

Wi-Fi - informacje

Położenie: (nie dotyczy)

© 3bird Projects 2022, <http://edukacja.3bird.pl>

Plik konfiguracyjny

Konfiguracja mieści się w pliku `/etc/conf.d/net`:

```
dns_domain_lo="3bird"
# KABLOWY Dom / ZS4 / ZS6 sala 10 / 401 (interface ma 3 adresy IP i trzy bramy, wiec wymagane
sa nawiasy):
# Blokada modulu iproute2 umozliwia stworzenie 2 instancji interfejsu ETH:
modules_enp0s20f0u1="ifconfig netplug !iproute2"
config_enp0s20f0u1=( "192.168.7.3 netmask 255.255.255.0 broadcast 192.168.7.255"
"10.2.6.222 netmask 255.255.255.0 broadcast 10.2.6.255" "192.168.10.222 netmask
255.255.255.0 broadcast 192.168.10.255" )
# Cel rozumiany jako 'default' mozna zamiennie uzywac z adresem 0.0.0.0:
routes_enp0s20f0u1=( "default via 192.168.7.1" "0.0.0.0 via 10.2.6.1" "0.0.0.0 via
192.168.10.1" )
metric_enp0s20f0u1="0"
dns_domain_enp0s20f0u1="3bird"
dns_servers_enp0s20f0u1="1.1.1.1 1.0.0.1"
#mtu_enp0s20f0u1="1500"

modules_enp0s20f0u2="ifconfig netplug !iproute2"
config_enp0s20f0u2=( "192.168.7.3 netmask 255.255.255.0 broadcast 192.168.7.255"
"10.2.6.222 netmask 255.255.255.0 broadcast 10.2.6.255" "192.168.10.222 netmask
255.255.255.0 broadcast 192.168.10.255" )
routes_enp0s20f0u2=( "default via 192.168.7.1" "0.0.0.0 via 10.2.6.1" "0.0.0.0 via
192.168.10.1" )
metric_enp0s20f0u2="0"
dns_domain_enp0s20f0u2="3bird"
dns_servers_enp0s20f0u2="1.1.1.1 1.0.0.1"
#mtu_enp0s20f0u2="1500"

modules_enp0s20f0u3="ifconfig netplug !iproute2"
config_enp0s20f0u3=( "192.168.7.3 netmask 255.255.255.0 broadcast 192.168.7.255"
"10.2.6.222 netmask 255.255.255.0 broadcast 10.2.6.255" "192.168.10.222 netmask
255.255.255.0 broadcast 192.168.10.255" )
routes_enp0s20f0u3=( "default via 192.168.7.1" "0.0.0.0 via 10.2.6.1" "0.0.0.0 via
192.168.10.1" )
metric_enp0s20f0u3="0"
dns_domain_enp0s20f0u3="3bird"
dns_servers_enp0s20f0u3="1.1.1.1 1.0.0.1"
#mtu_enp0s20f0u3="1500"

##### Wi-Fi #####
# Wi-Fi Dom
modules_3bird_5GHz="wpa_supplicant"
# Sterownik wext jest juz przestarzaly (stanowil stara alternatywe wobec WPA2).
wpa_supplicant_3bird_5GHz="-Dnl80211 -iwlp2s0 -B -c/etc/wpa_supplicant/wpa_supplicant.conf"
config_3bird_5GHz="192.168.7.2 netmask 255.255.255.0 broadcast 192.168.7.255"
routes_3bird_5GHz="default via 192.168.7.1"
metric_3bird_5GHz="1"
dns_domain_3bird_5GHz="3bird"
dns_servers_3bird_5GHz="1.1.1.1 1.0.0.1"
#mtu_3bird_5GHz="1500"
```

```
# Wi-Fi ZS4
modules_3bird_Most="dhclient wpa_supplicant"
wpa_supplicant_3bird_Most="-Dnl80211 -iwlp2s0 -B -c/etc/wpa_supplicant/wpa_supplicant.conf"
config_3bird_Most="10.2.6.200 netmask 255.255.255.0 broadcast 10.2.6.255"
routes_3bird_Most="default via 10.2.6.1"
metric_3bird_Most="0"
dns_domain_3bird_Most="ZS4"
dns_servers_3bird_Most="1.1.1.1 1.0.0.1"
```

```
# Wi-Fi dynamiczne:
# Moduły DHCP: dhclient, dhcpcd. Zaleca się korzystanie z dhclient, a nie z dhcpcd:
fallback_wlp2s0="wpa_supplicant dhclient"
fallback_config_wlp2s0="dhcp"
```

```
# Można także skonfigurować sieć na podstawie nazwy SSID:
# config_3bird="192.168.10.190 netmask 255.255.255.0 broadcast 192.168.10.255"
# routes_3bird="default via 192.168.10.1"
# dns_servers_3bird="8.8.8.8 8.8.4.4"
# fallback_3bird="dhcp"
# fallback_routes_3bird="default via 192.168.12.1"
```

