

目次

USB WiFi Dongleで無線LANに接続する

1

ファームウェアの作成

1

Kernel config のカスタマイズ

1

無線LANデバイス ファームウェアファイルの導入

4

無線LAN関係 パッケージの導入

5

ファームウェアのビルド

7

作成したファームウェアでの起動

8

設定

9

クライアントとして利用

9

アクセスポイントとして利用

11

アクセスポイントとして利用 (Multiple SSID)

18

ダウンロード

18

USB WiFi Dongleで無線LANに接続する

MA-E3xxは、本体に無線LAN I/Fを持っていません¹⁾が、USB WiFi Dongleを使用することで無線LANに接続することができます。



ファームウェアの作成

Kernel config のカスタマイズ

標準のファームウェアの Kernel では、

- 無線LANのスタック
- 無線LANデバイスのドライバ

を組み込んでいませんのでKernel をカスタマイズしたファームウェアを作成します。

参照: [アップデート用ファームウェアの作成\(クロス\)](#)

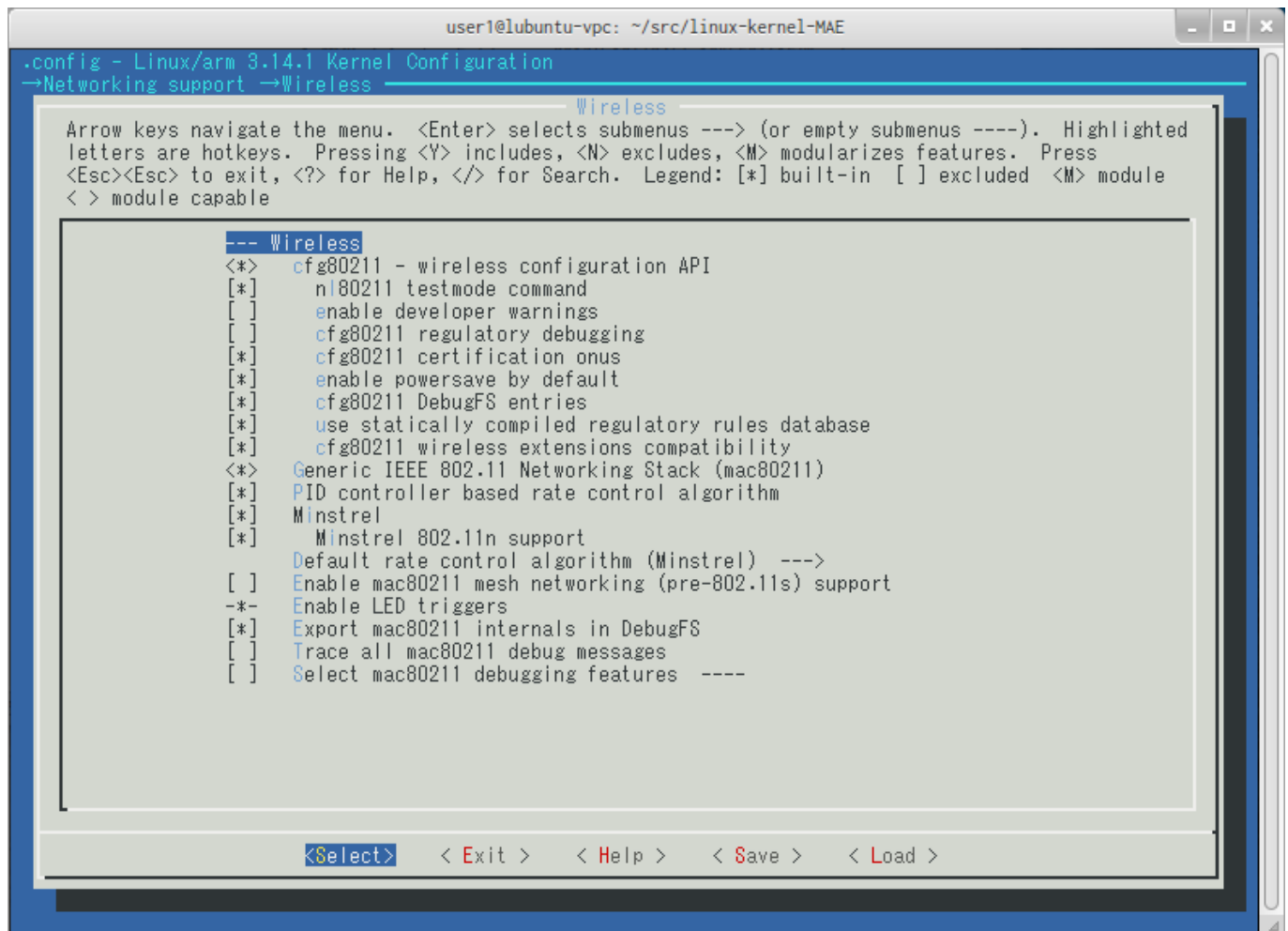
v2.5.0系で、無線 LAN のサポートを追加しました。対応デバイスは、MediaTek(旧Ralink)/Atheros の USB-WiFi デバイスとなります。

