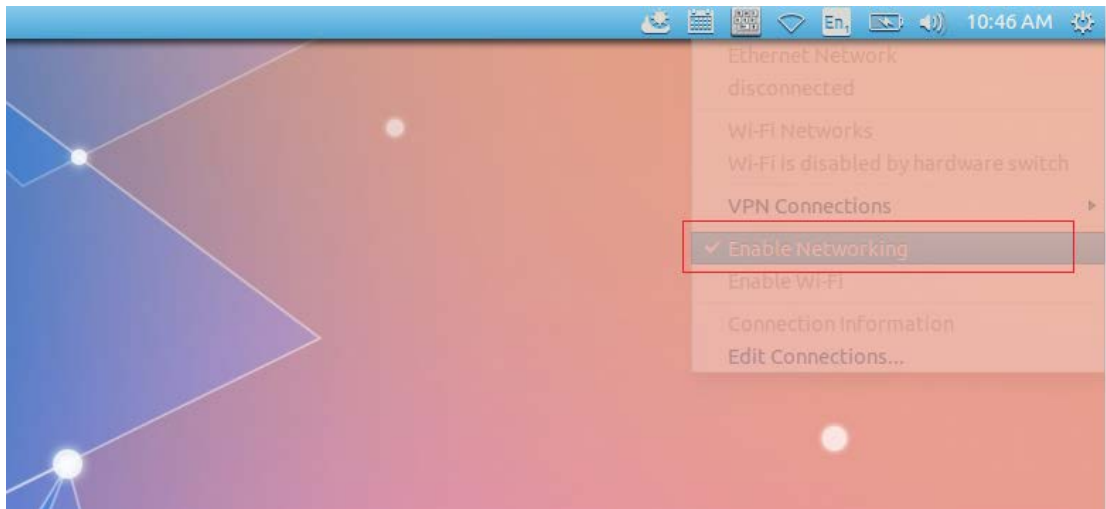
lsmod

```

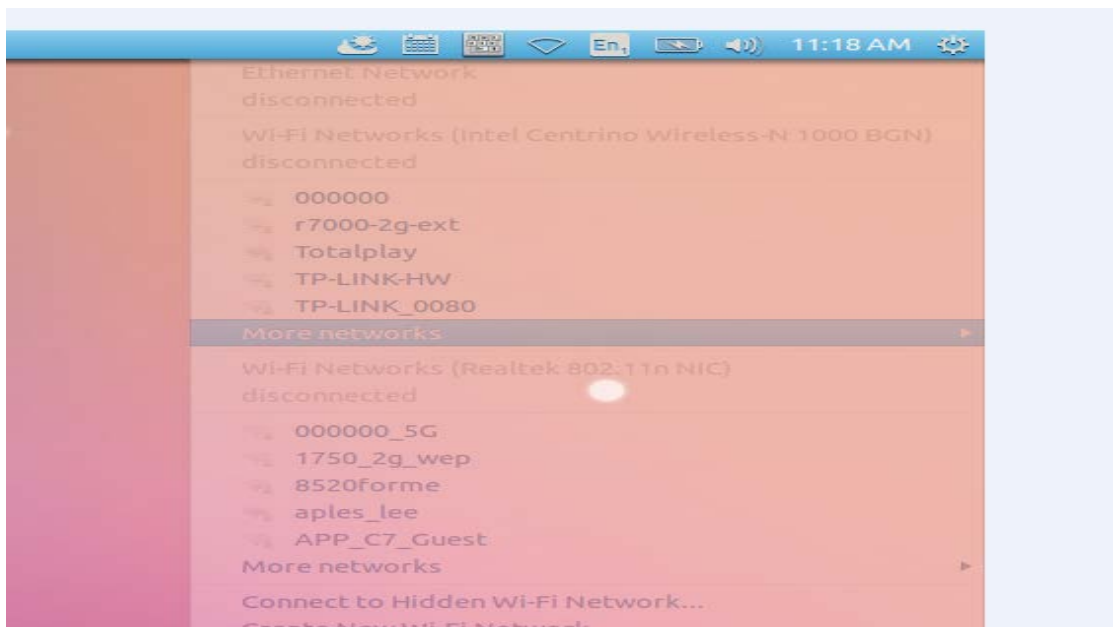
### 4. Join the Wireless Network

#### 4.1. Use the Graphical Interface

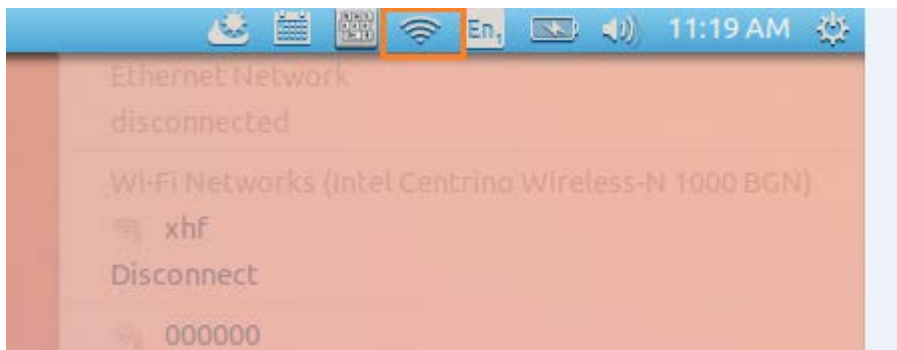
1. Enable your network shown as below.



2. Select the desired wireless network and specify the corresponding adapter interface for connection.



3. The wireless icon indicates a successful connection.



## 4.2. Use the Command

### 4.2.1. Identify the device

Insert the USB adapter and run the following command to check if the adapter is identified.

