Get It Together

John W. Linville

LinuxCon

21 September 2009

Who am I?

Why am I here?

Highlight current Linux wireless LAN topics...

- Not developer-centric...
- Not terribly detailed...
- Not Bluetooth/WiMAX/UWB...
- Not exhaustive coverage!

Background

Background and historical information...

- Full MAC vs “Soft” MAC
- What was ieee80211?
- What was ieee80211softmac?
- What is mac80211?

Full MAC vs. “Soft” MAC

Designs for wireless LAN adapters fall into two broad categories...

- Full MAC devices look a lot like Ethernet
    - Behavior is determined solely by vendor...
    - Lots of firmware, needs μC and memory resources — expensive...
    - Lots of vendor control, easier to support open source drivers!
- “Soft” MAC devices expose the ugly truth...
    - Little or no firmware — cheaper to produce...
    - Hardware vendor has less control...
    - Behavior can be made consistent across lots of devices!

What was ieee80211?

In the beginning, there was hostap...

- ieee80211 derived from hostap...
- Developed by Intel to support ipw2100 and ipw2200...
- Mostly a support library for low-level driver functions...
- Not sufficient to support most "soft" MAC devices...
- Renamed to libipw in current kernels, only used by ipw2100/ipw2200.

What was ieee80211softmac?

ieee80211softmac was developed as an add-on to support "soft" MAC devices...

- Several drivers developed, including zd1211rw and bcm43xx...
- Suffered from lack of features and inherent(?) design flaws...
- Replaced in upstream kernels by mac80211...
- Still lives-on (in multiple forms) under `drivers/staging/`...

What is mac80211?

mac80211 is the kernel infrastructure for "soft" MAC devices...

- Original code contribution by Devicescape in late-2005...
- Known for a while as "d80211"...
- Provides basic functionality for "soft" MAC devices...
- Provides common implementation across a large number of devices.

Development

- RFKill
- Regulations
- Configuration
- Consolidation

RFKill

RFKill is a subsystem to provide “killswitch” functionality for disabling radio hardware on mobile devices.

- Hardware or software?
- What kind of hardware?
- Who owns the switch?

Drivers originally implemented RFKill support on an ad-hoc basis...

RFKill (cont.)

No more jiggery-pokery!

- Drivers just register with the rfkill subsystem...
- Platform drivers ensure operation of special keys...
- `/dev/rfkill` talks to userland...
- cfg80211 implements "soft" RFKill as a backup...

Enables generic management applications!

Regulations

- Why regulations matter...
- CRDA
- Regulatory Database

Why regulations matter...

Governmental regulations are a much larger concern for wireless networking than for wired technologies (e.g. Ethernet).

- Legal repurcusions if we do the wrong things!
- Substantial financial penalties possible for vendors...
- Some vendors are reluctant to cooperate with us...

Vendors implemented regulatory compliance on their own...

CRDA

CRDA is a userland component used to update the kernel with regulatory information...

- Invoked by udev when regulatory domain is set...
- Uses signed database of regulatory rules...
- Supports verfication of multiple signatures...
- Without CRDA, kernel relies on limited set of regulatory rules!

Regulatory Database

Regulatory database used by CRDA is maintained separately...

- Facilitates separate maintenance of database...
- Contains signing infrastructure if you perfer to roll your own...
- `git://git.kernel.org/pub/scm/linux/kernel/git/wireless-regdb.git`

Configuration

- The Old Way...
- The New Way!
- Driver Support

The Old Way...

The “wireless extensions” API is ancient and venerable?

- Based upon the IOCTL system call...
- Drivers forced to reimplement lots of code...
- Specification is vague about behavioral details...
- Semantics are based on individual attributes rather than specific actions...
- Get the picture?

The New Way!

The new configuration API for wireless is cfg80211...

- Built around Netlink sockets...
- Drivers implement a small set of configuration methods...
- Proper implementation behavior is clearly defined...
- Semantics are based on flows defined in the IEEE802.11 specification.

A wireless extensions implementation on top of cfg80211 is provided for backward compatibility!

Driver Support

Support for cfg80211 is growing!

- mac80211-based drivers support cfg80211...
- So does iwm3200wifi...
- So does rndis_wlan...
- The orinoco driver is heading that way...
- And ipw2200 has started an implementation too!

But, there is still a lot of work to be done...

Consolidation

Due to its history, wireless code yearns for consolidation...

- Header file overlap...
- Cryptography code...
- Support for cfg80211 consolidates wireless extensions implementation...
- Other common bits related to frame parsing, etc...

The lib80211 component will ultimately corral these bits...

Community

- E-mail, Wiki, IRC
- Mini-Summits
- Vendor Support
- Patch Activity
- Staging Drivers
- Open Source Firmware

E-mail, Wiki, IRC

- `linux-wireless@vger.kernel.org`
- `http://wireless.kernel.org`
- #linux-wireless on Freenode

Mini-Summits

- Portland – September 2009
- Berlin – June 2009
- Ottawa – July 2008
- London – January 2007
- Portland – April 2006

Vendor Support

Support from wireless hardware vendors varies greatly...

- Atheros, Intel, Marvell, and Nokia have active developers!
- Ralink, Realtek, TI, and VIA provide some information...
- Broadcom provides a limited, closed solution...
- Other hardware is mostly old and/or reverse-engineered...

Patch Activity

Wireless LANs are an extremely active part of kernel development!

- Since 2.6.24, Linville #4 for “non-author signoff lines” (6%) `http://www.linuxfoundation.org/publications/whowriteslinux.pdf`
- For 2.6.32, over 850 wireless LAN patches!
- No sign of stopping...

Staging Drivers

An embarassingly large number of wireless drivers are in `drivers/staging/`...

- Most of them carry their own infrastructure...
- Many of them are atrociously bad code...
- Several of them overlap existing upstream drivers...
- Some of them have no supporting documents or code...

Please help!

Open Source Firmware

Open-source device firmware is starting to appear!

- Atheros released open source firmware for AR9170 devices... `http://git.sipsolutions.net/ar9170-fw.git`
- UniBS NTW has produced open source firmware for b43 devices! `http://www.ing.unibs.it/openfwwf/`
- Maybe more...?

Questions?

Contact

Feel free to contact me!

- Email linville@tuxdriver.com
    - ...@redhat.com
    - ...@gmail.com
    - ...@kernel.org
- IRC linville on Freenode, OFTC, and LinuxNET
- Facebook as “John W. Linville”

Slides available:

`http://www.kernel.org/pub/linux/kernel/people/linville/linuxcon2009/`