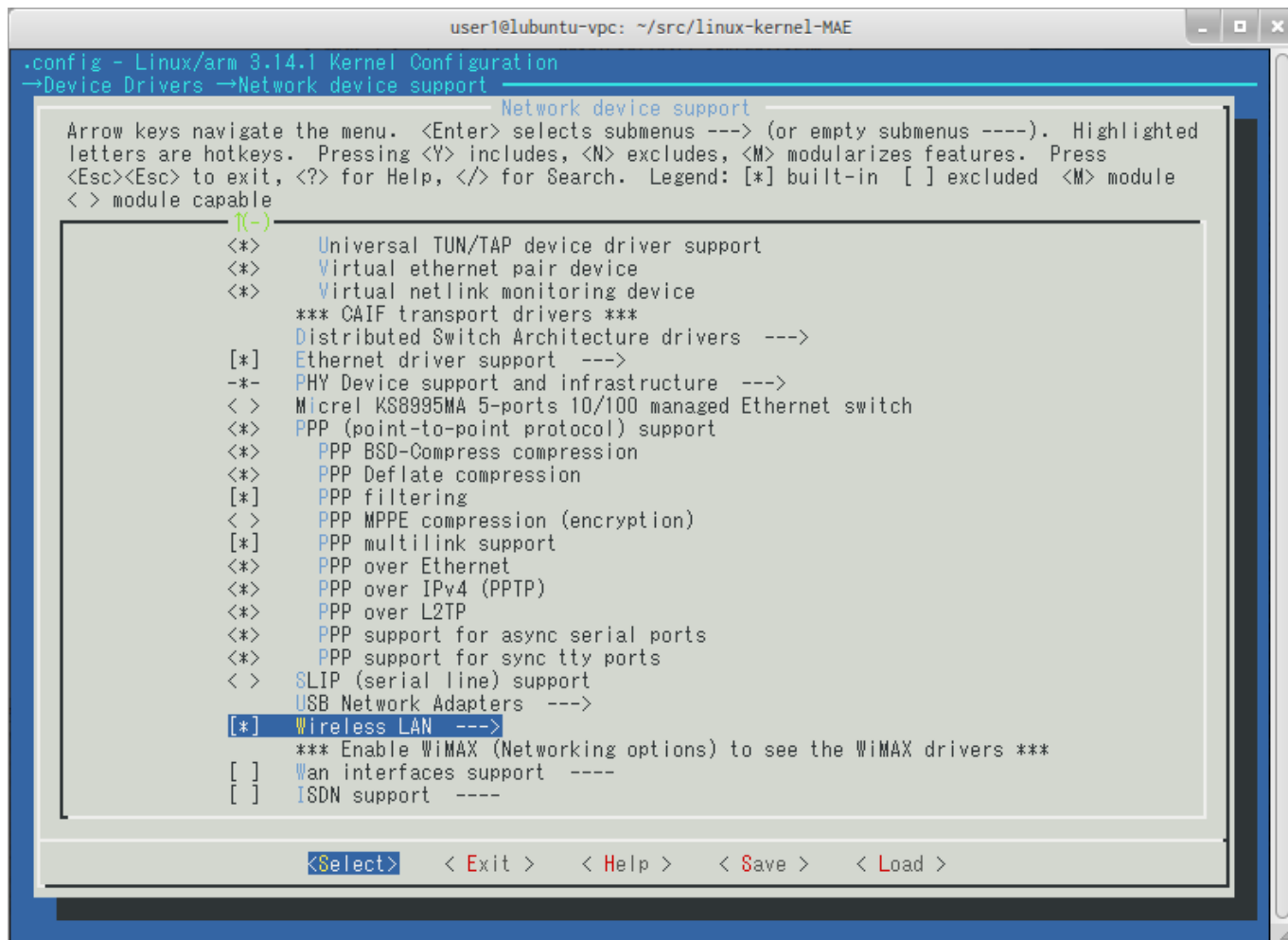
[こちら](#)からダウンロード可能です。

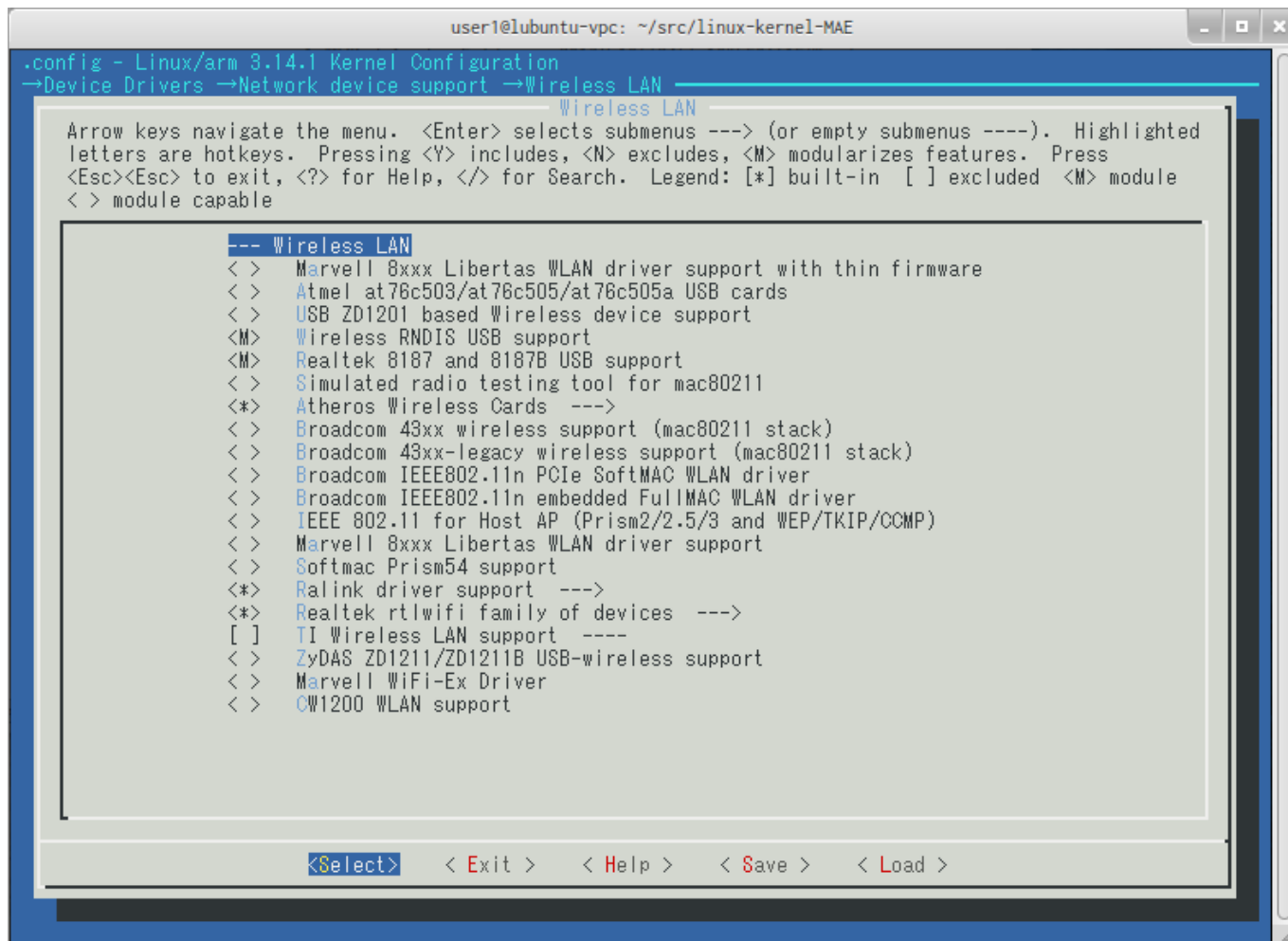
Kernel の config (make menuconfig) にて、

- Networking support —> Wireless
- Device Drivers —> Network device support —> Wireless LAN
- Wireless LAN以下の各種デバイス用ドライバ (Atheros/Ralink/Realtek)

を組み込みもしくはモジュールとして有効化します。







無線LANデバイス ファームウェアファイルの導入

たいていのUSBWiFi Dongleは、バイナリで配布されているファームウェアファイルが必要となります。下記ページを参考にデバイスを特定、必要なファームウェアファイルを rootfs のベースにコピーしておきます。

[Linux Wireless - Supported devices - USB](#)

[Debian Wiki - WiFi](#)

Realtek RTL8192CU, Ralink RT2870 を使う場合の例

```
root@lubuntu-vpc:/home/user1/work/mae320_trusty_wifi_rootfs# ls -lR
lib/firmware/
lib/firmware/:
合計 12
-rw-r--r-- 1 root root 8192  3月 6 00:45 rt2870.bin
drwxr-xr-x 2 root root 4096  4月 16 13:56 rtlwifi

lib/firmware/rtlwifi:
合計 16
-rw-r--r-- 1 root root 16116  3月 6 00:45 rtl8192cufw_TM5C.bin
root@lubuntu-vpc:/home/user1/work/mae320_trusty_wifi_rootfs#
```

無線LAN関係 パッケージの導入

無線LANに接続するために必要なパッケージを、ホストPC上で導入しておきます。

参照: [PC上で root filesystem を変更する\(QEMU使用\)](#)

クライアントとして利用する場合

wpa_supplicant パッケージを導入します。

```
root@lubuntu-vpc:~# apt-cache search wpa_supplicant
wpa_supplicant - client support for WPA and WPA2 (IEEE 802.11i)
root@lubuntu-vpc:~#
```

```
root@lubuntu-vpc:~# apt-get install wpa_supplicant
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libnl-3-200 libnl-genl-3-200 libreadline5
Suggested packages:
  wpagui libengine-pkcs11-openssl
The following NEW packages will be installed:
  libnl-3-200 libnl-genl-3-200 libreadline5 wpa_supplicant
0 upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
Need to get 794 kB of archives.