```
lsusb
```

### 4.2.2. Create the interface

Run the following command to check if the wireless network interface is created.

```
ifconfig
```

### 4.2.3. Change the interface status to *up*

Check if the WLAN interface is *up*. If not, run the following command. Here we use *wlan1* as an example.

```
ifconfig wlan1 up
```

If it failed to change to *up*, run the following command.

```
rfkill unblock wifi  
ifconfig wlan1 up
```

### 4.2.4. Start wpa\_supplicant in the background

Run the following command:

```
wpa_supplicant -Dnl80211 -iwlan1 -c ./wpa_0_8.conf -B
```

*Note: wpa\_0\_8.conf is a file in the current driver directory, go to the driver directory when running the command.*

If the command above is not effective, run the following command to end the wpa\_supplicant procedure and then run the above command again.

```
killall wpa_supplicant
```

If your Linux kernel does not support 802.11, run the following command.

```
wpa_supplicant -Dwext -iwlan0 -c ./wpa_0_8.conf -B
```

#### 4.2.5. Scan AP and See Results

Run the following command:

```
wpa_cli -p /var/run/wpa_supplicant scan  
wpa_cli -p /var/run/wpa_supplicant scan_results
```

#### 4.2.6. Connect to the AP

##### 1) Open

Run the following command:

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0  
wpa_cli -p /var/run/wpa_supplicant ap_scan 1  
wpa_cli -p /var/run/wpa_supplicant add_network  
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink" //tplink is the SSID  
of the desired AP. The SSID is in double quotation marks and then as a whole  
enclosed by single quotation marks.  
wpa_cli -p /var/run/wpa_supplicant set_network 0 key_mgmt NONE  
wpa_cli -p /var/run/wpa_supplicant select_network 0
```

