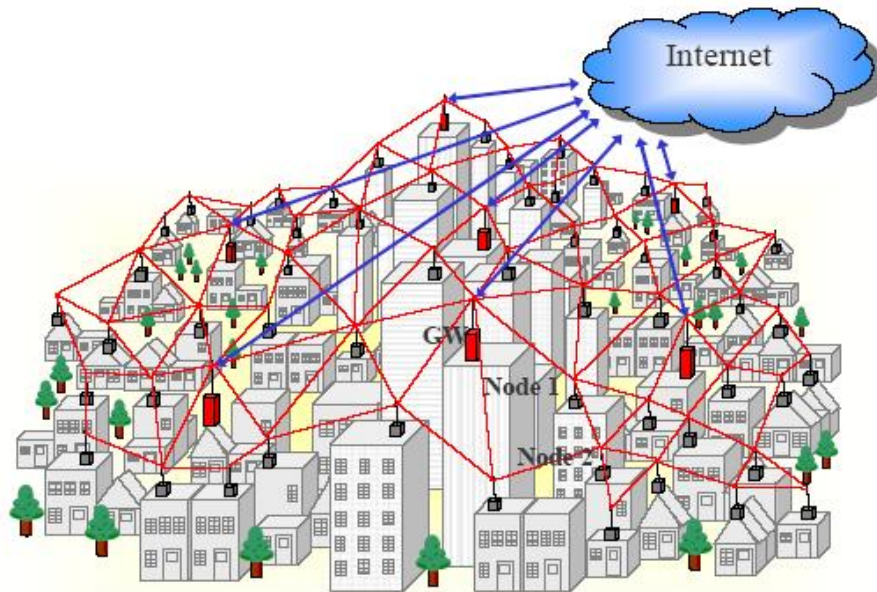


Fachstudie

Hardwareplattformen und Systemsoftware für drahtlose vermaschte Kommunikationsnetze



Bearbeiter: Alex Egorenkov
Sergey Telejnikov
Valeri Schneider
Betreuer: Dipl.-Inf. Frank Dürr
Prüfer: Prof. Dr. Kurt Rothermel
Zeitraum: November 2007 - Januar 2008

Abstract

†Absatz wie im Dokument!

TODO

Mesh-Netze (engl. Wireless Mesh Network, WMN) sind drahtlose Ad-Hoc-Netze bestehend aus stationären Mesh-Routern, die einen Routing-Backbone bilden, und mobilen oder stationären Mesh-Clients. Die Mesh-Clients kommunizieren über den Backbone mit anderen Mesh-Clients oder erlangen über den Backbone Zugang zum Internet. Mesh-Netze können dabei auch größere Bereiche, beispielsweise ganze Städte abdecken (entsprechende Stadtnetze werden aktuell z.B. durch Google installiert).

Ein entsprechendes Mesh-Netz muss für die Forschungszwecke für den Sonderforschungsbereich (SFB) Nexus an der Universität Stuttgart eingerichtet werden.

Diese Fachstudie befasst sich mit der Ausarbeitung einer Empfehlung für die Beschaffung entsprechender Geräte (*Hardwareplattformen und Systemsoftware*) für den Aufbau eines WMN.

Inhaltsverzeichnis

1	Einleitung	6
1.1	Grundlagen von Mesh-Netzen	6
1.1.1	Hintergrund	6
1.1.2	Ad-Hoc	6
1.1.3	Mesh-Netz	7
1.1.4	IEEE 802.11a/b/g	7
1.1.5	Ad-Hoc Routing-Protokolle	8
1.1.5.1	OLSR	10
1.1.5.2	B.A.T.M.A.N.	10
1.2	Existierende Lösungen und Projekte	11
2	Aufgabenstellung	12
3	Anforderungen	12
3.1	Anforderungen an Hardware	13
3.2	Anforderungen an Software	13
3.3	Zusätzliche Anforderungen	13
4	Hardware-Lösungen für den Aufbau eines Mesh-Netzwerkes	15
4.1	PCs + WLAN-Karten	15
4.1.1	PCI-WLAN-Karten	16
4.1.1.1	Linksys WMP55AG	17
4.1.1.2	Netgear WAG311	19
4.1.1.3	D-Link DWL-A520	21
4.1.1.4	Gigabyte GN-WPEAG	23
4.1.1.5	Andere PCI-WLAN-Karten	24
4.1.2	Mini-PCI WLAN-Karten	25
4.1.2.1	Wistron CM9 Atheros AR5213A	26
4.1.2.2	Intel PRO/Wireless 3945	28
4.1.2.3	Intel PRO/Wireless 2915	30
4.1.2.4	Intel Wireless WiFi Link 4965AGN	32
4.1.3	PCMCIA WLAN-Karten	34
4.1.3.1	Proxim Orinoco Gold 8480-WD	35
4.1.3.2	Netgear WAG511	37
4.1.3.3	SMC 2536W-AG	39
4.1.3.4	Linksys WPC55AG	41
4.1.3.5	Andere PCMCIA-WLAN-Karten	42
4.2	WLAN-Router	44
4.2.1	SoHo-Router	44
4.2.1.1	Linksys WRT54G v1.0	45
4.2.1.2	Linksys WRT55AG	47
4.2.1.3	Asus WL500G/GP	49

4.2.1.4	Andere WLAN-Router	50
4.2.2	Professionelle Router	51
4.2.3	Access Points	52
4.3	PDA's und Handys	53
5	Systemsoftware für Mesh-Netzwerk	54
5.1	Linux MadWiFi-Treiber	54
5.2	OpenWRT	55
5.3	Mesh-Routing Software	56
5.3.1	olsr.org OLSR daemon	56
5.3.2	Open-Mesh B.A.T.M.A.N. daemon	57
6	Tests	58
6.1	Erster Test	58
6.1.1	Hardware	58
6.1.2	Software	59
6.1.2.1	Madwifi	59
6.1.2.2	OLSR Daemon	61
6.1.3	Ergebnisse	61
6.1.3.1	Topologie	62
6.1.3.2	Http Info	62
6.2	Zweiter Test	63
7	Fazit	64
7.1	Übersicht	64
7.2	Aufwandschätzung und Bewertung	65
7.3	Empfehlung für eine geeignete WMN	65

Abbildungsverzeichnis

1	Linksys WMP55AG	17
2	Netgear WAG311	19
3	D-Link DWL-A520	21
4	Gigabyte GN-WPEAG	23
5	Wistron CM9 Atheros AR5213A	26
6	Intel PRO/Wireless 3945	28
7	Intel PRO/Wireless 2915	30
8	Intel Wireless WiFi Link 4965AGN	32
9	Proxim Orinoco Gold 8480-WD	35
10	Netgear WAG511	37
11	SMC 2536W-AG	39
12	Linksys WPC55AG	41
13	Linksys WRT54G v1.0	45

14	Linksys WRT55AG	47
15	Asus WL500G/GP	49
16	Topologie	58
17	Topologie Graph	62
18	Http Info	63
19	Übersicht und Bewertung von Hardware	64
20	Übersicht und Bewertung von Software	65

1 Einleitung

In diesem Abschnitt werden einige wichtige Begriffe, die im Laufe des Dokuments auftauchen werden, kurz erläutert.

1.1 Grundlagen von Mesh-Netzen

1.1.1 Hintergrund

Ein drahtloses vermaschtes Netz (engl. Wireless Mesh Network, WMN) besteht aus einer Menge von Knoten, die über drahtlose Kommunikationstechniken wie beispielsweise IEEE 802.11 (LINK) Nachrichten austauschen. Die Vermaschung der Knoten ermöglicht dabei nicht nur den Austausch von Nachrichten zwischen unmittelbar benachbarten Knoten, sondern auch die Vermittlung von Nachrichten an entfernte Knoten über mehrere Knoten hinweg. Die Vermittlungsfunktionalität wird dabei oft von dedizierten Vermittlungsknoten (engl. Mesh Router) bereitgestellt, die somit eine drahtlose Kommunikationsinfrastruktur für die Klienten (engl. Mesh Client) bilden. Durch den Einsatz vergleichsweise kostengünstiger Hardwarekomponenten und die Vermaschung der Knoten ermöglichen WMNs die kostengünstige Vernetzung auch größerer Gebiete. Entsprechende Netze werden beispielsweise von Community-Projekten wie das Freifunk-Projekt (LINK) oder Firmen wie Google (LINK) bereits heute in der Praxis für den Aufbau größerer Netze eingesetzt, um beispielsweise kostengünstige Internetzugänge für Stadtteile oder ganze Städte zu realisieren.

WMNs sind auch für den Sonderforschungsbereich (SFB) Nexus an der Universität Stuttgart <http://www.nexus.uni-stuttgart.de> von großem Interesse. Im Zentrum der Forschungen des SFB stehen Umgebungsmodelle für mobile kontextbezogene Systeme. Umgebungsmodelle sind digitale Abbilder der physischen Welt, die von kontextbezogenen Systemen genutzt werden, um sich selbständig an die physische Umgebung des Benutzers anzupassen. Ein einfaches Beispiel sind ortsbezogene Anwendungen, die beispielsweise aufgrund der aktuellen geographischen Position eines Geräts automatisch Informationen über nahe Restaurants, Sehenswürdigkeiten, usw. selektieren können. Zur Kommunikation, insbesondere mit mobilen Geräten, werden dabei hybride Systeme betrachtet, in denen sowohl eine infrastrukturbasierte Kommunikation als auch die direkte Ad-hoc-Kommunikation zwischen mobilen Endsystemen möglich ist. Hierbei spielen WMNs als eine spezielle Ausprägung eines hybriden Kommunikationssystems eine wesentliche Rolle.

1.1.2 Ad-Hoc

Ein Ad-hoc-Netz bezeichnet in der Informationstechnologie eine drahtlose Netzwerktopologie zwischen zwei oder mehr Endgeräten, die ohne feste Infrastruktur auskommt.

(TODO) Mehr Text.. Bilder: Adhoc, Infrastruktur!