After this operation, 1981 kB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ports.ubuntu.com/ubuntu-ports/ trusty/main libnl-3-200 armhf
3.2.21-1 [39.3 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports/ trusty/main libnl-genl-3-200
armhf 3.2.21-1 [8562 B]
Get:3 http://ports.ubuntu.com/ubuntu-ports/ trusty/main libreadline5 armhf
5.2+dfsg-2 [106 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports/ trusty/main wpa_supplicant armhf
2.1-0ubuntu1 [641 kB]
Fetched 794 kB in 5s (155 kB/s)
Selecting previously unselected package libnl-3-200:armhf.
(Reading database ... 17511 files and directories currently installed.)
Preparing to unpack .../libnl-3-200_3.2.21-1_armhf.deb ...
Unpacking libnl-3-200:armhf (3.2.21-1) ...
Selecting previously unselected package libnl-genl-3-200:armhf.
Preparing to unpack .../libnl-genl-3-200_3.2.21-1_armhf.deb ...
Unpacking libnl-genl-3-200:armhf (3.2.21-1) ...
Selecting previously unselected package libreadline5:armhf.
Preparing to unpack .../libreadline5_5.2+dfsg-2_armhf.deb ...
Unpacking libreadline5:armhf (5.2+dfsg-2) ...
Selecting previously unselected package wpa_supplicant.
Preparing to unpack .../wpa_supplicant_2.1-0ubuntu1_armhf.deb ...
Unpacking wpa_supplicant (2.1-0ubuntu1) ...
```

```
Setting up libnl-3-200:armhf (3.2.21-1) ...
Setting up libnl-genl-3-200:armhf (3.2.21-1) ...
Setting up libreadline5:armhf (5.2+dfsg-2) ...
Setting up wpasupplicant (2.1-0ubuntu1) ...
Processing triggers for libc-bin (2.19-0ubuntu6) ...
localepurge: Disk space freed in /usr/share/locale: 0 KiB
localepurge: Disk space freed in /usr/share/man: 0 KiB

Total disk space freed by localepurge: 0 KiB

root@lubuntu-vpc:~#
```