Procedura dla laptopa Asus F455L

W jądrze należy uaktywnić moduł *Atheros* (ath). Zawartość pliku */etc/conf.d/net*:

```
# Wi-Fi Dom
modules_wlp3s0="wpa_supplicant ifplugd"
# Sterownik wext jest już przestarzały (to alternatywny sposób obsługi haseł w stosunku do WPA2), mogą pojawić się błędy (loctl[SIOCSIWMODE]: Operation not supported, itp.) więc dajemy nl80211:
wpa_supplicant_wlp3s0="-Dnl80211 -iwlp3s0 -B -c/etc/wpa_supplicant/wpa_supplicant.conf"
...
```

Procedura dla laptopa Acer Aspire 5741G

Karta: Broadcom 802.11 b/g/n, model *BCM43225 [14e4:4357]*. Rodzaj karty sprawdzamy poleceniem:

```
# lspci -vnn -d 14e4:
```

Istnieją różne rodzaje sterowników (modułów) do tej karty, które nawzajem się wykluczają (nie mogą być jednocześnie ładowane; w razie potrzeby należy je zablokować w */etc/modprobe.d/blacklist.conf*).

Informacje ogólne

Karty sieciowe *wi-fi* mogą być podłączone na płycie do następujących złączy:

- **ssb** (*Sonics Silicon Backplane*, autorskie rozwiązanie *Broadcom* przeznaczone dla inteligentnych sieciowych urządzeń SoC; urządzenia te mają własny mostek PCI-SSB; w moim przypadku);
- **bcma** (w moim przypadku);
- **usb**;

Sterownik *ndiswrapper*

Służy do wdrożenia sterowników z *Windows*; jeśli *Linux* jest 64-bitowy, sterowniki *windowsowe* też muszą być 64-bitowe). Aby załadować oryginalny sterownik należy wydać polecenie:

```
# ndiswrapper -i bcmwlhigh6.inf
# ndiswrapper -m
# ndiswrapper -ma
# ndiswrapper -mi
```

modprobe ndiswrapper

To ostatnie polecenie powinno utworzyć urządzenie wlan0. U mnie łąduje moduły, ale nie tworzy urządzenia:

ndiswrapper -l

bcmwl6 : driver installed

device (14e4:4357) present

Nie pomaga także użycie innych sterowników, w tym *bcmwl5*.

Sterownik b43legacy

Znajduje się w jądrze systemu; wspiera tylko karty BCM4306 oraz karty tylko z technologią 802.11b).

Sterownik b43

Zalecany wybór (u mnie działa na jądrze 3.4.9!). Znajdujący się w jądrze moduł o nazwie *b43*, tworzy urządzenie o nazwie *wlp3s0*. Obsługuje mój model karty od jądra 3.1 wzwyż.

Moduł potrzebuje dodatkowej instalacji binarnego zamkniętego firmware od firmy *Broadcom* o nazwie *b43-firmware* dostępnego w portage albo na stronie producenta (więcej info: <http://wireless.kernel.org/en/users/Drivers/b43>). Bez tego dodatkowego wyciętego kodu (*fwcutter* wycina fragment kodu → sprawa licencji), nie jest w stanie działać.