1.1.3 Mesh-Netz

(TODO) Mesh-Netze sind Ad-hoc Netze

In einem vermaschten Netz (Mesh-Netz) ist jeder Netzwerkknoten mit einem oder mehreren anderen verbunden. Die Informationen werden von Knoten zu Knoten weitergereicht, bis sie das Ziel erreichen. Vermaschte Netze sind im Regelfall selbstheilend und dadurch sehr zuverlässig: Wenn ein Knoten oder eine Verbindung blockiert ist oder ausfällt, kann sich das Netz darum herum neu stricken. Die Daten werden umgeleitet und das Netzwerk ist nach wie vor betriebsfähig.

(TODO) Einarbeitung in grundlegende WMN-Technologien???

1.1.4 IEEE 802.11a/b/g

IEEE 802.11 (auch: Wireless LAN, WLAN, WiFi) bezeichnet eine IEEE-Norm für drahtlose Netzwerkkommunikation. Herausgeber ist das Institute of Electrical and Electronics Engineers (IEEE).

802.11a spezifiziert eine weitere Variante der physikalischen Schicht, die im 5-GHz-Band arbeitet und Übertragungsraten bis zu 54 MBit/s ermöglicht.

Vorteile

- weniger genutztes Frequenzband, dadurch häufig störungsfreier Betrieb möglich
- in Deutschland 19 (bei BNetzA-Zulassung) nicht überlappende Kanäle
- höhere Reichweite, da mit 802.11h bis zu 1000 mW Sendeleistung möglich

Nachteile

- stärkere Regulierungen in Europa: auf den meisten Kanälen DFS nötig
- auf einigen Kanälen kein Betrieb im Freien erlaubt
- falls kein TPC benutzt wird, muss die Sendeleistung reduziert werden
- Ad-hoc-Modus wird von den meisten Geräten nicht unterstützt
- geringere Verbreitung, daher wenig verfügbare Geräte auf dem Markt und hohe Kosten

802.11b ist ebenfalls eine alternative Spezifikation der physikalischen Schicht, die mit dem bisher genutzten 2,4-GHz-Band auskommt und Übertragungsraten bis zu 11 MBit/s ermöglicht.

Vorteile

- gebührenfreies freigegebenes ISM-Frequenzband
- hohe Verbreitung und daher geringe Gerätekosten

Nachteile

- Frequenz muss mit anderen Geräten/Funktechniken geteilt werden (Bluetooth, Mikrowellenherde, etc.)
- störungsfreier Betrieb von nur maximal 3 Netzwerken am selben Ort möglich, da effektiv nur 3 brauchbare (kaum überlappende) Kanäle zur Verfügung stehen (in Deutschland: 1, 7, 13)

1.1.5 Ad-Hoc Routing-Protokolle

Ein Ad-hoc Netzwerk beispielsweise ein Sensornetzwerk besteht aus einer großer Anzahl von Knoten, die sich typischerweise spontan vernetzen müssen. Über Funk können diese Knoten miteinander kommunizieren. Aber Knoten kann wegen der Leistungsfähigkeit, des Energieverbrauches und auch der Mobilität nicht mit allen anderen Knoten direkt Daten übertragen. Routing Methoden bieten die Möglichkeit, Kommunikation zwischen zwei Knoten mit Hilfe der anderen Knoten auszuführen, wenn die zwei Knoten miteinander nicht direkt kommunizieren können.

Es gibt mehr als 70 konkurrierende Entwürfe für das Routing der Pakete durch ein Mobiles Ad-Hoc/Maschennetzwerk. Eine Klassifikation der Routingprotokolle kann durch Anzahl der Empfänger getroffen werden:

- unicast Routing - Ziel der Datenübertragung ist ein einzelner Knoten
- multicast Routing - Ziel sind mehrere Knoten
- geocast Routing - Ziel sind alle Knoten in einem bestimmten geografischen Bereich
- broadcast Forwarding - Ziel sind alle Knoten in der Reichweite des Senders

Eine andere Möglichkeit der Klassifikation besteht in der Einteilung der Protokolle hinsichtlich des grundsätzlichen Ansatzes. Diese Ansätze werden im Folgenden vorgestellt.

Positionsbasierte Routingverfahren

Positionsbasierte Routingverfahren nutzen geodätische Informationen über die genauen Positionen der Knoten. Diese Informationen werden z. B. über GPS-Empfänger gewonnen. Anhand dieser Ortsinformationen lässt sich der kürzeste oder der beste Pfad zwischen Quell- und Zielknoten bestimmen. Ein Beispiel für ein positionsbasiertes Routingprotokoll ist LAR.

Topologiebasierte Routingverfahren

Die topologiebasierten Routingverfahren kommen ohne geodätische Informationen über die Positionen der Knoten des mobilen Ad-hoc-Netzes aus. Ihnen genügen logische Informationen über die Nachbarschaftsbeziehungen der Knoten, also welche Knoten eine direkte Verbindung haben oder über einen oder mehrere Zwischenknoten (hop) in Verbindung treten können. Diese Nachbarknoten können miteinander kommunizieren. Die topologischen Informationen werden meistens durch den Versand so genannter HELLO-Pakete gewonnen. Je nach Zeitpunkt des Aufbaus der Topologiedatenbasis handelt es sich um proaktives oder reaktives Routing. Ein Beispiel für ein Protokoll aus dieser Klasse ist das Neighbourhood Discovery Protocol (NHDP), das Elemente des Optimized Link State Routing Protocol (OLSR) verwendet.

Proaktive Verfahren

Proaktive Routingverfahren bestimmen die zu verwendenden Pfade zwischen zwei Knoten bereits, bevor diese für die Übertragung von Nutzdaten benötigt werden. Sollen dann Nutzdaten verschickt werden, so muss nicht auf die Bestimmung des Pfads zum Zielknoten gewartet werden. Nachteilig ist dafür jedoch, dass diese Verfahren auch ohne Verkehr von Nutzdaten viele Kontrollpakete verschicken, um Pfade zu bestimmen, die womöglich später nicht benötigt werden. Ein Beispiel für ein Protokoll aus dieser Klasse ist das Optimized Link State Routing Protocol (OLSR).

Reaktive Verfahren

Im Gegensatz zu den proaktiven Verfahren bestimmen reaktive Routingverfahren für mobile Ad-hoc-Netze die benötigten Pfade zwischen zwei Knoten erst, wenn Nutzdaten übertragen werden sollen. Daraus ergibt sich, dass das erste Datenpaket einer Verbindung erst mit Verzögerung versendet werden kann, da zunächst auf den Abschluss der Routenbestimmung gewartet werden muss. Dafür werden allerdings auch nur Kontrollpakete versendet, wenn Nutzdaten verschickt werden und dies zur Routenbestimmung notwendig ist. Dies schlägt sich positiv im Energieverbrauch der Knoten nieder. Das Protokoll Ad hoc On-Demand Distance Vector (AODV) ist ein Beispiel für ein Protokoll dieser Kategorie.

Hybride Verfahren

Hybride Verfahren kombinieren proaktive und reaktive Routingverfahren. Dabei soll das

Ziel erreicht werden, die Vorteile der beiden Ansätze in einem neuen Routingprotokoll zusammenzufassen. Beispielsweise kann in einem lokal beschränkten Bereich ein proaktives Verfahren eingesetzt werden, während für weiter entfernte Ziele ein reaktives Verfahren eingesetzt wird. Dies vermindert die Belastung des Netzes durch Kontrollpakete, die bei einem rein proaktiven Verfahren über das gesamte Netz versendet würden. Trotzdem stehen für lokale Ziele sofort Pfade zur Verfügung, ohne dass auf deren Bestimmung wie bei einem rein reaktiven Verfahren gewartet werden müsste. ZRP ist ein Routingprotokoll, das diesen Ansatz umsetzt.

1.1.5.1 OLSR



Optimized Link State Routing, kurz OLSR, ist ein Routingprotokoll für mobile Ad-hoc-Netze, das eine an die Anforderungen eines mobilen drahtlosen LANs angepasste Version des Link State Routing darstellt. Es wurde von der IETF mit dem RFC 3626 standardisiert. Bei diesem verteilten flexiblen Routingverfahren ist allen Routern die vollständige Netztopologie bekannt, sodass sie von Fall zu Fall den kürzesten Weg zum Ziel festlegen können. Als proaktives Routingprotokoll hält es die dafür benötigten Informationen jederzeit bereit. Ein in Mesh-Netzwerken bekannter Vertreter von LSR ist OLSR von olsr.org. Inzwischen existieren für OLSR spezielle Erweiterungen. Mit der ETX-Erweiterung wird dem Umstand Rechnung getragen, dass Links asymmetrisch sein können. Mit dem Fisheye-Algorithmus ist OLSR auch für größere Netzwerke brauchbar geworden, da Routen zu weiter entfernten Knoten weniger häufig neu berechnet werden. Der entscheidende Nachteil ist aber der trotz Fisheye-Algorithmus noch recht hohe Rechenaufwand von OLSRD, sobald die Anzahl an Knoten ein gewisses Maß übersteigt.

1.1.5.2 B.A.T.M.A.N.



Ausgehend von den Erfahrungen mit Freifunk-OLSR begannen die Entwickler aus der Freifunk-Community im März 2006 in Berlin damit, ein neues Routingprotokoll für drahtlose Meshnetzwerke zu entwickeln. Alle bisher bekannten Routingalgorithmen versuchen Routen entweder zu berechnen (proaktive Verfahren) oder sie dann zu suchen, wenn sie gebraucht werden (reaktive Verfahren). Das neue Protokoll B.A.T.M.A.N. (BETTER APPROACH TO MOBILE ADHOC NETWORKING) berechnet oder sucht im Gegensatz zu diesen Protokollen keine Routen, es erfasst lediglich, ob Routen zu anderen Knoten existieren und überwacht ihre Qualität. Dabei interessiert es sich nicht dafür, wie eine Route verläuft, sondern ermittelt lediglich, über welchen direkten Nachbarn ein bestimmter Netzwerkknoten am besten zu erreichen ist, und trägt diese Information proaktiv in die Routingtabelle ein.