アクセスポイントとして利用する場合

hostapd パッケージを導入します。

```
root@lubuntu-vpc:~# apt-cache search hostapd
hostapd - user space IEEE 802.11 AP and IEEE 802.1X/WPA/WPA2/EAP
Authenticator
root@lubuntu-vpc:~#
```

```
root@lubuntu-vpc:~# apt-get install hostapd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  hostapd
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 360 kB of archives.
After this operation, 874 kB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports/ trusty/universe hostapd armhf
1:2.1-0ubuntu1 [360 kB]
Fetched 360 kB in 2s (138 kB/s)
Selecting previously unselected package hostapd.
(Reading database ... 17586 files and directories currently installed.)
Preparing to unpack .../hostapd_1%3a2.1-0ubuntu1_armhf.deb ...
Unpacking hostapd (1:2.1-0ubuntu1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Setting up hostapd (1:2.1-0ubuntu1) ...
runlevel:/var/run/utmp: No such file or directory
Processing triggers for ureadahead (0.100.0-16) ...
localepurge: Disk space freed in /usr/share/locale: 0 KiB
localepurge: Disk space freed in /usr/share/man: 0 KiB

Total disk space freed by localepurge: 0 KiB

root@lubuntu-vpc:~#
```


ファームウェアのビルド

以上の作業で、

- Kernel config のカスタマイズ
- root filesystem へのUSB WiFiファームウェア/パッケージ導入

が完了しましたので、ファームウェアとしてビルドします。

参照: [アップデート用ファームウェアの作成\(クロス\)](#)

```
user1@lubuntu-vpc:~/work$ sudo ./build_firm.sh -k ../src/linux-kernel-MAE -r
mae320_trusty_wifi_rootfs -o mae3xx_trusty_wifi.img
[sudo] password for user1:
MA-E3xx series Firmware build script
* kernelrelease: 3.14.1+
  CHK      include/config/kernel.release
  CHK      include/generated/uapi/linux/version.h
  CHK      include/generated/utsrelease.h
make[1]: `include/generated/mach-types.h' は更新済みです
  CALL     scripts/checksyscalls.sh
―― 中略 ――
  crontab (103)
  fuse (105)
* create initramfs ...
update-initramfs: Generating /boot/initrd.img-3.14.1+
/bin/df: Warning: cannot read table of mounted file systems: No such file or
directory
warning: failed to read mtab
  adding: kernel.img (stored 0%)
  adding: kmod.img (stored 0%)
  adding: initramfs.img (stored 0%)
  adding: rootfs.img (stored 0%)
  adding: dtb.img (deflated 78%)
* Firmware filename: /home/user1/work/mae3xx_trusty_wifi.img
done.
please remove working directory '/home/user1/work/build_1397625772'
manually.
user1@lubuntu-vpc:~/work$
```

ファームウェアがビルドできました。

```
user1@lubuntu-vpc:~/work$ ls -l mae3xx_trusty_wifi.img
-rw-r--r-- 1 root root 67858264  4月 16 14:26 mae3xx_trusty_wifi.img
user1@lubuntu-vpc:~/work$
```

これをファームアップデートコマンドを利用して適用します。

参照: [ファームウェアの更新](#)

```
root@plum:~# mount /dev/mmcblk0p1 /mnt/sd/
root@plum:~# firmup /mnt/sd/mae3xx_trusty_wifi.img
### MA-E3xx series firmware update utility (for 2-area firmware).
```

```
Updating Area: 1 ...
* image file for kernel found.
  writing "kernel" image to /dev/mtdblock3 ... succeeded.
* image file for kmod found.
  writing "kmod" image to /dev/mtdblock7 ... succeeded.
* image file for initramfs found.
  writing "initramfs" image to /dev/mtdblock5 ... succeeded.
* image file for rootfs found.
  writing "rootfs" image to /dev/mtdblock9 ... succeeded.
* DTB file found.
  writing "dtb" image to /dev/mtd16 ... succeeded.
* updating default firmware area: 1
Succeeded.
root@plum:~#
```

