
Vortragsreihe der LUG Tübingen

04. Mai 2004

Ulrich Ölmann

WLAN

Hintergründe und Betrieb unter Linux



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - Praktische Beispiele
 - Zusammenfassung



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - Praktische Beispiele
 - Zusammenfassung



Drahtlose digitale Datenübertragung im Alltag

- Rauch- und Morsezeichen: bit/min oder gar bit/h
- Videotext
- Pager: Scall, Quix, digitale Funkmeldeempfänger (FW, DRK, ...)
- Schnurlostelefone nach DECT-Standard
(Digital Enhanced Cordless Telecommunications):
1.8 GHz, 32 kbit/s



Drahtlose digitale Datenübertragung im Alltag

- Digitale Mobiltelefonnetze:
 - GSM: 900 MHz oder 1.8 GHz, 14.4 kbit/s
 - UMTS: 2 Mbit/s
- GPS: 1.2 GHz und 1.5 GHz
- Satellitentelefone (Iridium-Netz)
- Digitaler Rundfunk (Radio, Fernsehen),
terrestrisch oder über Satellit



Drahtlose digitale Datenübertragung unter Linux

- Amateur-Funk, AX.25: 1-80kbit/s
- IrDA: 115.2 kbit/s
- Metricom Starmode Radio IP: 100 kbit/s
- Bluetooth: 2.45 GHz, 723 kbit/s
- WLAN nach IEEE 802.11: 2.4 GHz oder 5 GHz, 2 – 108 Mbit/s



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - Praktische Beispiele
 - Zusammenfassung



Nomenklatur

- MAC - Medium Access Control
Teil des WLAN-Gerätes, das Protokoll und Verbindung organisiert
- Funkmodem
Teil des WLAN-Gerätes, das durch Modulation von Funksignalen Daten mit dem „Äther“ austauscht
- Diversity
Generisches Konzept zur Steigerung der Zuverlässigkeit durch Einbringen von Redundanz



Nomenklatur

- Spread Spectrum
 - Technik zur Erhöhung der Zuverlässigkeit der Funkverbindung
 - Erlaubt unabhängigen Systemen die Koexistenz
 - Zwingt dabei zur Teilung der Bandbreite
- DS - Direct Sequence (Spread Spectrum Technik)
 - Vorteil: Läuft auf einem großen Kanal
 - Nachteil: Benötigt aufwendigeres Funkmodem
- FH - Frequency Hopping (Spread Spectrum Technik)
 - Vorteil: Kostengünstigeres Modem
 - Nachteil: Läuft auf mehreren schmalen Kanälen