1.2 Existierende Lösungen und Projekte

(TODO) Einbißchen Text zu jedem Projekt (googeln)

- Freifunk <http://freifunk.net/wiki/Meshing>
- OpenNet <http://wiki.opennet-initiative.de/index.php/Hauptseite>
- <http://www-i4.informatik.rwth-aachen.de/mcg/projects/umic-mesh/>
- <http://umic-mesh.net/>
- (TODO) Google..

2 Aufgabenstellung

Für Forschungszwecke soll innerhalb des SFB Nexus (<http://www.nexus.uni-stuttgart.de>) ein WMN installiert werden.

Dieses WMN dient

- einerseits Nexus-Anwendungen, insbesondere Anwendungen auf mobilen Geräten, als *Kommunikationsmedium*.
- andererseits auch als *Testbed* zur Erforschung verschiedener Erweiterungen von WMNs.

Dieses WMN soll beispielsweise der Untersuchung neuartige kontextbezogener Kommunikationsmechanismen, der Erforschung von Publish/Subscribe-Diensten für WMNs oder der Verwaltung von Umgebungsmodellen innerhalb eines hybriden Systems wie es ein WMN darstellt, dienen.

Ziel dieser Fachstudie ist die Ausarbeitung einer Empfehlung für die Beschaffung entsprechender Geräte (*Hardwareplattformen und Systemsoftware*) für den Aufbau eines WMN.

Das Vorgehen umfasst im einzelnen:

- Einarbeitung in grundlegende WMN-Technologien
- Analyse der Anforderungen des Nexus-Projektes an ein WMN
- Erstellung einer Übersicht über aktuelle verfügbare Hardwareplattformen und Systemsoftware für WMN
- Bewertung der analysierten Systeme hinsichtlich der ermittelten Anforderungen
- Ausarbeitung einer Empfehlung für eine geeignete WMN hinsichtlich Hardwareplattform und Systemsoftware

3 Anforderungen

(TODO) korrigieren, vielleicht ganze Sätze machen!

Nach der Einarbeitung in WMN-Technologien und Analyse der Anforderungen des Nexus-Projektes wurde folgendes festgehalten:

3.1 Anforderungen an Hardware

- IEEE 802.11a kompatibel (5Ghz Frequenzen)
- Ad-hoc Modus (auch bei 5Ghz Frequenzen)
- Treiber für Linux (und Windows)
- Open-Source Firmware (nur für Router)
- Zusätzlich zu Wireless Mesh Network auch weitere (Netzwerk-)Schnittstelle zur Verwaltung vorsehen
- Nach Möglichkeit keine selber gebastelten Lösungen
- Schön wäre, die angestrebte Standardisierung von Mesh-Netzen zu unterstützen (MESH STANDARD 11n Draft als Vorteil)
- 2 Mini PCI-Slots (nur für Router) - Funk auf verschiedenen Frequenzbandern (Performanceverbesserung)

3.2 Anforderungen an Software

- Betriebssystem ist nicht festgelegt, soll Ergebnis der Fachstudie sein
- Freiheit bei Routingprotokollen (Austauschbarkeit)
- Software soll Konfigurierung und Instrumentierung von WMN ermöglichen
- Topologie verändern bzw. erfassen
- Visualisierung von Anfragen und Topologie

3.3 Zusätzliche Anforderungen

- Abdeckung des Gebäudes Universitätsstraße 38
- Verbindung mit Informatik-Netz nur über Gateway mit strikter Filterung (nur eine Richtung (UNI->Mesh) für die Verwaltung)
- Separates Gateway notwendig? Vermutlich sinnvoll.
- AUFWANDSCHATZUNG (wie viele Knoten usw.)
- Budget max. 25.000 Euro (evtl. mehr in Zukunft)
- 4-5 Mesh-Knoten pro Quadrat
- Speicherkapazität wichtig (nur bei Router)

- FOCUS: PC + Wlan-Karten, da mehr Flexibilität
- PC + 2 Wlan-Karten vorteilhaft
- PDAs (bzw. andere kleine Clients) mit 802.11a

4 Hardware-Lösungen für den Aufbau eines Mesh-Netzwerkes

Es gibt verschiedene Möglichkeiten ein Meshnetzwerk aufzubauen. Im Weiteren werden einige davon im Detail beschrieben. Die Vorteile und Nachteile von einzelnen Möglichkeiten werden ebenfalls erläutert.

4.1 PCs + WLAN-Karten

(TODO) korrigieren, vielleicht vorteile und Nachteile erweitern..

Die einfachste Möglichkeit wäre die herkömmlichen PCs mit WLAN-Karten zu einem Mesh-Router einzurichten. Man nimmt dabei einfach die WLAN-Karten (PCI, Mini-PCI oder PCMCIA) und baut diese in PCs oder in Laptops ein.

Generelles Problem dabei ist: *Ad-Hoc Modus bei Karten im 5Ghz Bereich ist von unausgereift bis nicht vorhanden.* (LINK) nachteile 5Ghz

Hersteller haben gespart an der Entwicklung, da Ad-Hoc Modus einigermaßen kompliziert ist, und alle meist nur Infrastrukturmodus benutzt haben. Fehler liegen in Firmware von Chipsatz und im Treiber.

Es gibt einen MadWiFi-Treiber, der für eine Vielzahl von Chipsätzen entwickelt wurde und mit dem sollte es einigermaßen funktionieren, sobald dieser noch zusätzlich gepatcht ist, und Firmware der Karte Ad-Hoc zulässt. Generell wegen der geringen Verbreitung von 802.11a in Europa, sind nur wenige Karten erhältlich. Z.B konnten Karten mit Atheros Chipsatz, z.B AR5004X, uns weiterhelfen.

Vorteile:

- Sehr Flexibel (Hardware kann noch nützlich sein)
- Installation ist relativ einfach
- Software Unterstützung ist vorhanden
- Mehrere WLAN und Ethernet Interfaces möglich

Nachteile:

- Groß und stationär
- Stromversorgung
- Schlechte Sende- und Empfangqualität, da sich die Antenne im elektromagnetischen Stromnebel des PCs befindet

4.1.1 PCI-WLAN-Karten

(TODO) mehr Text?

PCI-WLAN-Karten werden in einen freien PCI-Steckplatz des Mainboards gesteckt. Ein Vorteil von PCI-WLAN-Karten ist die bessere Stabilität im Betrieb. Weiterhin besitzen die meisten PCI-WLAN-Karten die Möglichkeit die mitgelieferte Antenne gegen eine andere zu tauschen. Zu beachten ist, dass die Antenne üblicherweise direkt hinten an der Karte angebracht ist und somit in unmittelbarer Nähe zum PC-Gehäuse ist. Dies kann jedoch negative Auswirkungen auf die Reichweite oder den Datendurchsatz haben. Deshalb kann es für eine bessere Verbindung notwendig sein, die Antenne mit einem Koaxialkabel vom Rechnergehäuse zu entfernen.

Vorteile:

- Bessere Stabilität im Betrieb
- Meistens abschraubbare Antenne
- Verschwinden im Gehäuse, Platz wird nicht verschwendet

Nachteile:

- Oft recht schlechte Empfangs/Sendeleistung, weil die kleine Antenne ja direkt hinten am Rechner rauskommt (Lösung: zusätzliche Antenne)

4.1.1.1 Linksys WMP55AG



Abbildung 1: Linksys WMP55AG

Chipsatz:

- Atheros AR5213A

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- LEAP

Treiber:

- Sehr gute Linux-Unterstützung, MadWifi-Treiber funktioniert mit dieser WLAN PCI-Karte ohne Probleme. Windows-Treiber werden von Linksys bereitgestellt.

Preis:

- ca. 90 Euro

Installation:

- Lasst sich leicht sowohl unter Windows als auch unter Linux (MadWifi-Treiber) installieren.

<http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- <http://madwifi.org/wiki/Compatibility/Linksys>
- <http://forums.fedoraforum.org/showthread.php?t=91165>
- http://www.pcworld.com/product/specs/prtprdid,704176/wireless_ag_54mbps_pci_adptr_80211a80211b80211g_compatible.html
- http://www.linksys.com/servlet/Satellite?c=L_CASupport_C2&childpagename=US%2FLayout&cid=1169671168007&pagename=Linksys%2FCommon%2FVisitorWrapper&lid=6800768007N09

4.1.1.2 Netgear WAG311



Abbildung 2: Netgear WAG311

Chipsatz:

- Atheros AR5212

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA, WPA-PSK
- PPTP, P2TP, IPSec VPN pass-through

Treiber:

- Sehr gute Linux-Unterstützung, MadWifi-Treiber funktioniert mit dieser WLAN PCI-Karte ohne Probleme.

Preis:

- ca. 50-60 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>
http://www.packetpro.com/~peterson/linux-netgear_wg311t_pci.html

Weitere Informationen:

- Externe Antenne, die mit der WLAN-PCI-Karte durch langes Kabel verbunden ist. Das Kabel lässt sich nicht von der PCI-Karte trennen.

Links:

- <http://www.netgear.com/Products/Adapters/AGDualBandWirelessAdapters/WAG311.aspx>
- <http://madwifi.org/wiki/Compatibility/Netgear>
- <http://www.linuxquestions.org/questions/mandriva-30/using-netgear-wag311-via-ma>
- http://www.packetpro.com/~peterson/linux-netgear_wg311t_pci.html
- http://www.netgear.com/upload/product/wag311/enus_ds_wag311.pdf

4.1.1.3 D-Link DWL-A520



Abbildung 3: D-Link DWL-A520

Chipsatz:

- Atheros AR5210

IEEE Standards:

- 802.11a

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)

Treiber:

- Von D-Link werden nur Treiber für Windows bereitgestellt. Sehr gute Linux-Unterstützung, MadWifi-Treiber funktioniert mit dieser WLAN PCI-Karte ohne Probleme.

Preis:

- ca. 70-80 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Weitere Informationen:

- Antenne ist nicht abschraubbar.