作成したファームウェアでの起動

ファームウェアを適用し、再起動します。
試しに、[Planex GW-USNano2](#) をUSBに挿してみます。



```
[ 324.113638] usb 1-1: new high-speed USB device number 4 using musb-hdrc
[ 324.243130] usb 1-1: device v2019 pab2a is not supported
[ 324.249772] usb 1-1: New USB device found, idVendor=2019, idProduct=ab2a
[ 324.249811] usb 1-1: New USB device strings: Mfr=1, Product=2,
SerialNumber=3
[ 324.249837] usb 1-1: Product: GW-USNANO2
[ 324.249862] usb 1-1: Manufacturer: Planex Communications Inc.
[ 324.249886] usb 1-1: SerialNumber: 00e04cxxxxxx
[ 324.263160] rtl8192cu: Chip version 0x10
[ 324.373640] rtl8192cu: MAC address: 00:22:cf:66:xx:xx
[ 324.373699] rtl8192cu: Board Type 0
[ 324.373854] rtl_usb: rx_max_size 15360, rx_urb_num 8, in_ep 1
```

```
[ 324.374099] rtl8192cu: Loading firmware rtlwifi/rtl8192cufw_TMSC.bin
[ 324.376378] ieee80211 phy2: Selected rate control algorithm 'rtl_rc'
[ 324.390511] rtlwifi: wireless switch is on
user1@plum:~$
```

```
user1@plum:~$ ifconfig wlan0
wlan0      Link encap:Ethernet  HWaddr 00:22:cf:66:xx:xx
           BROADCAST MULTICAST  MTU:1500  Metric:1
           RX packets:0 errors:0 dropped:0 overruns:0 frame:0
           TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
           collisions:0 txqueuelen:1000
           RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

user1@plum:~$
```

