## NAME

**rsu** - Realtek RTL8188SU/RTL8192SU USB IEEE 802.11b/g/n wireless network driver

## SYNOPSIS

To compile this driver into the kernel, place the following lines in your kernel configuration file:

```
device ehci
device uhci
device ohci
device usb
device rsu
device rsufw
device wlan
```

Alternatively, to load the driver as a module at boot time, place the following lines in loader.conf(5):

```
if_rsu_load="YES"
rsu-rtl8712fw_load="YES"
```

## DESCRIPTION

The **rsu** driver supports USB 2.0 wireless network devices based on Realtek RTL8188SU, RTL8191SU and RTL8192SU chipsets.

The RTL8188SU is a highly integrated 802.11n adapter that combines a MAC, a 1T1R capable baseband and an RF in a single chip. It operates in the 2GHz spectrum only.

The RTL8191SU is a highly integrated multiple-in, single-out (MISO) 802.11n adapter that combines a MAC, a 1T2R capable baseband and an RF in a single chip. It operates in the 2GHz spectrum only.

The RTL8192SU is a highly integrated multiple-in, multiple-out (MIMO) 802.11n adapter that combines a MAC, a 2T2R capable baseband and an RF in a single chip. It operates in the 2GHz spectrum only.

These are the modes the **rsu** driver can operate in:

BSS mode — Also known as *infrastructure* mode, this is used when associating with an access point, through which all traffic passes. This mode is the default.

monitor mode — In this mode the driver is able to receive packets without associating with an access point. This disables the internal receive filter and enables the card to capture packets

from networks which it wouldn’t normally have access to, or to scan for access points.

The **rsu** driver can be configured to use Wired Equivalent Privacy (WEP) or Wi-Fi Protected Access (WPA-PSK and WPA2-PSK). WPA is the de facto encryption standard for wireless networks. It is strongly recommended that WEP not be used as the sole mechanism to secure wireless communication, due to serious weaknesses in it.

The **rsu** driver can be configured at runtime with ifconfig(8).

## HARDWARE

The **rsu** driver provides support for Realtek RTL8188SU/RTL8192SU USB IEEE 802.11b/g/n wireless network adapters, including:

- ASUS USB-N10
- ASUS WL-167G V3
- Belkin F7D1101 v1
- D-Link DWA-131 A1
- EDUP EP-MS150N(W)
- Edimax EW-7622UMN
- Hercules HWGUn-54
- Hercules HWNUm-300
- Planex GW-USNano
- Sitecom WL-349 v1
- Sitecom WL-353
- Sweex LW154
- TRENDnet TEW-646UBH
- TRENDnet TEW-648UB
- TRENDnet TEW-649UB

## FILES

*/usr/share/doc/legal/realtek.LICENSE* **rsu** firmware license

The driver needs at least version 1.2 of the following firmware file, which is loaded when an interface is attached:

*/boot/kernel/rsu-rtl8712fw.ko*

## EXAMPLES

Join an existing BSS network (i.e., connect to an access point):

```
ifconfig wlan create wlandev rsu0 inet 192.0.2.20/24
```

Join a specific BSS network with network name *my_net*:

```
ifconfig wlan create wlandev rsu0 ssid my_net up
```

Join a specific BSS network with 64-bit WEP encryption:

```
ifconfig wlan create wlandev rsu0 ssid my_net \
  wepmode on wepkey 0x1234567890 weptxkey 1 up
```

## DIAGNOSTICS

**%s: failed load firmware of file rsu-rtl8712fw** For some reason, the driver was unable to read the microcode file from the filesystem. The file might be missing or corrupted.

**device timeout** A frame dispatched to the hardware for transmission did not complete in time. The driver will reset the hardware. This should not happen.

## SEE ALSO

intro(1), netintro(4), rsufw(4), usb(4), wlan(4), networking(7), arp(8), hostapd(8), ifconfig(8), wpa_supplicant(8)

## HISTORY

The **rsu** driver first appeared in OpenBSD 4.9 and FreeBSD 10.0.

## AUTHORS

The **rsu** driver was written by Damien Bergamini <*damien@openbsd.org*> and ported by Rui Paulo <*rpaulo@freebsd.org*>. The 802.11n support was added by Adrian Chadd <*adrian@freebsd.org*>.

## CAVEATS

The **rsu** driver currently does not support 802.11n transmit aggregation, either A-MSDU or A-MPDU.

The **rsu** driver does not capture management frames in non-monitor modes; without this limitation some firmware functions (e.g., ’join bss’) will not work properly.