Links:

- <http://support.dlink.com/products/print.asp?productid=DWL-A520>
- <http://madwifi.org/wiki/Compatibility/D-Link>

4.1.1.4 Gigabyte GN-WPEAG

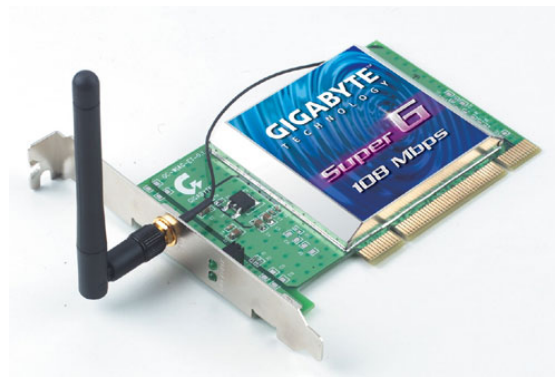


Abbildung 4: Gigabyte GN-WPEAG

Chipsatz:

- Atheros AR5212

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- WPA2

Treiber:

- Von Gigabyte werden nur Treiber für Windows bereitgestellt.
http://www.gigabyte.com.tw/Support/Communication/Driver_Model.aspx?ProductID=952
Sehr gute Linux-Unterstützung, MadWifi-Treiber funktioniert mit dieser WLAN PCI-Karte ohne Probleme.
<http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Preis:

- ca. 70-80 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Weitere Informationen:

- Abschraubbare Antenne mit reversed SMA. Eigentlich ist das eine Mini-PCI-Karte mit PCI-Adapter.

Links:

- http://www.gigabyte.com.tw/Products/Communication/Products_Spec.aspx?ProductID=952
- http://www.gigabyte.com.tw/Support/Communication/Driver_Model.aspx?ProductID=952
- <http://madwifi.org/wiki/Compatibility/Gigabyte>

4.1.1.5 Andere PCI-WLAN-Karten

- Intel PRO/Wireless 5000
Chipsatz Intel, 802.11a WLAN PCI-Karte, unterstützt Ad-Hoc- und Infrastruktur-Modus, Treiber von Intel nur für Windows vorhanden, für Linux werden keine Treiber entwickelt, kostet ca. 200 Euro
<http://support.intel.com/support/wireless/wlan/pro5000/pciadapter>
ftp://download.intel.com/support/wireless/wlan/pro5000/PR05000_INF0.pdf
- D-Link DWL-AG530
Chipsatz Atheros AR5212 oder AR5213, 802.11a/b/g WLAN-Karte, MadWifi-Treiber Unterstützung, Externe abschraubbare Antenne, kostet ca 80 Euro
<http://www.dlink.com/products/?pid=306>
<http://madwifi.org/wiki/Compatibility/D-Link>
- D-Link DWL-G550
Chipsatz Atheros AR5212, 802.11a/b/g WLAN-Karte, MadWifi-Treiber Unterstützung, Externe abschraubbare Antenne, kostet ca 60 Euro
<http://www.dlink.com/products/?pid=414>
<http://madwifi.org/wiki/Compatibility/D-Link>

4.1.2 Mini-PCI WLAN-Karten

Mini-PCI ist eine vor allem für die Nutzung in Notebooks und Laptops miniaturisierte Version des PCI Steckplatzes, wie er in allen Desktop PCs vorkommt. PCI steht dabei für Peripheral Component Interconnect. Die Abmessungen einer Mini-PCI Card betragen 6,0 x 4,6 x 0,5 cm.

Mini-PCI WLAN-Karten sind ursprünglich für Laptops gedacht, sind aber mit entsprechenden Adaptoren (PCI-zu-MiniPCI) und externen Antennen auch im normalen PCs zu verwenden. Als Vorteil ist dabei die Flexibilität zu nennen. Als Nachteil - die Zusätzliche Kosten und Installationen. Meist sind Mini-PCI Cards für Wireless LAN bereits vom Hersteller eingebaut. Der Vorteil der Ausführung als standardisiertes Modul liegt darin, daß eine Mini-PCI Card in aller Regel einfach gegen eine andere Card - auch eines anderen Herstellers - ausgetauscht werden kann. Im Falle der WLAN Mini-PCI Module kann z.B. problemlos vom langsameren 802.11b Standard auf ein schnelleres WLAN Modul nach 802.11g gewechselt werden.

Vorteile:

- Können mit Hilfe eines Adapters zu einer PCI-WLAN-Karte umgebaut werden
- Können leicht ausgetauscht werden

Nachteile:

- Meistens kostenintensiv
- Installationen

4.1.2.1 Wistron CM9 Atheros AR5213A



Abbildung 5: Wistron CM9 Atheros AR5213A

Chipsatz:

- Atheros AR5213A

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- WPA2

Treiber:

- Hervorragende Unterstützung von MadWifi-Treiber, auch Ad-Hoc-Modus.
<http://madwifi.org/>

Preis:

- ca. 40 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- <http://www.alix-board.de/produkte/wistroncm9.html>
- <http://www.pcengines.ch/cm9.htm>
- <http://forum.openwrt.org/viewtopic.php?pid=10213>
- <http://madwifi.org/>
- <http://madwifi.org/ticket/1209>

4.1.2.2 Intel PRO/Wireless 3945



Abbildung 6: Intel PRO/Wireless 3945

Chipsatz:

- Intel

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-bit)
- WPA
- WPA2

Treiber:

- Es werden von Intel Treiber sowohl für Windows als auch für Linux bereitgestellt.
http://downloadcenter.intel.com/Product_Filter.aspx?ProductID=2259
Von Intel wurde ein Projekt für die Unterstützung von Intel PRO/Wireless 3945 erstellt.
<http://ipw3945.sourceforge.net>

Der ipw3945-Treiber funktioniert auch im Ad-Hoc-Modus, aber nicht sehr stabil, es kommt oft zu Verbindungsabbrüchen.

Preis:

- ca. 20-30 Euro

Installation:

- Im Gegensatz zu den „klassischen“ Intel Wireless-Chipsätzen 2100- und 2200BG-Chipsätzen ist der Treiber für den 3945ABG noch nicht im Kernel verfügbar. Um auch damit kabellos ins Internet zu gehen, sind ein paar Handgriffe notwendig.

<http://ipw3945.sourceforge.net/README.ipw3945>

<http://ipw3945.sourceforge.net/INSTALL>

Links:

- http://www.intel.com/network/connectivity/products/wireless/prowireless_mobile.htm
- http://downloadcenter.intel.com/Product_Filter.aspx?ProductID=2259
- <http://ipw3945.sourceforge.net/>
- <http://ipw3945.sourceforge.net/README.ipw3945>
- <http://ipw3945.sourceforge.net/INSTALL>

4.1.2.3 Intel PRO/Wireless 2915



Abbildung 7: Intel PRO/Wireless 2915

Chipsatz:

- Intel

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-bit)
- WPA
- WPA2

Treiber:

- Es werden von Intel Treiber sowohl für Windows als auch für Linux bereitgestellt.

http://downloadcenter.intel.com/Product_Filter.aspx?ProductID=1847

Von Intel wurde ein Projekt für die Unterstützung von Intel PRO/Wireless 2915 erstellt.

<http://ipw2200.sourceforge.net>

Der ipw2200-Treiber funktioniert auch im Ad-Hoc-Modus, aber nicht sehr stabil, es kommt oft zu Verbindungsabbrüchen. Der ipw2200-Treiber ist im Kernel 2.6 enthalten, kann aber auch separat als Modul kompiliert werden. Der im Kernel enthaltene Treiber unterstützt den Monitor-Modus nicht.

Preis:

- ca. 30 Euro

Installation:

- <http://ipw2200.sourceforge.net/README.ipw2200>
<http://ipw2200.sourceforge.net/INSTALL>

Links:

- <http://support.intel.com/support/wireless/wlan/pro2915abg>
- http://download.intel.com/support/wireless/wlan/pro2915abg/sb/303330002us_channel.pdf
- <http://ipw2200.sourceforge.net/>
- <http://www.intel.com/cd/personal/computing/emea/deu/234998.htm>
- http://downloadcenter.intel.com/Product_Filter.aspx?ProductID=1847

4.1.2.4 Intel Wireless WiFi Link 4965AGN



Abbildung 8: Intel Wireless WiFi Link 4965AGN

Chipsatz:

- Intel

IEEE Standards:

- 802.11a/b/g/n(draft)

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-bit)
- WPA
- WPA2

Treiber:

- <http://www.intellinuxwireless.org/>

Preis:

- ca. 30 Euro

Installation:

- <http://www.intellinuxwireless.org/>

Links:

- http://www.intel.com/network/connectivity/products/wireless/wireless_n/overview.htm
- <http://www.intellinuxwireless.org/>
- <http://www.wifi-info.de/intel-kuendigt-11n-chipsatz-fuer-centrino-notebooks-an/01/2007/>
- http://downloadcenter.intel.com/filter_results.aspx?strTypes=all&ProductID=2753&OSFullName=Linux*&lang=eng&strOSs=39&submit=Go%21

4.1.3 PCMCIA WLAN-Karten

Diese WLAN-Karten sind für Notebooks gedacht. Heutzutage ist es jedoch üblich, dass die Notebooks schon ein integriertes WLAN-Modul (Mini-PCI) eingebaut haben. Damit ist die Notwendigkeit dieser Module nur noch für Notebooks älterer Generationen notwendig. Die meisten Module haben keinen Anschluss für eine externe Antenne.