認識され、wlan0 インターフェイスが作成されていることが確認できました。

設定

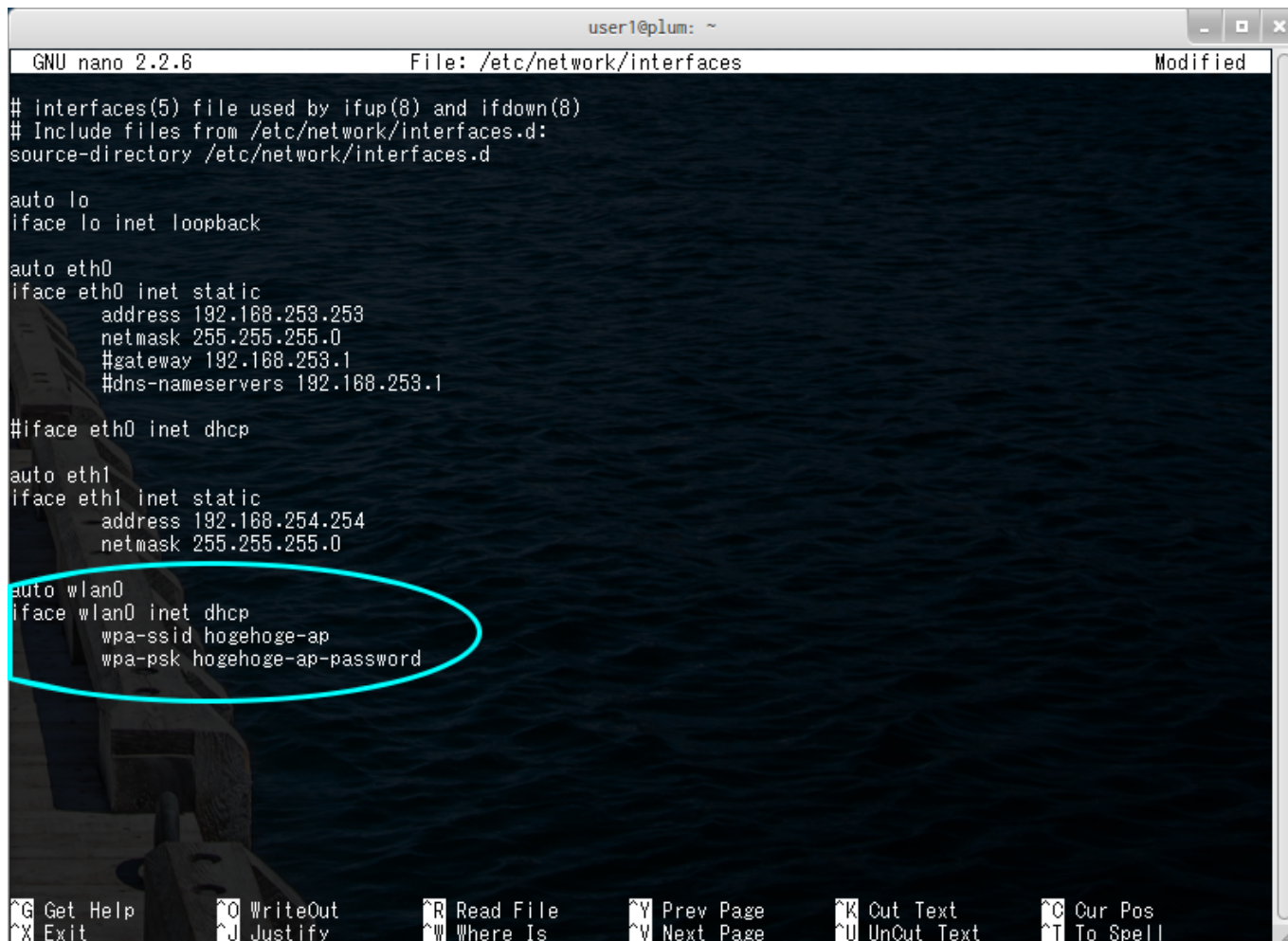
クライアントとして利用

SSID, プリシェアードキーの設定

導入しておいた wpa_supplicant を利用する設定をします。

参照: [Debian Wiki - WiFi HowToUse](#)

```
user1@plum:~$ sudo nano -w /etc/network/interfaces
```



```
user1@plum: ~  
GNU nano 2.2.6 File: /etc/network/interfaces Modified  
# interfaces(5) file used by ifup(8) and ifdown(8)  
# Include files from /etc/network/interfaces.d:  
source-directory /etc/network/interfaces.d  
  
auto lo  
iface lo inet loopback  
  
auto eth0  
iface eth0 inet static  
    address 192.168.253.253  
    netmask 255.255.255.0  
    #gateway 192.168.253.1  
    #dns-nameservers 192.168.253.1  
  
#iface eth0 inet dhcp  
  
auto eth1  
iface eth1 inet static  
    address 192.168.254.254  
    netmask 255.255.255.0  
  
auto wlan0  
iface wlan0 inet dhcp  
    wpa-ssid hogehoge-ap  
    wpa-psk hogehoge-ap-password  
  
^G Get Help  ^O WriteOut  ^R Read File  ^V Prev Page  ^K Cut Text   ^C Cur Pos  
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

“wlan0” の設定を追加します。

“wpa-ssid”, “wpa-psk” はそれぞれ接続するアクセスポイントのSSID, プリシェアードキーを設定してください。

暗号キーを設定してますので、一般ユーザで読めないよう、忘れずにファイルのパーミッションを適切に設定しておいてください。

```
user1@plum:~$ sudo chmod 640 /etc/network/interfaces  
user1@plum:~$ ls -l /etc/network/interfaces  
-rw-r----- 1 root root 497 Apr 16 14:57 /etc/network/interfaces  
user1@plum:~$ cat /etc/network/interfaces  
cat: /etc/network/interfaces: Permission denied  
user1@plum:~$
```

接続

ifup コマンドで接続します。

```
user1@plum:~$ sudo ifup wlan0  
dhcpcd.sh: interface wlan0 has been configured with new IP=192.168.253.212
```

```
user1@plum:~$
```

```
user1@plum:~$ ifconfig wlan0
wlan0      Link encap:Ethernet  HWaddr 00:22:cf:66:xx:xx
            inet addr:192.168.253.212  Bcast:192.168.253.255
Mask:255.255.255.0
            inet6 addr: fe80::222:cfff:fe66:xxxx/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:9 errors:0 dropped:0 overruns:0 frame:0
            TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:1383 (1.3 KB)  TX bytes:2040 (2.0 KB)
```

```
user1@plum:~$
```

Kernelのログを見ると、このように記録されています。

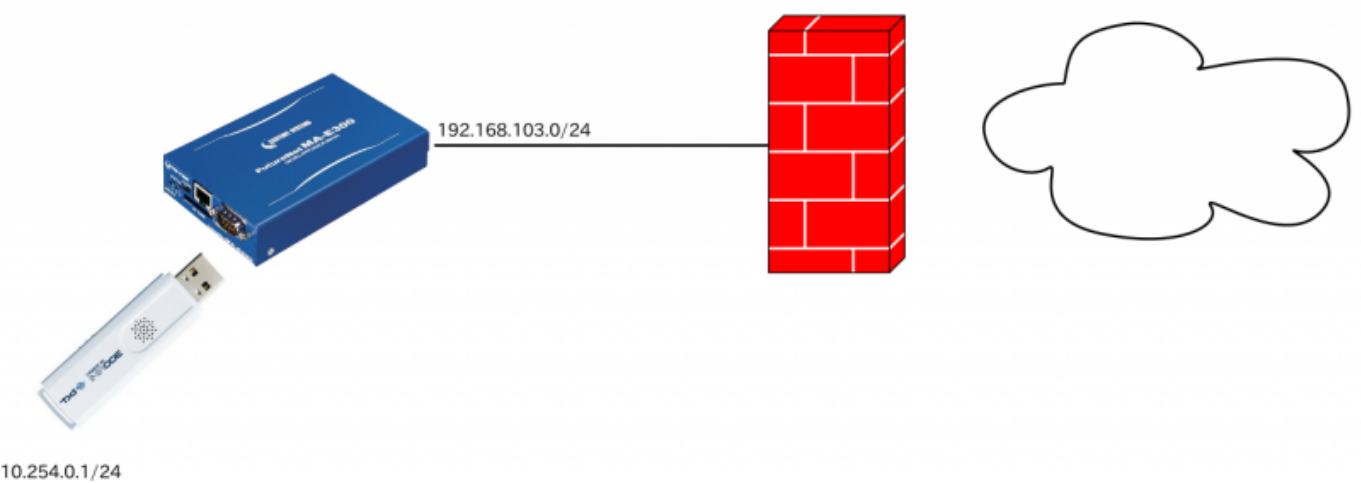
```
[ 2288.048683] rtl8192cu: MAC auto ON okay!
[ 2288.065009] rtl8192cu: Tx queue select: 0x05
[ 2288.440590] IPv6: ADDRCONF(NETDEV_UP): wlan0: link is not ready
[ 2290.860780] wlan0: authenticate with 00:0a:79:d4:xx:xx
[ 2290.884444] wlan0: send auth to 00:0a:79:d4:xx:xx (try 1/3)
[ 2290.906213] wlan0: authenticated
[ 2290.908574] wlan0: associate with 00:0a:79:d4:xx:xx (try 1/3)
[ 2290.929501] wlan0: RX AssocResp from 00:0a:79:d4:xx:xx (capab=0x431
status=0 aid=4)
[ 2290.930005] wlan0: associated
[ 2290.930333] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready
```

アクセスポイントとして利用

『クライアントとして利用』の例で使用したデバイスではUbuntu標準の hostapd ではアクセスポイントとして動作しませんので、アクセスポイントとして動作する [GW-US300MiniS](#) を利用します。



下図のような構成とします。



項目	設定値
SSID	MAE3xx_AP
PSK	password
無線LAN ネットワーク	10.254.0.0/24
DHCP配布アドレス	10.254.0.100 - 10.254.0.200

設定項目としては、

- 無線LAN ネットワークアドレス設定 (/etc/network/interfaces)
- アクセスポイントの設定 (hostapd)
- DHCPサーバDNSサーバの設定 (dnsmasq)
- iptablesの設定²⁾ (NAPT有効化IP FORWARDING有効化)

を設定する必要があります。

無線LAN ネットワークアドレス設定

/etc/network/interfaces に、wlan0 の設定を追加します。

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.103.100
    netmask 255.255.103.0
    gateway 192.168.103.1
    dns-nameservers 192.168.103.1
```

```
auto eth1
iface eth1 inet static
    address 192.168.254.254
    netmask 255.255.255.0

auto wlan0
iface wlan0 inet static
    address 10.254.0.1
    netmask 255.255.255.0
```