Aby system stworzył działające urządzenie sieciowe, musi najpierw utworzyć następujące moduły: *b43*, *ssb*, *bcma*, *mac80211*, *cfg80211*, *rkill*, *acer_wmi* (*Windows Management Interface*). Aby wyprodukować te moduły, należy w jądrze uaktywnić:

Networking support:

<M> *RF switch subsystem support*

Networking support / Wireless:

<M> *cfg80211 - wireless configuration API*

[*] *enable powersave by default*

[*] *cfg80211 wireless extensions compatibility*

[*] *Wireless extensions sysfs files*

<M> *Common routines for IEEE802.11 drivers*

[*] *lib80211 debugging messages*

<M> *Generic IEEE 802.11 Networking stack (mac80211)*

[*] *Enable mac80211 mesh networking (pre-802.11s) support*

[*] *Enable LED triggers*

Device Drivers / Network device support:

{M} *PHY Device support and infrastructure*

<M> *Drivers for Broadcom PHYs*

Device Drivers / Network device support / Wireless LAN:

<M> *Broadcom 43xx wireless support (mac80211 stack) - (to jest moduł b43)*

[*] *Support for BCMA bus*

[*] *Hardware support that overlaps with the brcmsmac driver*

[*] *Support for 802.11n (N-PHY) devices (EXPERIMENTAL)*

[*] *Support for low-power (LP-PHY) devices*

[*] *Support for HT-PHY (high throughput) devices (EXPERIMENTAL)*

[*] *Broadcom 43xx debugging*

Device Drivers / Input device support:

<M> *Polled input device skeleton*

Device Drivers / LED support:

{*} *LED class support*

-*- *LED Trigger support*

<*> *LED Default ON Trigger*

Device Drivers / x86 Platform specific Device Drivers:

<M> *Acer WMI Laptop Extras*

<*> WMI

Sprawdzamy ewentualne błędy:

```
# dmesg | grep b43
```

Wynik w przypadku błędu:

```
b43-phy0: Broadcom 43225 WLAN found (core revision 23)
b43-phy0 ERROR: FOUND UNSUPPORTED PHY (Analog 8, Type 4, Revision 6)
b43: probe of bcma0:0 failed with error -95
```

Wynik w przypadku powodzenia:

```
b43-phy0: Broadcom 43225 WLAN found (core revision 23)
b43-phy0 debug: Found PHY: Analog 8, Type 4, Revision 6
b43-phy0 debug: Found Radio: Manuf 0x17F, Version 0x2056, Revision 11
b43-phy0: Loading firmware version 666.2 (2011-02-23 01:15:07)
b43-phy0 debug: Chip initialized
b43-phy0 debug: 64-bit DMA initialized
b43-phy0 debug: QoS enabled
b43-phy0 debug: Wireless interface started
b43-phy0 debug: Adding Interface type 2
b43-phy0 debug: Using hardware based encryption for keyidx: 0, mac: f8:d1:11:4d:29:58
b43-phy0 debug: RX: Packet size underrun (2)
b43-phy0 debug: Disabling hardware based encryption for keyidx: 0, mac: f8:d1:11:4d:29:58
b43-phy0 debug: Using hardware based encryption for keyidx: 0, mac: f8:d1:11:4d:29:58
```

Sterownik **broadcom-sta**

Jest to pakiet binarny, zamknięty, zawiera moduł o nazwie **wl**, ale najpierw musi być załadowany moduł bezpieczeństwa/szyfrowania, tj. *cfg80211* lub alternatywnie *lib80211* lub *ieee80211_crypt_tkip*). Jeśli po restarcie nie pojawia się urządzenie odpowiadające za kartę wi-fi (ma ono mylącą postać *eth**), należy spróbować wydać polecenie:

```
# insmod wl.ko
```

Należy także zauważyć, że ostatnio firma Broadcom zmieniła nazwę tego pakietu na *hybrid-port-src_x86_64-*.tar.gz*. Wielu użytkowników zauważa, że binarny sterownik firmy Broadcom nie działa na nowszych kernelach; należy także pamiętać o tym, aby moduły *ssb* / *bcma* / *b43* / *brcmsmac* / *acer_wmi* nie były ładowane, gdyż powodują konflikt z *wl*.