Standardisierung innerhalb von 802.11

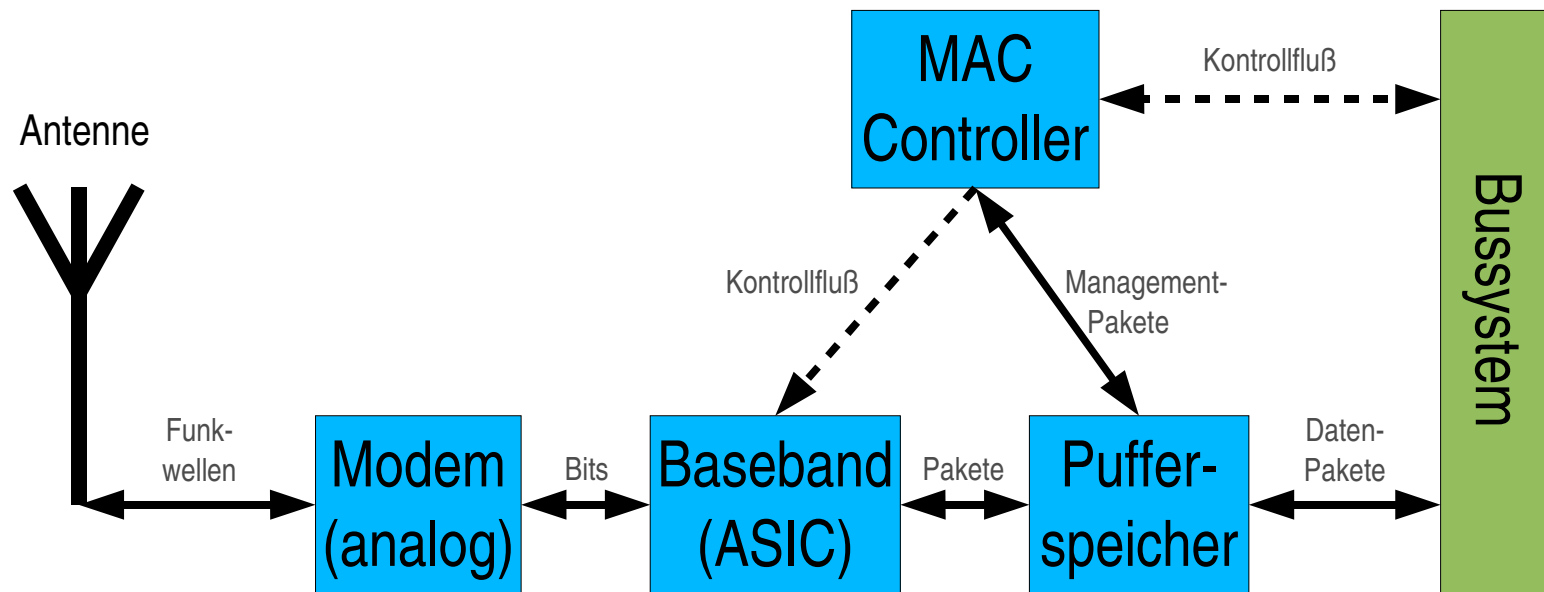
- Langwieriger Prozess innerhalb des IEEE 802.11 Komitees
- Sehr komplexes Ergebnis durch zuviele enthaltene Features

802.11 spezifiziert:

- ein MAC Protokoll
- drei alternative physikalische Schichten:
 - Frequency Hopping (1 Mbit/s)
 - Direct Sequence (1 und 2 Mbit/s)
 - Infrarot (1 Mbit/s bei 850 nm)



Aufbau einer Funknetzwerkkarte



MAC - Medium Access Control

- Sinnvolle Netzwerke bestehen aus mindestens 2 Knoten, die über ein gemeinsames Medium kommunizieren.
- Erfolgreiche Kommunikation bedeutet hierbei:
Ein Knoten spricht, der Rest lauscht.
- MAC-Protokoll sorgt für faire Verteilung der Redezeit und möglichst geringe Beeinträchtigungen in Folge von Durch-einandersprechen.
- Beispiel: Tokennet
- Beispiel: Ethernet



MAC - Medium Access Control

- 802.11 MAC-Protokoll ähnlich bewährter Ethernet-Technik (CSMA/CD – Carrier Sense Multiple Access / Collision Detection)
- Entscheidender Unterschied: Keine Kollisionserkennung möglich
⇒ CSMA/CA – Carrier Sense Multiple Access / Collision Avoidance
- Zusätzliche Techniken zur Effizienzsteigerung:
 - MAC Retransmissions
 - Fragmentation
 - RTS/CTS (Request To Send / Clear To Send)



MAC - Netzwerktopologie: Ad-Hoc-Netzwerk

- Alle Knoten sind gleichberechtigt
- Knoten können wie bei Ethernet beliebig aus dem Netz entfernt oder zu dem Netz hinzugefügt werden.
- Voraussetzung für ein Ad-Hoc-Netzwerk: Jeder Knoten muß die restlichen Knoten „sehen“ können.



MAC - Netzwerktopologie: Verwaltetes Netzwerk

- Es gibt ausgezeichnete Knoten: Access Points
- Sie erfüllen besondere Funktionen:
 - Bridge zwischen WLAN und Ethernetsegment
 - Out of Range Forwarding
 - Access Control
 - Roaming



MAC - Sicherheit

- Zugangskontrolle:
 - ESSID (Extended Service Set ID) → „Netzwerkname“
 - MAC-ACLs (Access Control Lists)
 - Reichweite der Funknetzwerkkarte

 - Sicherheit gegen Belauschen
 - Sicherstellung der Datenintegrität
- } → WEP
(Wire Equivalent Privacy)



MAC - Sicherheit

⇒ Quintessenz:

Traue den aktuellen Sicherheitsfeatures von 802.11 nicht!

