RT-Preempt patch Linux

EtienneAr

April 2019

This document will present what I did to install (patch) real-time Linux.

1 Ubuntu 16.04

1.1 Download and install Ubuntu

I did the install, by downloading Ubuntu 16, and creating a live USB stick. Then I followed the Live USB instructions.

```
wget http://releases.ubuntu.com/16.04/ubuntu-16.04.6-desktop-amd64.iso
```

```
#Choose the correct device which will be the output (a.k.a 'of') to create
   Live USB.
lsblk
```

```
sudo dd if=ubuntu-16.04.6-desktop-amd64.iso of=/dev/sdb status=progress &&
   sync
```

1.2 (optional) Setup ssh connection

```
# On the freshly installed machine
sudo apt-get install openssh-server
sudo service ssh start
```

After this step everything can be done remotely :

```
# On another machine (connected to the same network ...)
ssh login@ip
```

From now on, everything will be done on the freshly installed machine (either by typing physically on it, or by using a ssh conneciton).

1.3 Downgrading to kernel 4.14

```
wget https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.14.109/linux-headers
   -4.14.109-0414109_4.14.109-0414109.201903271718_all.deb
wget https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.14.109/linux-headers
   -4.14.109-0414109-generic_4.14.109-0414109.201903271718_amd64.deb
wget https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.14.109/linux-image-
```

```
    unsigned-4.14.109-0414109-generic_4.14.109-0414109.201903271718_amd64.
    deb
wget https://kernel.ubuntu.com/~kernel-ppa/mainline/v4.14.109/linux-modules
    -4.14.109-0414109-generic_4.14.109-0414109.201903271718_amd64.deb


sudo dpkg -i linux-headers-4.14.109-0414109_4.14.109-0414109.201903271718
    _all.deb
sudo dpkg -i linux-headers-4.14.109-0414109-generic_4
    .14.109-0414109.201903271718_amd64.deb
sudo dpkg -i linux-modules-4.14.109-0414109-generic_4
    .14.109-0414109.201903271718_amd64.deb
sudo dpkg -i linux-image-unsigned-4.14.109-0414109-generic_4
    .14.109-0414109.201903271718_amd64.deb
```

1.4 Setup grub :

Grub can let you select (at boot time) which kernel you want to use. It can be very useful if you want to try several kernels, patched and not patched. If it is not setup correctly, here is how you can do it :

```
# Modify grub to boot on one kernel automatically
sudo vim /etc/default/grub
   # Modify GRUB_DEFAULT=0 to GRUB_DEFAULT="Advanced options for Ubuntu>
     Ubuntu, with Linux 4.14.109-0414109-generic"
      # The second part (after '>') can be found in the menu entry.
      # /!\ To do after a first sudo update-grub :
         grep menuentry /boot/grub/grub.cfg

   #If grub does not appear at boot time, you need to :
   # place "#" symbol at the start of line GRUB_HIDDEN_TIMEOUT=0

sudo update-grub
```

```
sudo reboot
```

1.5 (optional) Get the system info (before changing the kernel too much)

```
mkdir info
cd info

uname -a > uname.txt
sudo cat /proc/cpuinfo > cpuinfo.txt
lspci > lspci.txt
```

1.6 (optional/Recommended) Test real-time capabilities

This software will test how your OS handles real time. You should launch it, run some other app while it's running and save the result to compare it later after patching your OS.

```
git clone git://git.kernel.org/pub/scm/utils/rt-tests/rt-tests.git
cd rt-tests
git checkout stable/v1.0

sudo apt-get install build-essential libnuma-dev

make
sudo ./cyclictest -a -t -n -p99
```

1.7 Get kernel and patch source code

```
sudo passwd root
su

mkdir -p /usr/src/kernels
cd /usr/src/kernels

wget http://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.14.109.tar.gz
wget https://mirrors.edge.kernel.org/pub/linux/kernel/projects/rt/4.4/patch
    -4.14.109-rt57.patch.xz
```