Sterownik **brcmsmac**

Znajduje się w jądrze systemu (inna nazwa: **brcm80211**) i zawiera moduł **brcmsmac**. Teoretycznie obsługuje moją kartę zarówno w systemach 32/64-bitowym. W tej opcji należy także zainstalować *linux-firmware* (z odpowiednią opcją) lub skopiować *brcm/bcm43xx-0.fw* oraz *brcm/bcm43xx_hdr-0.fw* do folderu */lib/firmware/brcm*.

Aby wyprodukować moduł *brcmsmac* należy uaktywnić w jądrze:

```
Device Drivers \ Network device support \ Wireless LAN:
```

```
<M> Broadcom IEEE802.11n PCIe SoftMAC WLAN driver (moduł brcmsmac)
< > Broadcom IEEE802.11n embedded FullMAC WLAN driver (moduł brcmfmac)
[*] SDIO bus interface support for FullMAC driver
[*] USB bus interface support for FullMAC driver
```

oraz moduły do obsługi złącza:

```
Device Drivers \ Broadcom specific AMBA:
```

```
<M> BCMA support
[*] Support for BCMA on PCI-host bus
```

Należy pamiętać, aby w jądrze uaktywnić ogólne wsparcie dla wireless: *Networking support* / *Wireless...* (m.in. *cfg80211* oraz *mac80211*).

Konfiguracja **wpa_supplicant**

W pliku */etc/conf.d/net* wpisujemy:

```
modules_wlp2s0="wpa_supplicant ifplugd"
wpa_supplicant_wlp2s0="-Dwext -iwlp2s0 -B -c/etc/wpa_supplicant/wpa_supplicant.conf"
```

...

zaś w pliku */etc/wpa_supplicant/wpa_supplicant.conf* wpisujemy:

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=wheel
network={
    ssid="3bird"
    scan_ssid=1
    key_mgmt=WPA-PSK
    psk="MojeTajneHaslo"
}
```

Dodatkowe narzędzia i polecenia

Do działania *wi-fi* potrzebne będą także następujące pakiety:

ifplugd - ustawia domyślny routing po wpięciu kabla eth0 i po jego wypięciu;

wpa_supplicant - obsługa Wi-Fi, zapisuje sieci i hasła do nich;

wpa_gui - graficzny applet z pakietu *wpa_supplicant* pozwalający na łatwe dodawanie i usuwanie sieci (jest to graficzna nakładka na *wpa_cli* i pracuje na pliku */etc/wpa_supplicant/wpa_supplicant.conf*); konfigurację sieci może przeprowadzić każdy użytkownik dodany do grupy „*wheel*” pod warunkiem, że w pliku */etc/wpa_supplicant/wpa_supplicant.conf* znajduje się wpis „*update_config=1*”; aby applet był ładowany przy starcie *icewm*, należy dodać go do skryptu (wykonywalnego) *~/icewm/startup*:

```
#!/bin/bash
dbus-launch /usr/bin/wpa_gui -t &
```

wireless-tools - zawiera polecenie *iwconfig*;

rkill - włącza / wyłącza diodę *wifi* (jeśli taką mamy na obudowie laptopa);

```
# iwconfig - pokazuje do jakiej sieci jesteśmy obecnie połączeni (i inne informacje)
# iwlist wlan0 scan (wykrywa sieci wifi działające w pobliżu)
# ifplugstatus - pokazuje czy kabel jest wpięty czy nie
# /etc/init.d/net.eth0 status (w wyniku działania ifplugd, net.eth0 ma po starcie status „inactive”, a nie „stopped”; jest to konieczne, gdyż inaczej interface nie wykryje włożonej do portu wtyczki)
# traceroute 8.8.8.8 - można sprawdzić jaki jest routing po włożeniu wtyczki i po jej wyjęciu
# tail -f /var/log/kern.log oraz wcisnąć klawisze Fn+F3 - sprawdzamy, jaką akcję generuje wcisnięcie tych klawiszy
# iwconfig wlan0 txpower on - służy do obniżenia zużycia energii (włącza funkcję Power Management, czyli dynamicznego zarządzania energią wifi), ale jeśli karta często wyłącza się, należy tę opcję zmienić na off; inne możliwe wartości: auto, 0, 15, 20 (mW - miliWatt); u mnie obniżenie wartości z 20 na 0 poskutkowało (o paradoksie) szybszym przesyłem danych i większą stabilnością łącza (prawdopodobnie chodzi o zlikwidowanie odbić i zniekształceń); pomagają także wyłączenie wi-fi w extenderze (jego emisja zakłócała emisję routera - fale nawzajem wchodziły na siebie lub znosiły).
# rkill list (musi być załadowany moduł rkill) - wykaz urządzeń sieciowych, które mogą być blokowane (wyłączane) albo fizycznie, albo programowo:
0: acer-wireless: Wireless LAN
   Soft blocked: no
   Hard blocked: no
1: phy0: Wireless LAN
   Soft blocked: no
   Hard blocked: no
# rkill unblock all - odblokowuje wszystkie urządzenia sieciowe (zapala / gasi diodę odpowiedzialną za aktywność wifi); zazwyczaj robi to polecenie /etc/init.d/net.wlp3s0 start/stop.
```

Alternatywnie: NetworkManager

Alternatywnie (wobec *wpa_supplicant*), sieć (szczególnie *Wi-Fi*) można skonfigurować za pomocą *NetworkManager*:

```
# emerge -vp networkmanager
# rc-update add NetworkManager default
```

Należy także dodać globalnie flagę „networkmanager” do `/etc/portage/make.conf` oraz:

```
# emerge -vp -uDN @world
```

```
# emerge -vp dhcp
```

```
# emerge -vp nm-applet
```

Aby applet uruchamiany był automatycznie, należy dodać do `~/.xinitrc` lub do `~/.icewm/startup` następujący wpis:

```
#!/bin/bash
```

```
/usr/bin/pktttyagent &
```

```
sleep 1 && /usr/libexec/polkit-gnome-authentication-agent-1 &
```

```
sleep 1 && dbus-launch nm-applet &
```

Uwaga: Aby zwykły użytkownik miał prawo zarządzać siecią, należy dodać go do grupy „plugdev”.

W pliku `/etc/dhcp/dhclient.conf` określamy nazwę hosta:

```
send host-name „asus-i3”;
```

W pliku `/etc/rc.conf`:

```
rc_hotplug="!net.* *"
```

Zarządzanie w trybie tekstowym:

```
# nmcli dev (wykaz dostępnych interfejsów sieciowych i aktywnych połączeń)
```

```
# nmcli dev disconnect iface wlp3s0 (przerwanie połączenia z danym interfacem)
```

```
# nmtui (konfiguracja połączeń w trybie tekstowym)
```

Główny plik konfiguracyjny nie istnieje, gdyż serwis działa dobrze na domyślnych parametrach. Ale jakby ktoś chciał, może je zmienić tworząc plik `/etc/NetworkManager/NetworkManager.conf`. Oto przykładowy plik konfiguracyjny:

[main]

```
# Program próbuje zapisać połączenia używając kolejno wymienionych pluginów (najpierw pierwszy, potem drugi, aż do skutku). Plugin "keyfile" powinien być zawsze na końcu tej listy.
```

```
# Lista pluginów:
```

```
# keyfile (domyślny, zapisuje dane w /etc/NetworkManager/system-connections/*; ignoruje zapisywanie do plików, które mają inne prawa niż root.root rw-----);
```

```
# ifcfg-rh (czyta i zapisuje w /etc/sysconfig/network-scripts/ifcfg-*),
```

```
# ifcfg-suse (tylko czyta konfigurację w SUSE, ale nie potrafi jej zapisywać),
```

```
# ifupdown (tylko odczytuje konfigurację sieciową na Debianie/Ubuntu z /etc/network/interfaces),
```

```
# ibft (odczytuje iSCSI Boot Firmware Table),
```

```
# ifnet (odczytuje i zapisuje konfigurację sieciową w Gentoo do /etc/conf.d/net)
```

```
plugins=ifnet,keyfile
```

```
# Program monitoruje wszelkie zmiany w plikach połączeń i wczytuje je w locie, bez restartu.
```

```
# Może to być uciążliwe, gdy wprowadzimy częściowe zmiany, dlatego domyślnie jest "false" (używamy wtedy "nmcli con load");
```

```
monitor-connection-files=false
```

```
# Umożliwia wprowadzanie zmian do połączeń przez zwykłych użytkowników (wartość "true").
```

```
# Może przydzielać dostęp do danych interface'ów dla wybranych użytkowników
```

```
auth-polkit=true
```

```
# Jaki klient DHCP będzie używany (dhclient, dhcpcd, internal):
```

```
dhcp=dhclient
```

```
# Dla jakich interface'ów kablowych, program ma nie tworzyć autokonfiguracji (MAC, eth*, *)
```

```
no-auto-default=*
```

```
# Program nie będzie zważał na odłączony kabel i będzie inicjował oraz podtrzymywał połączenie dla poniższych interface'ów (można stosować *):
```

```
# ignore-carrier=enp2s0
```

```
# Jeśli "true", pliki konfiguracyjne będą mogły być zmieniane przez inne programy, gdy Network-
```

Manager przestanie działać:

```
configure-and-quit=false
```

Czy NetworkManager może nadpisywać resolv.conf? ("default" na tak; "none")

```
dns=default
```

[keyfile]

```
hostname=asus-i3
```

Interface'y ignorowane, gdy używany jest plugin keyfile (MAC, eth, *):*

```
# unmanaged-devices=interface-name:eth0;mac:00:99:76:...
```

UWAGA: Plugin "ifnet" został ostatnio (2015 rok) usunięty z programu, gdyż nie był rozwijany.

[ifnet]

Jeśli "true", wtedy NetworkManager ma prawo zarządzać sieciowymi interface'ami w /etc/conf.d/net. Jeśli jest "false", wtedy te interface'y będą ignorowane.

```
managed=true
```

```
auto_refresh=false
```

[connectivity]

Sprawdzanie, czy cały czas jest połączenie z Internetem. Czasowo sprawdza dostępność poniższej strony i ustawia status "NetworkManager is online":

```
uri=http://3bird.net
```

Co ile sekund sprawdzane jest połączenie:

```
interval=300
```

Globalna konfiguracja połączeń z poszczególnymi sieciami znajduje się w `/etc/NetworkManager/system-connections/*`. Przykład jednego z połączeń:

[connection]

```
id=3bird
```

```
uuid=4d10e3d9-8506-4633-b543-8bacd3d499da
```

```
type=wifi
```

```
permissions=
```

```
secondaries=
```

```
timestamp=1440287985
```

[wifi]

```
mac-address=30:40:B3:82:B2:0A
```

```
mac-address-blacklist=
```

```
mode=infrastructure
```

```
mtu=1500
```

```
seen-bssids=
```

```
ssid=3bird
```

[wifi-security]

```
group=
```

```
key-mgmt=wpa-psk
```

```
pairwise=
```

```
proto=
```

```
psk=MojeHaslo
```

[ipv4]

```
address1=192.168.0.2/24,192.168.0.1
```

```
dns=8.8.8.8;8.8.4.4;
```

```
dns-search=3bird;
```

```
method=manual
```

```
route1=192.168.0.0/24,192.168.0.1,0
```

[ipv6]

```
dns-search=
```

```
ip6-privacy=0
```

`method=auto`

Konfiguracja poszczególnych użytkowników zachowana jest tutaj:
`~/.gconf/apps/nm-applet`

Uwaga: W chwili obecnej program wydaje się niekompatybilny z Gentoo. System przy starcie automatycznie inicjuje interface'y sieciowe z `/etc/conf.d/net` i powoduje konflikt z NetworkManagerem.

Problemy

Program `wpa_gui` wykrywa drugi nieistniejący adapter o nazwie "`p2p-dev-wlp2s0`".

Rozwiązanie: Należy w pliku `/etc/wpa_supplicant/wpa_supplicant.conf` dodać:

`p2p_disabled=1`

Po aktualizacji zmieniły się nazwy interfejsów sieciowych, np. zamiast `wlp2s0` jest teraz `wlan0`.

Rozwiązanie:

`emerge --unmerge iwd`

Uwaga: Należy rozważyć jednak zastąpienie `wpa_supplicant` pakietem `iwd` (rozwijany przez Intel), gdyż jest nowszy i mniej zasobożerny.

Ostatnia aktualizacja: 1 listopada 2022.