⇒ Ausweg:

Nimm die Sicherheit selbst in die Hand

(Verwendung von SSH, HTTP über SSL/TLS, VPN-Lösung, ...)

Ausblick:

IEEE 802.11-i enthält Spezifikation von WPA (Wi-Fi Protected Access), WPA2 (verwendet AES statt RC4) soll Mitte 2004 spezifiziert sein.



Standards innerhalb von 802.11

- 802.11-FH 2.4 GHz, 2 Mbit/s, FH = Frequency Hopping
- 802.11-DS 2.4 GHz, 2 Mbit/s, DS = Direct Sequence
- 802.11-b \cong 802.11-HR (modifiziertes 802.11-DS)
 + 5.5 Mbit/s + 11 Mbit/s
- 802.11+ \cong 802.11-b + 22 Mbit/s + 33 Mbit/s + 44 Mbit/s



Standards innerhalb von 802.11

- 802.11-g 802.11-b + 6, 9, 12, 18, 24, 36, 48, 54 & 108 Mbit/s
Standardisierung im Juni 2003 abgeschlossen.
Daher bei „frühen“ Geräten nicht notwendig Interoperabilität mit 802.11-g Geräten anderer Hersteller.
Bei „frühen“ Geräten ebenfalls Störung von 802.11-b Netzwerken möglich.
- 802.11-a 5 GHz, keine Interoperabilität mit 802.11-b



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - Praktische Beispiele
 - Zusammenfassung



Linux Wireless Extensions

- Als OpenSource-Projekt von Jean Tourrilhes 1996 angeregt (Autor des Linux Wireless LAN HOWTO)
- Enthalten im Linux-Kernel seit mind. Version 2.2.14 bzw. 2.3.30
- Aktuelle Version: WE-16
- Rahmenvorgaben für WLAN-Treiber:
Neue Menge von ioctl-Calls und /proc/net/wireless

```
ulla@laptop-ulla:~> cat /proc/net/wireless
Inter-| sta-| Quality          | Discarded packets          | Missed | WE
face | tus | link level noise | nwid  crypt  frag  retry  misc | beacon | 16
eth1: 0000 205.   0.   28.      0      0     0     0     0 | 0       |
```



Linux Wireless Tools

- Darauf aufbauend die Wireless-Tools:
 - iwconfig Konfiguriere ein WLAN-Gerät
 - iwpriv Konfiguriere optionale (private) Features
 - iwspy Statistiken von spezifischen WLAN-Knoten
 - iwlist Detailliertere Informationen von einem WLAN-Gerät
 - iwevent Zeige Wireless Events an
 - ifrename Benenne Netzwerkinterface um
- Aktuelle Version: Wireless Tools 26 (27-beta vorhanden)
- Referenzimplementation, Alternativen vorhanden



Linux Wireless Tools: iwconfig

- Konfiguration eines WLAN-Gerätes:

```
ulla@laptop-ulla:~  
laptop-ulla:~ # iwconfig eth1 essid LUGT channel 3 mode Managed ap 00:09:5B:97:D5:A3  
laptop-ulla:~ # iwconfig eth1 key s:"Linux User Group"  
laptop-ulla:~ # iwconfig eth1 key restricted  
laptop-ulla:~ #  
laptop-ulla:~ # iwconfig eth1  
eth1      NOT READY!  ESSID:"LUGT"  
          Mode:Managed Channel:3 Access Point: 00:09:5B:97:D5:A3  
          Tx-Power=31 dBm   Sensitivity=0/200  
          Retry min limit:0 RTS thr=0 B  Fragment thr=0 B  
          Encryption key:4C69-6E75-7820-5573-6572-2047-726F-7570 Security mode:restricted  
          Link Quality:0 Signal level:0 Noise level:0  
          Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0  
          Tx excessive retries:0 Invalid misc:0 Missed beacon:0  
  
laptop-ulla:~ # █
```