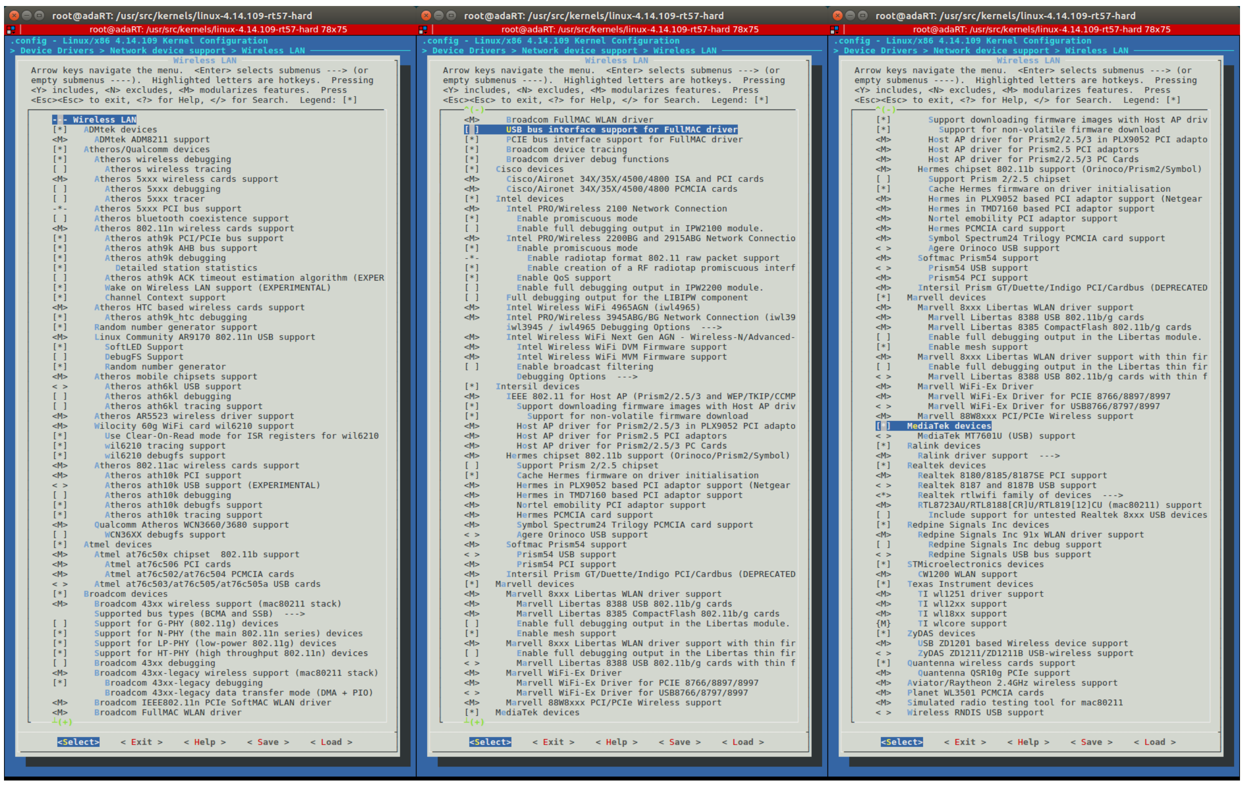
1.8 Extract code and apply patch

```
tar xf linux-4.14.109.tar.xz
mv linux-4.14.109 linux-4.4.177-rt57
cd linux-4.14.109-rt57

xz -d ../patch-4.14.109-rt57.patch.xz
patch -p1 <../patch-4.14.109-rt57.patch

cp /boot/config-4.15.x-xxx-generic .config
```

```
### You may need this to make the config menu
apt-get install libncurses5-dev
```

```
# 'minimum' configuration
make defconfig

# configure the kernel
make menuconfig
# The main setting to change is :
# * Processor type and features -> Preemption Model : Fully Preemptible
    Kernel (RT)
# * ACPI -> (disable) Hibernation
# * Networking -> (disable) Radio Amateur
# * Networking > Wireless -> (disable) powersave
# * Device drivers -> (disable) Macintosh
# * Device drivers > Network > Ethernet driver support -> (disable) every
```

```
    unecessary brand
# * Device drivers > Network > Wireless lan (see list)
# * Device drivers > Network -> (disable) USB
# ... Better settings can be done ...
```