##### 2) WEP40 with open system

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0  
wpa_cli -p /var/run/wpa_supplicant ap_scan 1  
wpa_cli -p /var/run/wpa_supplicant add_network  
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink"  
wpa_cli -p /var/run/wpa_supplicant set_network 0 key_mgmt NONE  
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_key0 1234567890  
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_tx_keyidx 0  
wpa_cli -p /var/run/wpa_supplicant select_network 0
```

##### 3) WEP40 with shared key

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0  
wpa_cli -p /var/run/wpa_supplicant ap_scan 1  
wpa_cli -p /var/run/wpa_supplicant add_network  
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink"  
wpa_cli -p /var/run/wpa_supplicant set_network 0 key_mgmt NONE  
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_key0 1234567890  
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_tx_keyidx 0  
wpa_cli -p /var/run/wpa_supplicant set_network 0 auth_alg SHARED  
wpa_cli -p /var/run/wpa_supplicant select_network 0
```

4) WEP 104 with open system

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0
wpa_cli -p /var/run/wpa_supplicant ap_scan 1
wpa_cli -p /var/run/wpa_supplicant add_network
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink"
wpa_cli -p /var/run/wpa_supplicant set_network 0 key_mgmt NONE
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_key0
12345678901234567890123456
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_tx_keyidx 0
wpa_cli -p /var/run/wpa_supplicant select_network 0
```

5) WEP 104 with shared key

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0
wpa_cli -p /var/run/wpa_supplicant ap_scan 1
wpa_cli -p /var/run/wpa_supplicant add_network
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink"
wpa_cli -p /var/run/wpa_supplicant set_network 0 key_mgmt NONE
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_key0
12345678901234567890123456
wpa_cli -p /var/run/wpa_supplicant set_network 0 wep_tx_keyidx 0
wpa_cli -p /var/run/wpa_supplicant set_network 0 auth_alg SHARED
wpa_cli -p /var/run/wpa_supplicant select_network 0
```

*Note:*

*If the WEP key is ASCII, run the following command:*

- *WEP40: wpa\_cli -p/var/run/wpa\_supplicant set\_network 0 wep\_key0 "12345"*
- *WEP104: wpa\_cli -p/var/run/wpa\_supplicant set\_network 0 wep\_key0  
"1234567890123"*

*If the index for WEP key is 0-3, run the following command*

- *wpa\_cli -p/var/run/wpa\_supplicant set\_network 0 wep\_keyX  
12345678901234567890123456*
- *wpa\_cli -p/var/run/wpa\_supplicant set\_network 0 wep\_tx\_keyidx X*

6) TIKP/AES

```
wpa_cli -p /var/run/wpa_supplicant remove_network 0
wpa_cli -p /var/run/wpa_supplicant ap_scan 1
wpa_cli -p /var/run/wpa_supplicant add_network
wpa_cli -p /var/run/wpa_supplicant set_network 0 ssid "tplink"
wpa_cli -p /var/run/wpa_supplicant set_network 0 psk "12345678"
wpa_cli -p /var/run/wpa_supplicant select_network 0
```



#### 4.2.7. Enable DHCP client

Run the following command

```
dhclient wlan1
```

After running the command, the adapter will get an IP assigned by the AP. Then you can run the ping command to check if the wireless connection is successful.

```
tplink@tplink-Inspiron-N4010:~/driver$ ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:295 errors:0 dropped:0 overruns:0 frame:0
          TX packets:295 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:22543 (22.5 KB)  TX bytes:22543 (22.5 KB)

wlan1     Link encap:Ethernet  HWaddr c4:e9:84:1f:df:3c
          inet addr:192.168.1.102  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::c6e9:84ff:fe1f:df3c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16 errors:0 dropped:699 overruns:0 frame:0
          TX packets:66 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2068 (2.0 KB)  TX bytes:11368 (11.3 KB)

tplink@tplink-Inspiron-N4010:~/driver$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data:
64 bytes from 192.168.1.1: icmp_seq=1 ttl=254 time=11.8 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=254 time=7.05 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=254 time=1.97 ms
^C
--- 192.168.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 1.971/6.968/11.882/4.046 ms
tplink@tplink-Inspiron-N4010:~/driver$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.1.1    0.0.0.0         UG    0      0        0 wlan1
192.168.1.0     *              255.255.255.0   U     0      0        0 wlan1
```

*Note: Run the commands under the root account.*