Linux Wireless Tools: iwpriv

- Konfiguration von treiberspezifischen Features des WLAN-Gerätes:

```
ulla@laptop-ulla:~  
laptop-ulla:~ # iwpriv eth1  
eth1 Available private ioctl :  
reset (8BE0) : set 0 & get 0  
getBeaconPeriod (8BE1) : set 0 & get 1 int  
setBeaconPeriod (8BE2) : set 1 int & get 0  
getPolicy (8BE3) : set 0 & get 1 int  
setPolicy (8BE4) : set 1 int & get 0  
getMac (8BE5) : set 0 & get 64 addr  
addMac (8BE6) : set 1 addr & get 0  
delMac (8BE8) : set 1 addr & get 0  
kickMac (8BEA) : set 1 addr & get 0  
kickAll (8BEC) : set 0 & get 0  
get_wpa (8BED) : set 0 & get 1 int  
set_wpa (8BEE) : set 1 int & get 0  
oid (8BF0) : set 1 int & get 0  
get_oid (8BF1) : set 0 & get 256 byte  
set_oid (8BF2) : set 256 byte & get 0  
  
laptop-ulla:~ # iwpriv eth1 setPolicy 2  
laptop-ulla:~ # iwpriv eth1 addMac 00:09:5B:97:D5:A3  
laptop-ulla:~ # █
```



Linux Wireless Tools: iwlist

- Detailliertere Informationen zum akt. Zustand des WLAN-Gerätes

```
ulla@laptop-ulla:~  
laptop-ulla:~ # iwlist eth1 key  
eth1      3 key sizes : 40, 104, 256bits  
          4 keys available :  
            [1]: 556C-7269-6368-20D6-6C6D-616E-6E (104 bits)  
            [2]: off  
            [3]: 4C69-6E75-78 (40 bits)  
            [4]: off  
          Current Transmit Key: [1]  
          Security mode:restricted  
  
laptop-ulla:~ # █
```



Paket Sniffer: Kismet

- Schaltet das WLAN-Gerät in besonderen Promiscuous Mode
- Detektiert auf diese passive Weise sämtlichen drahtlosen Verkehr, ohne selbst Spuren zu hinterlassen
- Analysiert eintreffende Pakete, ordnet sie nach Netzen und erlaubt direkten Zugriff auf Informationen über diese Netze
- Protokolliert den Paketdatenstrom zur weiteren Analyse mit tcpdump oder ethereal im WTAP-Format
- Erlaubt mit Hilfe von gpsd die Erstellung genauer Karten



Paket Sniffer: Kismet

- Kismet-Protokoll meiner Anfahrt von Stuttgart nach Tübingen

```
Network List (Latest Seen)
Name      T W Ch  Packts  Flags  IP Range  Size
-----
KrzNet    A N 011    24    0.0.0.0  0B
Privat    A Y 005     1    0.0.0.0  0B
T-Mobile_T-Com  A N 006     1    0.0.0.0  0B
T-Mobile_T-Com  A N 011     1    0.0.0.0  0B
Mohring   A Y 011     4    0.0.0.0  0B
Wireless  A N 006     1    0.0.0.0  0B
default   A Y 006     6    0.0.0.0  1k
default   A N 011     1    0.0.0.0  0B
win2linux8.2  A N 006     1    0.0.0.0  0B
andy      A N 006     2    0.0.0.0  0B
router    A N 009     2    0.0.0.0  0B
ConnectionPoint  A N 010     5    0.0.0.0  0B
tikogrupa  A Y 011     1    0.0.0.0  0B
WirelessTeam  A Y 006     1    0.0.0.0  0B
+ Probe Networks  G N ---    19    0.0.0.0  0B

Info
Ntwrks  16
Pckets  94
Cryptd   2
Weak     0
Noise    8
Discrd  14
Pkts/s   0
Elapsd  00:33:11

Status
Found new network "tikogrupa" bssid 00:09:5B:AD:B4:4C WEP Y Ch 11 @ 11.00 mbit
Saving data files.
Found new network "WirelessTeam" bssid 00:04:E2:7D:23:2E WEP Y Ch 6 @ 11.00 mbit
Found new probed network "^U^F^F^M^C^Y^[^Y^]^G^[^E^W^S^P" bssid 00:04:23:75:E2:49
Battery: 74% 1h38m0s
```