Vorteile:

- Leichte Installation
- Austauschbarkeit

Nachteile:

- Veraltet
- Keinen Anschluss für eine externe Antenne

4.1.3.1 Proxim Orinoco Gold 8480-WD



Abbildung 9: Proxim Orinoco Gold 8480-WD

Chipsatz:

- Atheros AR5212

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-bit)
- WPA
- WPA2

Treiber:

- Unter Linux hervorragende Unterstützung von MadWifi-Treiber, auch Ad-Hoc-Modus.
<http://madwifi.org/>

Preis:

- ca. 80 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- <http://www.proxim.com/products/wifi/client/abgcard/index.html>
- <http://madwifi.org/wiki/Compatibility/Proxim>

4.1.3.2 Netgear WAG511



Abbildung 10: Netgear WAG511

Chipsatz:

- Atheros AR5001X+

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- WPA2
- PTP, P2TP, IPSec, VPN pass-through

Treiber:

- Von Netgear werden nur Windows Treiber angeboten.
<http://www.netgear.de/de/Support/download.html?func=Detail&id=10676>
Unter Linux hervorragende Unterstützung von MadWifi-Treiber, auch Ad-Hoc-Modus.
<http://madwifi.org/>

Preis:

- ca. 50-60 Euro

Installation:

- http://www.lrz-muenchen.de/services/netz/mobil/funklan-installation/installation/windowsxp_wg511/flan-instwpx.html
<http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- <http://www.netgear.com/Products/Adapters/AGDualBandWirelessAdapters/WAG511.aspx>
- <http://www.netgear.de/de/Support/download.html?func=Detail&id=10676>
- <http://www.netgear.de/Produkte/Wireless/DualBand/WAG511/index.html>
- <http://madwifi.org/wiki/Compatibility/Netgear>

4.1.3.3 SMC 2536W-AG



Abbildung 11: SMC 2536W-AG

Chipsatz:

- Atheros AR5001

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- WPA2

Treiber:

- Von SMC werden nur Treiber für Windows angeboten.
http://www.smc.com/index.cfm?event=downloads.searchResultsDetail&localeCode=EN_USA&productCategory=9&partNumber=2916&modelNumber=348&knowsPartNumber=false&userPartNumber=&docId=3103
Unter Linux hervorragende Unterstützung von MadWifi-Treiber, auch Ad-Hoc-Modus.
<http://madwifi.org/>

Preis:

- ca. 80 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- <http://www.smc.com/index.cfm?event=viewProduct&cid=9&scid=49&localeCode=EN%5FUSA&pid=348>
- http://www.smc.com/files/AC/2536Wag_Ds_ww.pdf
- <http://madwifi.org/wiki/Compatibility/SMC>
- <http://forums.fedoraforum.org/archive/index.php/t-101517.html>

4.1.3.4 Linksys WPC55AG



Abbildung 12: Linksys WPC55AG

Chipsatz:

- Atheros AR5212 oder AR5006X

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Sicherheit:

- WEP (40-, 104-, 128-bit)
- WPA
- WPA2

Treiber:

- Von Linksys werden nur Treiber für Windows bereitgestellt.
http://www.linksys.com/servlet/Satellite?c=L_Product_C2&childpagename=US%2FLayout&cid=1115416827328&pagename=Linksys%2FCommon%2FVisitorWrapper
 Auch hier kann man Treiber für Windows finden:
<http://www.phoenixnetworks.net/atheros.php>
 Unter Linux hervorragende Unterstützung von MadWifi-Treiber, auch Ad-Hoc-Modus.
<http://madwifi.org/>

Preis:

- ca. 50-60 Euro

Installation:

- <http://madwifi.org/wiki/UserDocs/FirstTimeHowTo>

Links:

- http://www.linksys.com/servlet/Satellite?c=L_Product_C2&childpagename=US%2FLayout&cid=1115416827328&pagename=Linksys%2FCommon%2FVisitorWrapper
- <http://www.phoenixnetworks.net/atheros.php>
- <http://madwifi.org/>
- <http://madwifi.org/wiki/Compatibility/Linksys>
- http://reviews.cnet.com/adapters-nics/linksys-wpc55ag-dual-band/4505-3380_7-21128291.html
- <http://www.google.de/search?q=Linksys+WPC55AG>
- <http://lists.funkfeuer.at/pipermail/discuss/2006-September/001592.html>
- <http://www.uk-surplus.com/manuals/brochures/linksyswpc55duser.pdf>

4.1.3.5 Andere PCMCIA-WLAN-Karten

- Intel PRO/Wireless 5000
Chipsatz Intel, 802.11a WLAN PCMCIA-Karte, unterstützt Ad-Hoc- und Infrastruktur-Modus, Treiber von Intel nur für Windows vorhanden, für Linux werden keine Treiber entwickelt, kostet ca. 150 Euro
<http://www.intel.com/support/wireless/wlan/pro5000/lancardbus>
- Netgear WAB501 Chipsatz Atheros AR5211, 802.11a/b WLAN-Karte, MadWifi-Treiber Unterstützung
<http://kbserver.netgear.com/products/WAB501.asp>
<http://madwifi.org/wiki/Compatibility/Netgear>
- Netgear WG511U
Chipsatz Atheros AR5004X, 802.11a/g WLAN-Karte, MadWifi-Treiber Unterstützung
<http://www.netgear.com/Products/Adapters/AGDualBandWirelessAdapters/WG511U.aspx>
<http://madwifi.org/wiki/Compatibility/Netgear>

- Proxim Orinoco Silver 8481-WD
Chipsatz Atheros AR5001X+, 802.11a/b/g WLAN-Karte, MadWifi-Treiber Unterstützung, kostet ca 80-90 Euro
<http://www.proxim.com/products/wifi/client/abgcard/index.html>
<http://madwifi.org/wiki/Compatibility/Proxim>
- Cisco Aironet CB21AG
Chipsatz Atheros 5212, 802.11a/b/g WLAN-Karte, Madwifi-Treiber Unterstützung, kostet ca 100 Euro
<http://madwifi.org/wiki/Compatibility/Cisco>

4.2 WLAN-Router

Die Kombination aus Access Point und Router wird häufig als WLAN-Router bezeichnet. Das ist solange korrekt, soweit es einen WAN-Port gibt. Das Routing findet dann zwischen WLAN und WAN (und falls vorhanden auch zwischen LAN und WAN) statt. Fehlt dieser WAN-Port, handelt es sich hier lediglich um Marketing-Begriffe, da reine Access Points auf OSI-Ebene 2 arbeiten und somit Bridges und keine Router sind. Oft sind das aber keine vollständigen Router, da diese Geräte ausschließlich als Internetzugangs-Systeme dienen und nur mit aktiviertem PPPoE (oder PPPoA) sowie NAT-Routing (oder IP-Masquerading) eingesetzt werden können.

In diesem Abschnitt werden so genannte *stand-alone* Lösungen für eine WMN vorgestellt.

4.2.1 SoHo-Router

Man kann herkömmliche WLAN-Router für Heimanwender (SoHO-Router - small or home office) kaufen, die sich mit alternativer Firmware (spezielle Linux-Software mit OLSR-daemon) zu einem Mesh-Router umrüsten lassen. Ein WLAN-Router ist die Kombination eines normalen Routers (Kabelrouter) mit einem Accesspoint. Es gibt solche mit eingebauten Modem und andere mit einem Anschluss (WAN-Port) dafür (für Modems mit LAN-Anschluss). Ein Nachteil ist, dass es viele Modelle gibt, die eine fix verbaute Antenne haben, die nicht gewechselt werden kann.

(TODO) Weitere Vorteile und eventuell Nachteile..

Vorteile:

- Klein und handlich
- Mobil und flexibel
- Relativ günstig
- Gute Reichweite
- Wenig Stromverbrauch

Nachteile:

- Meistens fix verbaute Antenne
- Eingeschränkte Software Unterstützung
- Durch das Öffnen von Geräten und das Einspielen von fremder Firmware erlischt die Garantie des Herstellers

4.2.1.1 Linksys WRT54G v1.0



Abbildung 13: Linksys WRT54G v1.0

IEEE Standards:

- 802.11b/g
- 802.11a/b/g (wenn man die mitgelieferte Mini-PCI WLAN-Karte durch z.B. Atheros 802.11a/b/g WLAN-Karte austauscht)

Betriebsart:

- Ad-Hoc
- Infrastruktur

Firmware:

- Es gibt mehrere fremde frei verfügbare Firmware für dieses Gerät. Alle unten aufgeführten Firmware sind Open-Source Projekte:

OpenWRT

<http://wiki.openwrt.org/OpenWrtDocs/Hardware/Linksys/WRT54G>

DD-WRT

http://www.dd-wrt.com/wiki/index.php/Linksys_WRT54G/GL/GS/GX

Preis:

- ca. 40-50 Euro

Installation:

- Die mitgelieferte Mini-PCI WLAN-Karte durch z.B. Atheros 802.11a Mini-PCI austauschen und oben erwähnte frei verfügbare Firmware installieren (siehe oben Firmware).

Weitere Informationen:

- Ein Mini-PCI Slot ist für eine WLAN-Karte vorhanden.

Links:

- <http://wiki.openwrt.org/OpenWrtDocs/Hardware/Linksys/WRT54G>
- http://www.dd-wrt.com/wiki/index.php/Linksys_WRT54G/GL/GS/GX
- <http://forum.opennet-initiative.de/thread.php?threadid=505&sid=56c53647db6353a4>
- <http://www.linksysinfo.org/forums/showthread.php?t=47124>

4.2.1.2 Linksys WRT55AG



Abbildung 14: Linksys WRT55AG

IEEE Standards:

- 802.11a/b/g

Betriebsart:

- Ad-Hoc
- Infrastruktur

Firmware:

- Open-Source Firmware befindet sich noch in Entwicklung:
Modifizierte Version von OpenWRT Kamikaze
<http://legacy.not404.com/cgi-bin/trac.fcgi/wiki/OpenWRT/Atheros/Linksys/WRT55AGv2#KamikazeKernelonWRT55AGv2>
OpenWRT
<http://wiki.openwrt.org/OpenWrtDocs/Hardware/Linksys/WRT55AG>

Preis:

- ca. 70-80 Euro

Weitere Informationen:

- 2 Slots sind für Mini-PCI WLAN-Karten vorhanden.