アクセスポイントの設定

hostapd.conf の作成

hostapd の設定ファイルを作成します。
パッケージ付属のサンプル設定ファイルを雛形に、下記項目を編集します。

項目	設定値	Note
driver	nl80211	
ssid	MAE3xx_AP	
channel	任意 ³⁾	
ieee80211n	1	
wpa	2	IEEE 802.11i/RSN (WPA2)
wpa_passphrase	password	
wpa_key_mgmt	WPA-PSK	
wpa_pairwise	CCMP	
rsn_pairwise	CCMP	
auth_algs	1	Open System Authentication

編集ができましたら `/etc/hostapd/hostapd.conf` という名前で保存します。

hostapd 起動設定

起動スクリプトに、先ほど編集した設定ファイルの場所をセットします。
編集するファイルは、`/etc/default/hostapd` です。

```
# Defaults for hostapd initscript
#
# See /usr/share/doc/hostapd/README.Debian for information about alternative
# methods of managing hostapd.
#
# Uncomment and set DAEMON_CONF to the absolute path of a hostapd
configuration
```

```
# file and hostapd will be started during system boot. An example
configuration
# file can be found at /usr/share/doc/hostapd/examples/hostapd.conf.gz
#
DAEMON_CONF="/etc/hostapd/hostapd.conf"

# Additional daemon options to be appended to hostapd command:-
#       -d      show more debug messages (-dd for even more)
#       -K      include key data in debug messages
#       -t      include timestamps in some debug messages
#
# Note that -B (daemon mode) and -P (pidfile) options are automatically
# configured by the init.d script and must not be added to DAEMON_OPTS.
#
#DAEMON_OPTS=""
```

DHCPサーバ・DNSサーバの設定

クライアントに IPアドレス, DNSサーバを自動配布するため dnsmasq の設定ファイルを編集します。
ファイルは、/etc/dnsmasq.conf です。

項目	設定値	Note
dhcp-range	10.254.0.100,10.254.0.200,12h	

iptablesの設定 (v2.2.0 以前の場合)

ufw の設定をします。

/etc/default/ufw の編集

```
###
# Set the default forward policy to ACCEPT, DROP or REJECT. Please note
that
# if you change this you will most likely want to adjust your rules
DEFAULT_FORWARD_POLICY="DROP" <---
###
```

“ACCEPT” に変更します。

/etc/ufw/sysctl.conf の編集


```
###
# Uncomment this to allow this host to route packets between interfaces
#net/ipv4/ip_forward=1 <---
###
```

コメントを外します。

/etc/ufw/before.rules の編集

```
###
*nat
-A POSTROUTING -s 10.254.0.0/24 -o eth0 -j MASQUERADE

# don't delete the 'COMMIT' line or these rules won't be processed
COMMIT
```

wlan0 ⇒ eth0 へのNAPTエントリを追加します。

SSH/DNS/DHCP の許可とufwの有効化

SSHのアクセスを許可しておかないとログインができなくなりますので、SSHの許可をしてから ufw を有効化します。
またDNS/DHCPも許可しておきます。

```
user1@plum:~$ sudo ufw allow ssh
[sudo] password for user1:
Rules updated
Rules updated (v6)
user1@plum:~$ sudo ufw allow from any port 68 to any port 67 proto udp
Rule added
Rule added (v6)
user1@plum:~$ ufw allow 53
Rule added
Rule added (v6)
user1@plum:~$
```

```
user1@plum:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)?
y
Firewall is active and enabled on system startup
user1@plum:~$
```

ufw の状態を確認してみます。

```
user1@plum:~$ sudo ufw status
Status: active

To Action From
--
22 ALLOW Anywhere
67/udp ALLOW 68/udp
53 ALLOW Anywhere
22 (v6) ALLOW Anywhere (v6)
67/udp (v6) ALLOW 68/udp (v6)
53 (v6) ALLOW Anywhere (v6)

user1@plum:~$
```