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - **Praktische Beispiele**
 - Zusammenfassung



Praktische Beispiele

- Zwei beteiligte Laptops mit Cardbus WLAN-Karten Netgear WG511
- Verwendeter Chipsatz: Prism GT, unter Linux unterstützt durch prism54-Treiber (siehe Linksammlung am Ende)
- Karte wird in der aktuellen Knoppix-Distribution unterstützt
- Vorbereitung: Boote auf Laptops A und B Knoppix



Praktische Beispiele

- In den Beispielen werden Übertragungstests durchgeführt, hierzu werden abkürzend die folgenden Shell-Scripte benutzt:
 - ♦ **transmit.sh**

```
#!/bin/sh
time dd if=/dev/zero bs=1M count=32 |
  netcat -w 2 192.168.0.2 1234
```
 - ♦ **receive.sh**

```
#!/bin/sh
netcat -l -p 1234 > /dev/null
```
- Zur Messung müssen zunächst auf Laptop B `receive.sh` und dann auf Laptop A `transmit.sh` gestartet werden. Laptop A gibt anschließend die Zeit für die Übertragung von 32 MB Nullen über das WLAN aus.



Praktische Beispiele: Ad-Hoc Netzwerk

- Grundkonfiguration der Netzwerkkarten

Laptop A

```
iwconfig eth0 mode Ad-Hoc  
ifconfig eth0 192.168.0.1
```

Laptop B

```
iwconfig eth0 mode Ad-Hoc \  
    essid LUGT  
ifconfig eth0 192.168.0.2
```

- Ping von Laptop A an Laptop B (schlägt fehl!)

Laptop A

```
ping 192.168.0.2
```

- Bei Laptop A fehlt die ESSID: Korrektur, erfolgreicher Ping und Zeitmessung

Laptop A

```
iwconfig eth0 essid LUGT  
ping 192.168.0.2  
./transmit.sh
```

Laptop B

```
./receive.sh
```



Praktische Beispiele: Verwaltetes Netzwerk

- Neue Konfiguration der Netzwerkkarten (A wird Access-Point, B einfacher teilnehmender Knoten)

Laptop A

```
iwconfig eth0 mode Master
```

Laptop B

```
iwconfig eth0 mode Managed
```

- Ping von Laptop A an Laptop B funktioniert, anschließend Messung der Übertragungsrate

Laptop A

```
ping 192.168.0.2  
./transmit.sh
```

Laptop B

```
./receive.sh
```



Praktische Beispiele: WEP-Verschlüsselung

- Neue Konfiguration der Netzwerkkarten (Schlüssel wird eingetragen und automatisch auf 'restricted'-Modus geschaltet)

Laptop A

```
iwconfig eth0 key \  
s:"Ulrich Ölmann"
```

Laptop B

```
iwconfig eth0 key \  
s:"Ulrich Ölmann"
```

- Ping von Laptop A an Laptop B funktioniert, anschließend Messung der Übertragungsrate

Laptop A

```
ping 192.168.0.2  
./transmit.sh
```

Laptop B

```
./receive.sh
```



Gliederung des Vortrags

- Einleitung
 - IEEE 802.11
 - Schnittstelle zu Linux
 - Praktische Beispiele
- Zusammenfassung



Zusammenfassung

- 802.11 als eine von vielen aktuellen Techniken zur drahtlosen Datenübertragung ist praxistauglich
- Zugrundeliegende Technik ist sehr komplex
- Geräte werden günstiger, Linux-Treiber zahlreicher und stabiler
- Vielseitige Anwendbarkeit: von Ad-Hoc-Netzen bis zu verwalteten Netzwerken mit Roaming
- Drahtlose Sicherheit zur Zeit noch nicht „out of the box“



Links zum Thema

- **Wireless LAN resources for Linux**

[http://www.hpl.hp.com/personal/Jean_Tourrilhes/
↳Linux/Wireless.html](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html)

- **Kismet**

<http://www.kismetwireless.net/>

- **Airsnort**

<http://airsnort.shmoo.com>

- **Webseite der Treiberentwickler für Prism-Chipsätze**

<http://www.prism54.org/>



ENDE