Links:

- <http://wiki.openwrt.org/OpenWrtDocs/Hardware/Linksys/WRT55AG>
- http://www.tomsnetworking.de/content/tests/j2003a/test_linksys_wrt55ag/index.html
- http://reviews.cnet.com/routers/linksys-wrt55ag-wireless-a/4505-3319_7-21131921.html
- <http://legacy.not404.com/cgi-bin/trac.fcgi/wiki/OpenWRT/Atheros/Linksys/WRT55AGv2>

4.2.1.3 Asus WL500G/GP



Abbildung 15: Asus WL500G/GP

IEEE Standards:

- 802.11b/g
- 802.11a/b/g (wenn man die mitgelieferte Mini-PCI WLAN-Karte durch z.B. Atheros 802.11a/b/g WLAN-Karte austauscht)

Betriebsart:

- Ad-Hoc
- Infrastruktur

Firmware:

- Es sind mehrere fremde frei verfügbare Firmware für dieses Gerät. Alle unten aufgeführten Firmware sind Open-Source Projekte:
 OpenWRT
<http://wiki.openwrt.org/OpenWrtDocs/Hardware/Asus/WL500G>
<http://wiki.openwrt.org/OpenWrtDocs/Hardware/Asus/WL500GP>
 FreeWRT
<http://www.freewrt.org/trac/wiki/Documentation/Hardware/AsusWL500G>
<http://www.freewrt.org/trac/wiki/Documentation/Hardware/AsusWL500GP>
 Olegs custom firmware
<http://oleg.wl500g.info>

Preis:

- ca. 70-80 Euro

Installation:

- Die mitgelieferte Mini-PCI WLAN-Karte durch z.B. Atheros 802.11a Mini-PCI austauschen und oben erwähnte frei verfügbare Firmware installieren (siehe oben Firmware).

http://wiki.opennet-initiative.de/index.php/Mini-PCI_Umbau

Weitere Informationen:

- Ein Mini-PCI Slot ist für eine WLAN-Karte vorhanden.

Links:

- <http://wiki.opennet-initiative.de/index.php/AP9>
- <http://wiki.openwrt.org/OpenWrtDocs/Hardware/Asus/WL500G>
- <http://wiki.openwrt.org/OpenWrtDocs/Hardware/Asus/WL500GP>
- <http://www.freewrt.org/trac/wiki/Documentation/Hardware/AsusWL500G>
- <http://www.freewrt.org/trac/wiki/Documentation/Hardware/AsusWL500GP>
- <http://wl500g.dyndns.org/>
- <http://oleg.wl500g.info/>
- <http://au.asus.com/products.aspx?l1=12&l2=43>
- http://www.freifunk-bno.de/component/option,com_smf/Itemid,88/topic,910.msg10357/
- http://www.cyber-wulf.de/a_wl500g.html
- <http://wiki.openwrt.org/OpenWrtDocs/Hardware/Asus/WL500G>
- <http://forum.opennet-initiative.de/print.php?threadid=505&page=6&sid=460903353d70c65fad4960105ab76cdd>
- <http://forum.openwrt.org/viewtopic.php?pid=41756>
- <http://www.familie-prokop.de/asus-wl500gp/index.html>

4.2.1.4 Andere WLAN-Router

- Netgear HR314
802.11a WLAN-Router, unterstützt Ad-Hoc- und Infrastruktur-Modus, keine Open-Source Firmware vorhanden, kostet ca. 30 Euro
<http://www.wi-fiplanet.com/reviews/article.php/1559091>

4.2.2 Professionelle Router

In diesem Abschnitt werden so Professionelle Mesh-Router betrachtet. Die Begriffe, die dafür oft als Synonyme verwendet werden, sind dabei:

- Routerboards
- Stand-alone Mesh-Router
- Minicomputers
- Single-Board-Computers (SBC)
- Access Points (AP)

Im Projekt (<http://umic-mesh.net>) wurden professionelle Router eingesetzt, das sind spezielle Router-Boards mit Steckplätzen für MiniPCI WLAN-karten. Boards kosten etwa 100-200 Euro, dazu muss man allerdings noch passende WLAN-Karten kaufen + Antennen + Kabel + Netzteil + Gehäuse, also keine billige Lösung.

Man könnte aber nur diese Karten kaufen + Adapter PCI-MiniPCI und in Rechner einbauen (Das wäre dann die platzsparende Version von *PCs + WLAN-Karte*). WLAN-Karten z.B Wistron Neweb CM9 Atheros 802.11a/b/g Mini-PCI, hier (<http://www.pcengines.ch/cm9.htm>), und Boards sind hier (<http://www.pcengines.ch/wrap.htm>, <http://www.pcengines.ch/alix.htm>).

Es gibt noch diese kleine Mesh-Router, wie von Meraki. Die haben wohl ihre eigene Firmware drin und eigene Routingprotokolle oder eigene Implementierungen davon besser gesagt. Hier ein Paar, die 802.11a unterstützen, sind aber outdoor, haben also große Reichweiten. Ob es sinnvoll ist, sie im Gebäude einzusetzen:

- Aphelion 3300AG Outdoor Wireless Access Point - 802.11a/b/g
- Aphelion 600AG/605AG Intelligenter sequentieller Wireless Access Point für den Außenbereich mit den Standards 802.11a/b/g (http://www.abcddata.de/abcdataneu/WLAN_MESH_Aphelion.php)
- PLANET MAP-2100 - indoor - sind aber zum Teil sehr teuer (1200 Euro)

Vorteile:

- Outdoor (in unserem Fall irrelevant)
- Große Reichweiten

Nachteile:

- Sehr teuer
- Close source

Links:

- <http://wiki.opennet-initiative.de/index.php/WRAP>
- http://www.abcddata.de/abcdatanew/WLAN_MESH_Aphelion.php
- http://www.aerial.net/shop/product_info.php?cPath=33&products_id=351
- <http://forum.openwrt.org/viewtopic.php?id=9655>

4.2.3 Access Points

Ein WLAN-Accesspoint ist der Verbindungspunkt eines kabelbasierten Netzwerkes zu einem WLAN. Der Accesspoint ist Basisstation für alle WLAN-Clients, zu der sie eine drahtlose Verbindung aufbauen. Sendet ein WLAN-Client Daten, die für einen Empfänger im kabelbasierten Netzwerkteil bestimmt sind, so *reicht* der Accesspoint diese Daten über das Kabelnetz an den Empfänger weiter. Weiterhin kann ein Accesspoint auch mehrere WLAN-Clients untereinander verbinden. Somit ist der Accesspoint quasi ein kabelloser Switch.

Dieser hat (je nach Ausstattung) einige der folgenden Optionen:

- Ein oder mehrer integrierte WLAN-Module
- Einen integrierten DHCP-Server
- Umfangreiche Sicherheits- und Verschlüsselungsmöglichkeiten (WEP, WPA und WPA2 dienen der Verschlüsselung der zu ubetragenden Daten ; MAC-Filter und SSID Optionen ; Einstellungen bezüglich dem Remotezugriff)
- Verschiedene Arbeitsmodi (Accesspoint (AP), Bridge (Point-to-Point oder Point-to-Multipoint), Repeater, MESSID)

(TODO) Hier sind ein paar kandidaten: ..

Intel PRO/Wireless 5000

- <http://support.intel.com/support/wireless/wlan/pro5000/accesspoint>

- <http://www.pcmag.com/article2/0,1759,5524,00.asp>

Linksys WAP55AG

- http://www.tomsnetworking.de/content/aktuelles/news_beitrag/news/851/6/index.html

NETGEAR WAB102

- <http://kbserver.netgear.com/products/WAB102.asp>
- http://reviews.cnet.com/wireless-access-points/netgear-wab102-802-11a/4505-3265_7-20708150.html
- <http://archive.cert.uni-stuttgart.de/bugtraq/2003/12/msg00159.html>

4.3 PDAs und Handys

(TODO) einbischen Text zu jedem Punkt.. (aus Links)

Links:

- http://www.tecchannel.de/kommunikation/handy_pda/402279/
- <http://www.golem.de/0701/49894.html>
- http://www.computerbase.de/news/hardware/organizer_pdas/2005/november/asus_mypal_a636_pda_gps_wlan/
- <http://www.worldofppc.com/HWTests/oqo02.htm>
- http://www.dooyoo.de/handy-mit-vertrag/_ieee-802-11a/

5 Systemsoftware für Mesh-Netzwerk

Im Kapitel 4 haben wir mehrere Hardware-Lösungen für ein Mesh-Netzwerk vorgestellt. Für alle im Kapitel 4 vorgestellten PCI-, MiniPCI- und PCMCIA-Karten werden vom Hersteller dieser WLAN-Karten Windows-Treiber bereitgestellt. Und nur wenige Hersteller (z.B Intel) haben auch Linux-Treiber für ihre Karten implementiert. Glücklicherweise existiert für WLAN-Karten mit Atheros-Chipsatz der Open-Source Linux-Treiber MadWifi, der alle im Kapitel 4 aufgelisteten WLAN-Karten mit Atheros-Chipsatz unterstützt. Bei SoHo-Routern sieht das Ganze etwas anders aus. Hier gibt es nicht so viele Möglichkeiten bei der Wahl nach einer Software. Die meisten Hersteller von SoHo-Routern stellen den Source-Code des Betriebssystems für SoHo-Router nicht bereit. Deswegen müssen wir nach Open-Source Firmware greifen, wenn wir SoHo-Router im Ad-Hoc Modus betreiben wollen, denn die meisten SoHo-Router nur im Infrastruktur Modus arbeiten und Ad-Hoc Modus nicht realisieren.

In diesem Kapitel wird verschiedene Treiber- und Routing-Software vorgestellt, die zusammen mit der Hardware aus Kapitel 4 die Realisierung von Ad-Hoc Mesh-Netzwerken ermöglicht.

5.1 Linux MadWiFi-Treiber

Linux MadWifi-Treiber ist Linux Kernel Treiber für WLAN-Karten mit Atheros Chipsatz. Linux MadWifi-Treiber ist heutzutage einer der fortgeschrittensten Linux Treiber für WLAN-Karten. Der Treiber ist stabil und hat eine große Benutzergemeinschaft. Der MadWifi-Treiber selbst ist Open-Source, verwendet aber eine proprietäre Softwareschicht Hardware Abstraction Layer (HAL), die nur in binärer Form vorhanden ist.