(a)

```
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard 78x75
.config - Linux/x86 4.14.109 Kernel Configuration
> Device Drivers > Network device support > Wireless LAN
                              Wireless LAN
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or
  empty submenus ----).  Highlighted letters are hotkeys.  Pressing
  <Y> includes, <N> excludes, <M> modularizes features.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*]

      --- Wireless LAN
      [*]   ADMtek devices
      <M>     ADMtek ADM8211 support
      [*]   Atheros/Qualcomm devices
      [*]     Atheros wireless debugging
      [ ]       Atheros wireless tracing
      <M>     Atheros 5xxx wireless cards support
      [ ]       Atheros 5xxx debugging
      [ ]       Atheros 5xxx tracer
      -*-     Atheros 5xxx PCI bus support
      [ ]     Atheros bluetooth coexistence support
      <M>     Atheros 802.11n wireless cards support
      [*]       Atheros ath9k PCI/PCIe bus support
      [*]       Atheros ath9k AHB bus support
      [*]       Atheros ath9k debugging
      [*]         Detailed station statistics
      [ ]       Atheros ath9k ACK timeout estimation algorithm (EXPER
      [*]       Wake on Wireless LAN support (EXPERIMENTAL)
      [*]       Channel Context support
      <M>     Atheros HTC based wireless cards support
      [*]       Atheros ath9k_htc debugging
      [*]     Random number generator support
      <M>     Linux Community AR9170 802.11n USB support
      [*]       SoftLED Support
      [ ]       DebugFS Support
      [*]       Random number generator
      <M>     Atheros mobile chipsets support
      < >       Atheros ath6kl USB support
      [ ]       Atheros ath6kl debugging
      [ ]       Atheros ath6kl tracing support
      <M>     Atheros AR5523 wireless driver support
      <M>     Wilocity 60g WiFi card wil6210 support
      [*]       Use Clear-On-Read mode for ISR registers for wil6210
      [*]       wil6210 tracing support
      [*]       wil6210 debugfs support
      <M>     Atheros 802.11ac wireless cards support
      <M>       Atheros ath10k PCI support
      < >       Atheros ath10k USB support (EXPERIMENTAL)
      [ ]       Atheros ath10k debugging
      [*]       Atheros ath10k debugfs support
      [*]       Atheros ath10k tracing support
      <M>     Qualcomm Atheros WCN3660/3680 support
      [ ]       WCN36XX debugfs support
      [*]   Atmel devices
      <M>     Atmel at76c50x chipset  802.11b support
      <M>       Atmel at76c506 PCI cards
      <M>       Atmel at76c502/at76c504 PCMCIA cards
      < >     Atmel at76c503/at76c505/at76c505a USB cards
      [*]   Broadcom devices
      <M>     Broadcom 43xx wireless support (mac80211 stack)
              Supported bus types (BCMA and SSB)  --->
      [ ]     Support for G-PHY (802.11g) devices
      [*]     Support for N-PHY (the main 802.11n series) devices
      [*]     Support for LP-PHY (low-power 802.11g) devices
      [*]     Support for HT-PHY (high throughput 802.11n) devices
      [ ]     Broadcom 43xx debugging
      <M>     Broadcom 43xx-legacy wireless support (mac80211 stack)
      [*]       Broadcom 43xx-legacy debugging
                Broadcom 43xx-legacy data transfer mode (DMA + PIO)
      <M>     Broadcom IEEE802.11n PCIe SoftMAC WLAN driver
      <M>     Broadcom FullMAC WLAN driver
      ┴(+)

        <Select>    < Exit >    < Help >    < Save >    < Load >
```

(b)

```
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard 78x75
.config - Linux/x86 4.14.109 Kernel Configuration
> Device Drivers > Network device support > Wireless LAN
                              Wireless LAN
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or
  empty submenus ----).  Highlighted letters are hotkeys.  Pressing
  <Y> includes, <N> excludes, <M> modularizes features.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*]
     ^(-)
      <M>       Broadcom FullMAC WLAN driver
      [ ]       USB bus interface support for FullMAC driver
      [*]       PCIE bus interface support for FullMAC driver
      [*]       Broadcom device tracing
      [*]       Broadcom driver debug functions
      [*]     Cisco devices
      <M>       Cisco/Aironet 34X/35X/4500/4800 ISA and PCI cards
      <M>       Cisco/Aironet 34X/35X/4500/4800 PCMCIA cards
      [*]     Intel devices
      <M>       Intel PRO/Wireless 2100 Network Connection
      [*]         Enable promiscuous mode
      [ ]         Enable full debugging output in IPW2100 module.
      <M>       Intel PRO/Wireless 2200BG and 2915ABG Network Connectio
      [*]         Enable promiscuous mode
      -*-           Enable radiotap format 802.11 raw packet support
      [*]           Enable creation of a RF radiotap promiscuous interf
      [*]         Enable QoS support
      [ ]         Enable full debugging output in IPW2200 module.
      [ ]       Full debugging output for the LIBIPW component
      <M>       Intel Wireless WiFi 4965AGN (iwl4965)
      <M>       Intel PRO/Wireless 3945ABG/BG Network Connection (iwl39
                iwl3945 / iwl4965 Debugging Options  --->
      <M>       Intel Wireless WiFi Next Gen AGN - Wireless-N/Advanced-
      <M>         Intel Wireless WiFi DVM Firmware support
      <M>         Intel Wireless WiFi MVM Firmware support
      [ ]         Enable broadcast filtering
                  Debugging Options  --->
      [*]     Intersil devices
      <M>       IEEE 802.11 for Host AP (Prism2/2.5/3 and WEP/TKIP/CCMP
      [*]         Support downloading firmware images with Host AP driv
      [*]           Support for non-volatile firmware download
      <M>         Host AP driver for Prism2/2.5/3 in PLX9052 PCI adapto
      <M>         Host AP driver for Prism2.5 PCI adaptors
      <M>         Host AP driver for Prism2/2.5/3 PC Cards
      <M>       Hermes chipset 802.11b support (Orinoco/Prism2/Symbol)
      [ ]         Support Prism 2/2.5 chipset
      [*]         Cache Hermes firmware on driver initialisation
      <M>         Hermes in PLX9052 based PCI adaptor support (Netgear
      <M>         Hermes in TMD7160 based PCI adaptor support
      <M>         Nortel emobility PCI adaptor support
      <M>         Hermes PCMCIA card support
      <M>         Symbol Spectrum24 Trilogy PCMCIA card support
      < >         Agere Orinoco USB support
      <M>       Softmac Prism54 support
      < >         Prism54 USB support
      <M>         Prism54 PCI support
      <M>       Intersil Prism GT/Duette/Indigo PCI/Cardbus (DEPRECATED
      [*]     Marvell devices
      <M>       Marvell 8xxx Libertas WLAN driver support
      <M>         Marvell Libertas 8388 USB 802.11b/g cards
      <M>         Marvell Libertas 8385 CompactFlash 802.11b/g cards
      [ ]         Enable full debugging output in the Libertas module.
      [*]         Enable mesh support
      <M>       Marvell 8xxx Libertas WLAN driver support with thin fir
      [ ]         Enable full debugging output in the Libertas thin fir
      < >         Marvell Libertas 8388 USB 802.11b/g cards with thin f
      <M>       Marvell WiFi-Ex Driver
      <M>         Marvell WiFi-Ex Driver for PCIE 8766/8897/8997
      < >         Marvell WiFi-Ex Driver for USB8766/8797/8997
      <M>       Marvell 88W8xxx PCI/PCIe Wireless support
      [*]     MediaTek devices
      ┴(+)

        <Select>    < Exit >    < Help >    < Save >    < Load >
```

(c)

```
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard
root@adaRT: /usr/src/kernels/linux-4.14.109-rt57-hard 78x75
.config - Linux/x86 4.14.109 Kernel Configuration
> Device Drivers > Network device support > Wireless LAN
                              Wireless LAN
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or
  empty submenus ----).  Highlighted letters are hotkeys.  Pressing
  <Y> includes, <N> excludes, <M> modularizes features.  Press
  <Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*]
     ^(-)
      [*]         Support downloading firmware images with Host AP driv
      [*]           Support for non-volatile firmware download
      <M>         Host AP driver for Prism2/2.5/3 in PLX9052 PCI adapto
      <M>         Host AP driver for Prism2.5 PCI adaptors
      <M>         Host AP driver for Prism2/2.5/3 PC Cards
      <M>       Hermes chipset 802.11b support (Orinoco/Prism2/Symbol)
      [ ]         Support Prism 2/2.5 chipset
      [*]         Cache Hermes firmware on driver initialisation
      <M>         Hermes in PLX9052 based PCI adaptor support (Netgear
      <M>         Hermes in TMD7160 based PCI adaptor support
      <M>         Nortel emobility PCI adaptor support
      <M>         Hermes PCMCIA card support
      <M>         Symbol Spectrum24 Trilogy PCMCIA card support
      < >         Agere Orinoco USB support
      <M>       Softmac Prism54 support
      < >         Prism54 USB support
      <M>         Prism54 PCI support
      <M>       Intersil Prism GT/Duette/Indigo PCI/Cardbus (DEPRECATED
      [*]     Marvell devices
      <M>       Marvell 8xxx Libertas WLAN driver support
      <M>         Marvell Libertas 8388 USB 802.11b/g cards
      <M>         Marvell Libertas 8385 CompactFlash 802.11b/g cards
      [ ]         Enable full debugging output in the Libertas module.
      [*]         Enable mesh support
      <M>       Marvell 8xxx Libertas WLAN driver support with thin fir
      [ ]         Enable full debugging output in the Libertas thin fir
      < >         Marvell Libertas 8388 USB 802.11b/g cards with thin f
      <M>       Marvell WiFi-Ex Driver
      <M>         Marvell WiFi-Ex Driver for PCIE 8766/8897/8997
      < >         Marvell WiFi-Ex Driver for USB8766/8797/8997
      <M>       Marvell 88W8xxx PCI/PCIe Wireless support
      [*]     MediaTek devices
      < >       MediaTek MT7601U (USB) support
      [*]     Ralink devices
      <M>       Ralink driver support  --->
      [*]     Realtek devices
      <M>       Realtek 8180/8185/8187SE PCI support
      < >       Realtek 8187 and 8187B USB support
      <*>       Realtek rtlwifi family of devices  --->
      <M>       RTL8723AU/RTL8188[CR]U/RTL819[12]CU (mac80211) support
      [ ]         Include support for untested Realtek 8xxx USB devices
      [*]     Redpine Signals Inc devices
      <M>       Redpine Signals Inc 91x WLAN driver support
      [ ]         Redpine Signals Inc debug support
      < >         Redpine Signals USB bus support
      [*]     STMicroelectronics devices
      <M>       CW1200 WLAN support
      [*]     Texas Instrument devices
      <M>       TI wl1251 driver support
      <M>       TI wl12xx support
      <M>       TI wl18xx support
      {M}       TI wlcore support
      [*]     ZyDAS devices
      <M>       USB ZD1201 based Wireless device support
      < >       ZyDAS ZD1211/ZD1211B USB-wireless support
      [*]     Quantenna wireless cards support
      <M>       Quantenna QSR10g PCIe support
      <M>     Aviator/Raytheon 2.4GHz wireless support
      <M>     Planet WL3501 PCMCIA cards
      <M>     Simulated radio testing tool for mac80211
      < >     Wireless RNDIS USB support

        <Select>    < Exit >    < Help >    < Save >    < Load >
```

Figure 1: List of the recommended drivers for Wireless Network devices

Remark: A lot of optimization can be done during the 'make menuconfig' step, by disabling everything that is not necessary.

1.9 Compile

```
### In my case, I need these to compile ###
apt-get install libsdl1.2-dev libssl-dev
```

```
make -j8 #8 is my number of cores
make modules_install install
```

1.10 Reboot

```
reboot
```

1.11 Checks

```
uname -a
sudo cat /proc/cpuinfo
lspci

sudo ./cyclictest -a -t -n -p99
```

1.12 (Optionnal) Test with ESPNOW_Echo code

```
git clone https://github.com/thomasfla/Linux-ESPNOW.git
cd Linux-ESPNOW/ESPNOW_lib/

make

### Then the usual before testing ###
sudo ifconfig wlan0 down
sudo iwconfig wlan0 mode monitor
sudo ifconfig wlan0 up
sudo iwconfig wlan0 channel 10

sudo ./bin/exec wlan0
```

2 Miscellaneous

2.1 Remove an installed kernel :

```
#Current kernel
uname -r
#DO NOT REMOVE IT

dpkg --list 'linux-image-*'
sudo apt-get purge linux-image-x.x.x-x-generic
sudo update-grub2
```