再起動

設定ができましたので、再起動します。

```
user1@plum:~$ sudo overlaycfg -s etc
user1@plum:~$ sudo reboot
user1@plum:~$
Broadcast message from user1@plum
(/dev/pts/1) at 13:44 ...
```

iptablesの設定 (v2.2.0 以降の場合)

v2.2.0 から⁴⁾Firewall のツールを FirewallD に変更しています⁴⁾ので、設定方法が異なります。

zone の設定

wlan0 を “**trusted**” zone へ追加します。

```
user1@plum:~$ sudo firewall-cmd --add-interface=wlan0 --zone=trusted
success
user1@plum:~$ sudo firewall-cmd --add-interface=wlan0 --zone=trusted --
permanent
success
```

eth0 を “**external**” zone へ変更します。

```
user1@plum:~$ sudo firewall-cmd --change-interface=eth0 --zone=external
```

```
success
user1@plum:~$ sudo firewall-cmd --change-interface=eth0 --zone=external --
permanent
success
```

接続テスト

設定がうまくできてるか、クライアントから接続を試みます。



きちんと接続できていることが確認できました。

syslog への出力確認

MA-E3xx側には、次のようなログが出力されます。

```
Apr 17 13:47:38 localhost hostapd: wlan0: STA 00:19:d2:xx:xx:xx IEEE 802.11:
authenticated
Apr 17 13:47:38 localhost hostapd: wlan0: STA 00:19:d2:xx:xx:xx IEEE 802.11:
associated (aid 1)
```

```
Apr 17 13:47:38 localhost hostapd: wlan0: STA 00:19:d2:xx:xx:xx RADIUS:
starting accounting session 534F5C7F-00000000
Apr 17 13:47:38 localhost hostapd: wlan0: STA 00:19:d2:xx:xx:xx WPA:
pairwise key handshake completed (RSN)
Apr 17 13:47:58 localhost dnsmasq-dhcp[1172]: DHCPDISCOVER(wlan0)
10.254.0.133 00:19:d2:xx:xx:xx
Apr 17 13:47:58 localhost dnsmasq-dhcp[1172]: DHCP0FFER(wlan0) 10.254.0.133
00:19:d2:xx:xx:xx
Apr 17 13:48:01 localhost dnsmasq-dhcp[1172]: DHCPDISCOVER(wlan0)
10.254.0.133 00:19:d2:xx:xx:xx
Apr 17 13:48:01 localhost dnsmasq-dhcp[1172]: DHCP0FFER(wlan0) 10.254.0.133
00:19:d2:xx:xx:xx
Apr 17 13:48:01 localhost dnsmasq-dhcp[1172]: DHCPREQUEST(wlan0)
10.254.0.133 00:19:d2:xx:xx:xx
Apr 17 13:48:01 localhost dnsmasq-dhcp[1172]: DHCPACK(wlan0) 10.254.0.133
00:19:d2:xx:xx:xx t60
Apr 17 13:55:51 localhost hostapd: wlan0: STA 00:19:d2:xx:xx:xx WPA: group
key handshake completed (RSN)
```

サンプルコンフィグファイル

今回使用した、サンプルコンフィグファイルを置いておきます。

File	Stat	SHA1SUM	Info
hostapd.conf	2014/04/24 09:18 66.4 KB	a327335da1d5242f188f132d51a1814e6cda4d82	hostapd config
before.rules	2014/04/24 09:22 2.7 KB	c239802f6a45a65b2ef4e400ce72b24d533c3518	ufw config (/etc/ufw/before.rules)
sysctl.conf	2014/04/24 09:22 1.9 KB	d87b2bbdb1dd96a2f0bcd57596a0e2cdbcf85345	ufw config (/etc/ufw/syysctl.conf)
dnsmasq.conf	2014/04/24 09:25 24.7 KB	bb54c8d49c2f16f18d9ea237fe40533e11de2660	dnsmasq config (/etc/dnsmasq.conf)

アクセスポイントとして利用 (Multiple SSID)

ダウンロード

この例で作成したファームウェアファイルはこちらです。

File	Stat	SHA1SUM	Info
mae3xx_trusty_wifi_v2_0_0_beta1.img	2014/04/22 13:45 64.9 MB	dcb4b5bc2ecabe13443dffd1a09820514a7fd8c2	Ubuntu14.04LTS(2014/04/22)

1)

MA-E350-NLW/KLW は、本体に無線 LAN I/F を内蔵していますので、本体のみで本ページの内容が動作します

2)

v2.2.0以前 → ufw, v2.2.0以降 → FirewallD を使用します

3)

空いているchannelを設定してください

4)

[Firewall の設定 \(FirewallD\)](#)

From:

<https://ma-tech.centurysys.jp/> - **MA-X/MA-S/MA-E/IP-K Developers' Wiki**

Permanent link:

https://ma-tech.centurysys.jp/doku.php?id=mae3xx_tips:usb_wifi:start

Last update: **2021/03/17 15:58**