Das Hardware Abstraction Layer (HAL) wird vom MadWifi-Treiber gebraucht, um die Atheros-Chips ansprechen zu können. Dafür wurde bisher ein Closed-Source-Modul verwendet. Dies hat unter anderem damit zu tun, dass die Atheros-Chipsätze prinzipiell auf Frequenzen funken könnten, für die sie nicht zugelassen sind - beispielsweise weil diese vom Militär zur Kommunikation verwendet werden.

Durch das proprietäre Modul war der Madwifi-Treiber bisher jedoch von einer Aufnahme in den Linux-Kernel ausgeschlossen. Die Entwickler hatten außerdem das Problem, dass sie Fehler unter Umständen nicht beheben konnten, da sie nicht nachvollziehen konnten, wie der HAL-Baustein arbeitet.

MadWifi selbst wird daher ab sofort nicht weiterentwickelt. Stattdessen setzen die Programmierer auf OpenHAL, eine Linux-Portierung des HAL-Modules des in OpenBSD verfügbaren freien Atheros-Treibers. In der Vergangenheit wurde vom Software Freedom Law Center (SFLC) bestätigt, dass die durch Reverse Engineering entstandene Software keine Copyrights verletzt. Solche Behauptungen hatten die Entwicklung lange

ausgebremst.

Der neue Treiber „Ath5k“ wird MadWifi nun ersetzen und soll nicht nur die freie Komponente OpenHAL einsetzen, sondern auch mit dem neuen Linux-WLAN-System Mac80211 zusammenarbeiten, so dass der Treiber in den offiziellen Linux-Kernel gelangen kann. MadWifi soll jedoch weiter mit Fehlerkorrekturen und HAL-Updates versorgt werden.

Links:

- <http://madwifi.org/>
- <http://madwifi.org/wiki/About/ar5k>
- <http://madwifi.org/wiki/About/OpenHAL>
- <http://madwifi.org/wiki/UserDocs/GettingMadwifi>
- <http://madwifi.org/wiki/Compatibility>
- <http://www.intellinuxwireless.org/?p=mac80211>

5.2 OpenWRT

OpenWRT ist eine GNU/Linux-Distribution für WLAN-Router. Anstatt einer statischen Firmware setzt OpenWRT auf ein voll beschreibbares Dateisystem sowie einen Paketmanager. OpenWRT läuft unter anderem auf Geräten der Firmen Linksys, ALLNET, ASUS, Belkin, Buffalo, Microsoft und Siemens.

Vorteile:

- Flexibilität
- Erweiterbarkeit
- Individualisierbarkeit
- Sicherheit
- Gewohnte Linux-Flexibilität und Funktionsumfang!

Nachteile:

- Standardmäßig sind nur die nötigsten Unix-Tools vorhanden

Links:

- <http://openwrt.org/>
- <http://toh.openwrt.org/>

5.3 Mesh-Routing Software

In diesem Abschnitt werden Routing-Daemonen vorgestellt, die ein bestimmtes Routing-Protokoll implementieren und auf jedem Knoten in einem Mesh-Netzwerk ausgeführt werden. Diese Daemonen tauschen Routing-Informationen aus und machen es möglich, Nachrichten von einem Knoten zu einem anderen Knoten im Mesh-Netzwerk zu transportieren. Routing-Daemonen ermöglichen die Kommunikation zwischen 2 Knoten in einem Ad-Hoc Mesh-Netzwerk, zwischen denen mehr als 1 Hop liegt.

5.3.1 olsr.org OLSR daemon



Der olsr.org OLSR daemon ist eine Implementierung des Optimized Link State Routing Protokolls. OLSR ist ein Routing-Protokoll für mobile Ad-Hoc Netzwerke. Der Protokoll ist pro-aktiv, tabellengesteuert und nutzt die Technik Multipoint Relaying (MPR) zum Fluten von Nachrichten. olsrd implementiert ausserdem auch eine populäre Erweiterung Link Quality Extension. Zur Zeit ist die Implementierung von olsrd verfügbar für GNU/Linux, Windows, OS X, FreeBSD, OpenBSD and NetBSD. olsrd ist eine gut strukturierte und kodierte Implementierung, die leicht zu warten, zu erweitern und auf andere Plattformen zu portieren sein soll. Die Implementierung ist konform zu RFC3626 in Bezug auf die Kernfunktionalität und die zusätzlichen Funktionen. olsrd unterstützt das Konzept von ladbaren Plug-Ins. Mit diesen Plug-Ins kann man benutzerdefinierte Pakete mit Hilfe des OLSR MPR Flutens versenden und behandeln oder irgendeine andere zusätzliche Funktionalität bereitstellen.

Links:

- <http://www.olsr.org/>
- <http://ietf.org/rfc/rfc3626.txt>
- http://wiki.freifunk.net/OLSR_mit_Windows

- http://wireless.subsignal.org/index.php?title=Laptop_mit_OLSR
- <http://wiki.opennet-initiative.de/index.php/OLSR>

5.3.2 Open-Mesh B.A.T.M.A.N. daemon



Der B.A.T.M.A.N.-Daemon steht bislang für Linux, FreeBSD und Macintosh OS-X zur Verfügung. Die Entwicklungsarbeit konzentriert sich jedoch in erster Linie auf Linux, weshalb es vorkommen kann, dass erweiterte Funktionen unter anderen Betriebssystemen erst mit einer gewissen Verzögerung zur Verfügung stehen.

Linux-Installationspakete des B.A.T.M.A.N.-Daemon batmand gibt es für Debian, OpenZaurus und OpenWRT. Zum Kompilieren aus dem Quelltext genügt ein einfaches make und make install im Sourcecodeverzeichnis. Als einzige Abhängigkeit wird die Bibliothek libpthread vorausgesetzt, die auf einem Linux-System, üblicherweise bereits installiert sein sollte.

Um über ein B.A.T.M.A.N.-Mesh ins Internet gehen zu können, muss außerdem unter Linux das Kernelmodul tun installiert sein. Es ist im Standardkernel der Linuxdistributionen enthalten und wird beim ersten Start des B.A.T.M.A.N.-Daemons automatisch geladen. Wer einen selbstkompilierten Kernel einsetzt, findet es zum Beispiel in xconfig in der Abteilung Network Device Support unter der Bezeichnung Universal TUN/TAP device driver support.

Links:

- <https://www.open-mesh.net/batman>
- <https://www.open-mesh.net/Members/adagio/batman-install-howto-stichworte>
- http://open-mesh.net/batman/doc/batmand_howto.pdf
- https://www.opensourcepress.de/fileadmin/osp/pdf/mesh_leseprobe.pdf

6 Tests

(TODO) Vervollständigen

Die Tests, die während der Fachstudie in Nexus-Labor mit der Test-Hardware durchgeführt wurden, sind in diesen Abschnitt detailliert beschrieben.

6.1 Erster Test

6.1.1 Hardware

Zuerst wurde folgende Hardware für den Tests im Nexus-Labor eingerichtet:

- Zwei PCs mit *Wistron CM9 Atheros AR5213A* Wlan Karten
- Ein Laptop mit Intel mini-PCI Wlan Karte

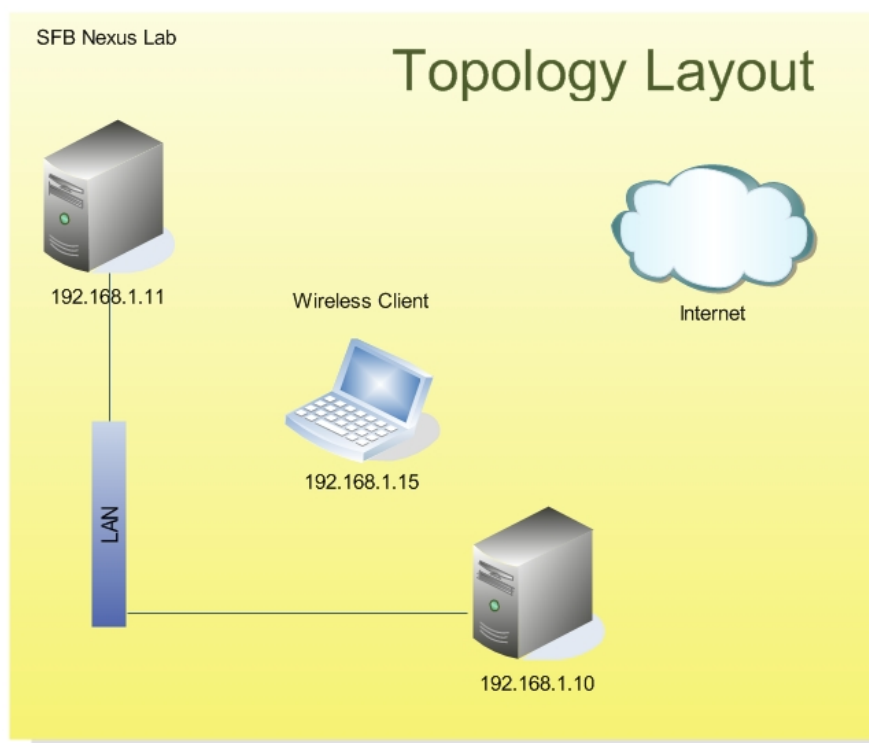


Abbildung 16: Topologie

6.1.2 Software

Dann wurde Betriebssystem Linux (Fedora 2.6.18) auf den Rechnern installiert.

