Wifi 802.11n Support in Linux

Vladimir Botka, PhD
Software Developer
SUSE LINUX GmbH
vbotka@suse.cz
IEEE Std 802.11n® - 2009

- October 2009 IEEE approved and published the 802.11n
- High throughput extension to the 802.11 standard
- Wireless adapters can achieve throughput up to 300 Mbps
- Physical layer data rates of 600 Mbps
Overview

- Features
- Implementation
- Deployment
- Troubleshooting
Features
Overview of Features

- Physical layer diversity techniques
- Channel bonding
- Frame aggregation
Overview of Features

user space applications

<table>
<thead>
<tr>
<th>nl80211</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cfg80211</td>
<td>wext</td>
</tr>
<tr>
<td>mac80211</td>
<td></td>
</tr>
</tbody>
</table>

wifi driver

physical layer
Features
Physical Layer Diversity Techniques

- Radio is sending and receiving with multiple antennas
- Multiple-input and multiple-output (MIMO)
- Spatial multiplexing
- Maximum Ratio Combining (MRC)
- Improved Signal to Noise Ratio (SNR)
Features
Physical Layer Diversity Techniques
Features
Channel Bonding

- Single channel is 20MHz in width
- Two bonded channels are 40MHz wide
- Both in 2.4GHz and 5GHz spectrum
- Limited number of non-overlapping channels in the 2.4GHz band (1,6,11)
## Features

### Channel Bonding

- Channels in 2.4 GHz band with 22MHz bandwidth 802.11 b/g/n
- Only channels 1,6,11 without overlap

<table>
<thead>
<tr>
<th>Channel</th>
<th>Frequency (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2412</td>
</tr>
<tr>
<td>2</td>
<td>2417</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2484</td>
</tr>
</tbody>
</table>
Features
Channel Bonding
Features
Channel Bonding

• Channels in 5 GHz band with 20/40MHz bandwidth 802.11 a/n
• 40 channels in the frequency range 4915 – 5825 (Mhz)
• Many different restrictions
• Countries apply their own rules
Features
Frame Aggregation

• Media Access Control (MAC)
• Transmit more data frames with “Block ACK”
• Burst
• Reduced overhead
Features
Frame Aggregation
Features

Summary

- 600Mbit/s physical transfer rate
- More antennas
- Max. 4 spatial streams using a 40MHz-wide channel
- Preferably in 5GHz band
Implementation
Overview of Implementation

- kernel.org
- linuxwireless.org
- New core mac80211 and cfg80211 components
- In-kernel nl80211 configuration interface
- nl80211 user-space applications
Physical layer (PHY) is implemented in the adapter
Firmware delivered by the producer
http://intellinuxwireless.org/
Implementation

Source code development trees
linux/kernel/git/pub/scm
wireless-testing - On-going wireless integration tree
iwlwifi - Intel Wireless WiFi Link driver
net/wireless
driver/net/wireless
Implementation
wireless-testing

- linuxwireless.org
- mac80211 subsystem for SoftMAC wireless devices
- cfg80211 new Linux wireless configuration API
- nl80211 new 802.11 netlink interface public header
Implementation

cfg80211

• Driver for wifi adapter can use cfg80211 operation callbacks and fill in the wiphy struct to store the device capabilities
• Bridges userspace and drivers
Implementation

nl80211 user-space applications

- Intended to replace Wireless-Extensions used by
- iw – manipulate configuration of wireless devices
- crda – central regulatory domain agent
- hostapd – AP implementation with authentication
- wpa_supplicant (with -Dnl80211)
Deployment
Overview of Deployment

- iw utility to manipulate and configure wireless devices
- crda central regulatory domain agent
- wireless regulatory database
- changes to the configuration of openSUSE
Deployment
iw utility

Can show and manipulate objects

- phy – physical layer of the device
- dev – network interface
- reg – regulatory database settings
Deployment
iw utility can show/manipulate physical layer

# iw phy
<snip>
Frequencies:
* 2412 MHz [1] (15.0 dBm)
* 2417 MHz [2] (15.0 dBm)
* 2422 MHz [3] (15.0 dBm)
....
* 2457 MHz [10] (15.0 dBm)
* 2462 MHz [11] (15.0 dBm)
* 2467 MHz [12] (15.0 dBm) (passive scanning, no IBSS)
* 2472 MHz [13] (15.0 dBm) (passive scanning, no IBSS)
<snip>
Deployment

iw utility can manipulate devices

# iw dev
phy#0
Interface wlan0
ifindex 3
type managed
Deployment

iw utility can manipulate regulatory settings

# iw reg set DE
# iw reg get

country DE:
(2400 - 2483 @ 40), (N/A, 20)
(5150 - 5350 @ 40), (N/A, 20), NO-OUTDOOR, DFS
(5470 - 5725 @ 40), (N/A, 26), NO-OUTDOOR, DFS
Deployment

crda central regulatory domain agent

- intended to be used by udev scripts
- triggered by the wireless kernel subsystem
- regulatory domain is read by crda from the regulatory.bin file

> cat /lib/udev/rules.d/85-regulatory.rules
KERNEL=="regulatory*", ACTION=="change", SUBSYSTEM=="platform",
RUN+="/sbin/crda"
Deployment
wireless regulatory database

- wireless-regdb.rpm
- Provides regulatory.bin file
- wireless.kernel.org/download/wireless-regdb/
- Integrity of regulatory file is ensured by signing
Deployment changes to the configuration of openSUSE

- `/etc/sysconfig/network/config`

- `WIRELESS_WPA_DRIVER='wext'` by default will be replaced with
  - `WIRELESS_WPA_DRIVER='nl80211'` for wlan drivers that are nl80211 ready

- `WIRELESS_REGULATORY_DOMAIN=''`
- Yast2 network changes
## Deployment

**Summary of changes in the user-space**

<table>
<thead>
<tr>
<th>g-standard</th>
<th>n-standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>iwconfig</td>
<td>iw</td>
</tr>
<tr>
<td>wpa_supplicant -D wext</td>
<td>wpa_supplicant -D nl80211</td>
</tr>
<tr>
<td>crda</td>
<td>wireless-regdb</td>
</tr>
</tbody>
</table>
Troubleshooting
Overview of Troubleshooting

- NetworkManager, wpa_supplicant, ifup/ifdown
- Interface setup to sniff the packets
- wireshark
- bugzilla.novell.com
Troubleshooting
NetworkManager, wpa_supplicant, ifup/ifdown

- en.opensuse.org/SDB:Tracking_down_wireless_problems
- Try to reduce the complexity and localize the problem

- /var/log/NetworkManager
- /var/log/wpa_supplicant
- /var/log/messages
Troubleshooting
Set debug options

- Wlan driver options
  > modinfo iwlagn | grep parm

- Set debug option
  > cat /etc/modprobe.d/50-iwlagn.conf
  options iwlagn debug=0xffffffff
Troubleshooting

Interface setup to sniff the packets

> iw dev wlan0 del
> iw phy phy0 interface add mon0 type monitor
> iw dev mon0 info
> ifconfig mon0 up
> tcpdump -i mon0 -w dump
Troubleshooting

wireshark

- wireshark.rpm

> wireshark dump
Troubleshooting
bugzilla.novell.com

openSUSE
bug reports can be posted to

bugzilla.novell.com
Have Fun!