6.1.2.1 Madwifi Danach wurden die Linux-Treiber Madwifi installiert: (*Treiber sind nur für Fedora 2.6.18 kernel!*)

Die einzelnen Schritte dabei waren:

1. Treiber installieren

```
# svn checkout http://svn.madwifi.org/madwifi/trunk madwifi
# cd madwifi
# make
# make install
```

2a. Treiber laden manuell

```
# modprobe ath_pci
```

2b. Treiber laden automatisch

```
# mkdir /etc/modules.autoload.d/
# echo ath_pci >> /etc/modules.autoload.d/kernel-2.6
```

3a. Network-Config manuel

```
# iwlist ath1 frequency
      Channel 01 : 2.412 GHz
      Channel 02 : 2.417 GHz
      Channel 03 : 2.422 GHz
      Channel 04 : 2.427 GHz
      Channel 05 : 2.432 GHz
      Channel 06 : 2.437 GHz
      Channel 07 : 2.442 GHz
      Channel 08 : 2.447 GHz
      Channel 09 : 2.452 GHz
      Channel 10 : 2.457 GHz
      Channel 11 : 2.462 GHz
      Channel 36 : 5.18 GHz
      Channel 40 : 5.2 GHz
```

```
Channel 42 : 5.21 GHz
Channel 44 : 5.22 GHz
Channel 48 : 5.24 GHz
Channel 50 : 5.25 GHz
Channel 52 : 5.26 GHz
Channel 56 : 5.28 GHz
Channel 58 : 5.29 GHz
Channel 60 : 5.3 GHz
Channel 64 : 5.32 GHz
Channel 149 : 5.745 GHz
Channel 152 : 5.76 GHz
Channel 153 : 5.765 GHz
Channel 157 : 5.785 GHz
Channel 160 : 5.8 GHz
Channel 161 : 5.805 GHz
Channel 165 : 5.825 GHz
Current Channel=0
```

```
# ifconfig ath1 inet 192.168.0.1/24
# iwconfig ath1 essid mesh
# iwconfig ath1 mode ad-hoc
# iwconfig ath1 channel 36
# iwconfig ath1 enc n1e2x3u4s5
```

3b. Network-Config automatisch Datei erstellen:

```
# mkdir /etc/sysconfig/network-scripts/ifcfg-ath1
```

und editieren..

```
# Silicon Integrated Systems [SiS] SiS900 PCI Fast Ethernet
DEVICE=ath1
ONBOOT=yes
```

```
BOOTPROTO=static
IPADDR=192.168.2.1x
NETMASK=255.255.255.0
```

```
ESSID=mesh
MODE=ad-hoc
CHANNEL=36 # 5.18 GHz
KEY=s:Onexus0suxen0
# 108-Bit WEP 13 zeichen
```

6.1.2.2 OLSR Daemon Nach der Installation von Madwifi wurde olsrd Daemon installiert und konfiguriert:

1. olsrd installieren

```
# cvs -d:pserver:anonymous@olsrd.cvs.sourceforge.net:/cvsroot/olsrd login
# cvs -z3 -d:pserver:anonymous@olsrd.cvs.sourceforge.net:/cvsroot/olsrd co olsrd-current
# cd olsrd-current
# make
# make install
```

2. Plug-ins für olsrd installieren

```
# cd lib/"plugin-name"
# make
# make install
# chcon -t textrel_shlib_t /usr/lib/olsrd_httpinfo.so.0.1 (!)
```

3. olsrd konfigurieren

- Datei /etc/olsrd.conf erstellen und editieren!!! (TODO) (LINK) Anhang (sieh file)
- TCP Port 8080 für Httpinfo und 8081 für Dot UDP 698 für Eingehende Pakete erlauben.
- Datei /etc/sysconfig/iptables editieren:

```
-A RH-Firewall-1-INPUT -p tcp --dport 8080 -m state --state NEW -j ACCEPT
-A RH-Firewall-1-INPUT -p tcp --dport 8081 -m state --state NEW -j ACCEPT
-A RH-Firewall-1-INPUT -i ath1 -p udp --sport 698 -j ACCEPT
```

4. olsrd starten

```
# olsrd
```

6.1.3 Ergebnisse

(TODO) korrigieren

6.1.3.1 Topologie Mittels *olsrd-Plugin dot*, wurde folgende Topologie-Abbildung erstellt. (TODO) Beschreiben..

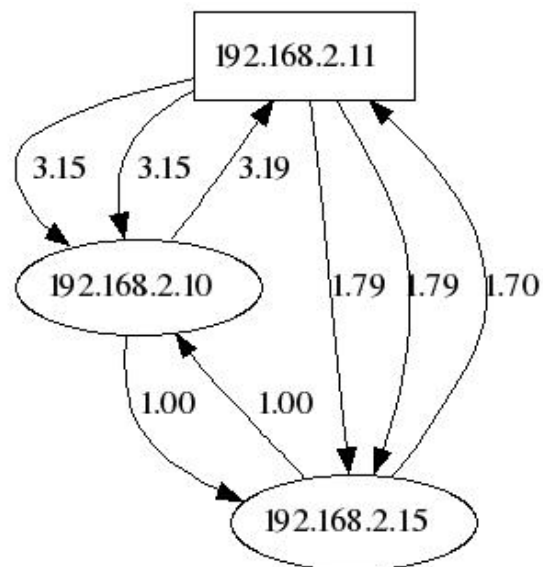


Abbildung 17: Topologie Graph

6.1.3.2 Http Info Mittels *olsrd-Plugin Http Info*, wurde folgende Routing-Übersicht-Abbildung erstellt. (TODO) Beschreiben..



olsr.org OLSR daemon

[Configuration](#)
[Routes](#)
[Links/Topology](#)
[All](#)
[About](#)

Version: olsr.org - 0.5.5pre (built on 2007-10-29 14:42:12 on pcvs63.informatik.uni-stuttgart.de)
 OS: GNU/Linux
 System time: Mon, 19 Nov 2007 17:08:40
 Olsrd uptime: 20 day(s) 22 hours 04 minutes 43 seconds
 HTTP stats(ok/dyn/error/illegal): 748/0/0/0
 Click [here](#) to generate a configuration file for this node.

Variables			
Main address: 192.168.2.11	IP version: 4	Debug level: 0	FIB Metrics: flat
Pollrate: 0.05	TC redundancy: 2	MPR coverage: 7	
Fisheye: Enabled	TOS: 0x0010	RtTable: 0x00fe/254	Willingness: 7
LQ extension: Enabled	LQ level: 2	LQ winsize: 100	

Interfaces		
ath1		
IP: 192.168.2.11	MASK: 255.255.255.0	BCAST: 192.168.2.255
MTU: 1472	WLAN: Yes	STATUS: UP

Olsrd is configured to run even if no interfaces are available

Plugins	
Name	Parameters
olsrd_dot_draw.so.0.3	KEY, VALUE
olsrd_httpinfo.so.0.1	KEY, VALUE

Announced HNA entries				
OLSR Routes in Kernel				
Destination	Gateway	Metric	ETX	Interface
192.168.2.10	192.168.2.15	2	2.449	ath1
192.168.2.15	192.168.2.15	1	1.449	ath1

Links							
Local IP	Remote IP	Hysteresis	LinkQuality	lost	total	NLQ	ETX
192.168.2.11	192.168.2.15		0.00	0.6832	100	1.00	1.47
192.168.2.11	192.168.2.10		0.00	0.5149	100	0.57	3.45

Neighbors					
IP Address	SYM	MPR	MPRS	Willingness	2 Hop Neighbors
192.168.2.10	YES	YES	YES	7	IP ADDRESS (1)
192.168.2.15	YES	YES	YES	3	IP ADDRESS (1)

Topology Entries				
Destination IP	Last Hop IP	LQ	ILQ	ETX
192.168.2.11	192.168.2.10		0.51	0.57
192.168.2.15	192.168.2.10		1.00	0.99
192.168.2.10	192.168.2.11		0.51	0.57
192.168.2.15	192.168.2.11		0.69	1.00
192.168.2.10	192.168.2.15		1.00	1.00
192.168.2.11	192.168.2.15		0.69	1.00

Abbildung 18: Http Info

6.2 Zweiter Test

(TODO) .. 2 laptops, windows, ..

7 Fazit

(TODO) TEXT

7.1 Übersicht

Dieser Abschnitt stellt ein Übersicht über aktuelle verfügbare Hardwareplattformen und Systemsoftware für WMN dar.

	IEEE 802.11a	Ad-Hoc Modus	Treiber (Linux/Windows)	Open-Source Firmware	LAN-Anschluss	Sicherheit	Installation	Konfiguration	Mini-PCI Slot	IEEE 802.11n
Linksys WMP55AG	++	+	++	-	++	++	+	+	-	-
Netgear WAG311	++	+	++	-	++	++	+	+	-	-
D-Link DWL-A520	+	+	++	-	++	+	+	+	-	-
Gigabyte GN-WPEAG	++	+	++	-	++	+++	+	+	-	-
Intel PRO/Wireless 5000	+	+	+	-	++	+	+	+	-	-
D-Link DWL-AG530	++	+	++	-	++	+++	+	+	-	-
D-Link DWL-G550	++	+	++	-	++	+++	+	+	-	-
Wistron CM9 Atheros AR5213A	++	+	++	-	++	+++	+	+	-	-
Intel PRO/Wireless 3945	++	+	++	-	++	+++	+	+	-	-
Intel PRO/Wireless 2915	++	+	++	-	++	+++	+	+	-	-
Intel Wireless WiFi Link 4965AGN	+++	+	++	-	++	+++	+	+	-	+
Linksys WRT54G v1.0	++	+	-	++	+	+++	-	+	+	-
Linksys WRT55AG	++	+	-	+	+	+	-	+	++	-
Asus WL500G/GP	++	+	-	+++	+	+++	-	+	+	-
Netgear HR314	+	-	-	-	+	+	-	+	-	-

Abbildung 19: Übersicht und Bewertung von Hardware

	Betriebssysteme	Installation	Konfiguration	Visualisierung
OLSRD	+++	+	+	++
B.A.T.M.A.N.	+	+	+	-

Abbildung 20: Übersicht und Bewertung von Software

7.2 Aufwandschätzung und Bewertung

(TODO) Bewertung der analysierten Systeme hinsichtlich der ermittelten Anforderungen..

7.3 Empfehlung für eine geeignete WMN

(TODO) Ausarbeitung einer Empfehlung für eine geeignete WMN hinsichtlich Hardware-plattform und Systemsoftware

olsrd config file: