WLan und Sicherheit - geht das überhaupt?

Johannes Bauer

3. Erlanger Linuxtage

16. Januar 2005

Wireless LAN

- Netzwerkstandard IEEE 802.11b/g im 2.4GHz-Band
- Drahtlos surfen, mailen, chatten
- WLan-Access-Points bzw. WLan-DSL-Router ermöglichen den bequemen Zugang zum Netz
- Durch billigste Hardware ist WLan für jeden erschwinglich

„Nur weil ich paranoid bin heisst das noch lange nicht, dass sie nicht hinter mir her sind.“

Unverschlüsselte Kommunikation

- Viele Protokolle (POP3, SMTP, HTTP) übermitteln Passwörter im **Klartext**
- Absolute Sicherheitskatastrophe
- Man „schreit“ sein Passwort förmlich in den Äther

Verschlüsselung mit WEP

So sicher wie drahtgebunden?

- Sämtliche Daten werden drahtlos übertragen, mithören ist problemlos möglich
- Verschlüsselung also notwendig
  $\longrightarrow$ WEP (Wired Equivalent Privacy)
- Benutzung einer Kombination aus RC4 zusammen mit Frequenzspreizverfahren
- „So sicher wie drahtgebunden“?

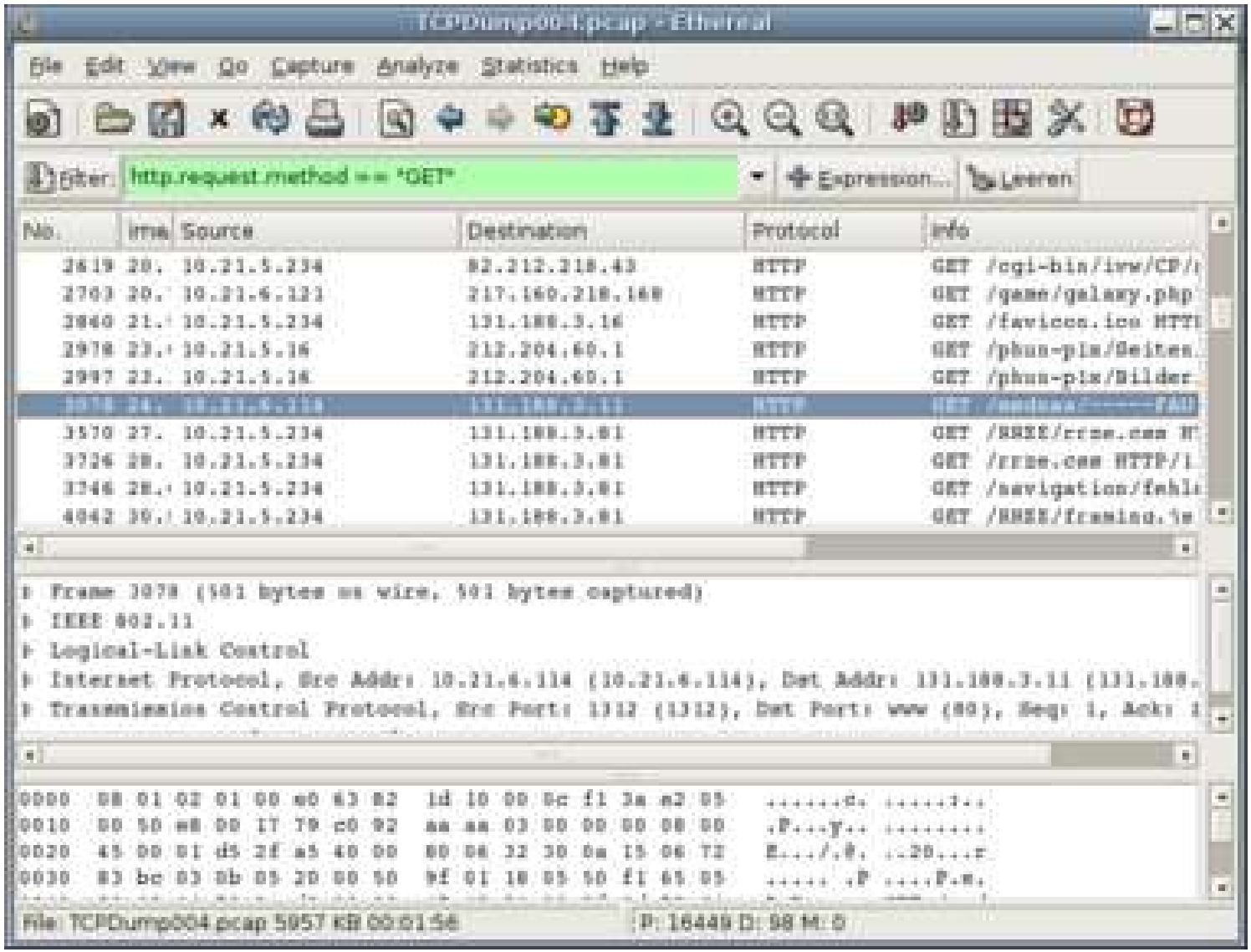
Verschlüsselung mit WEP

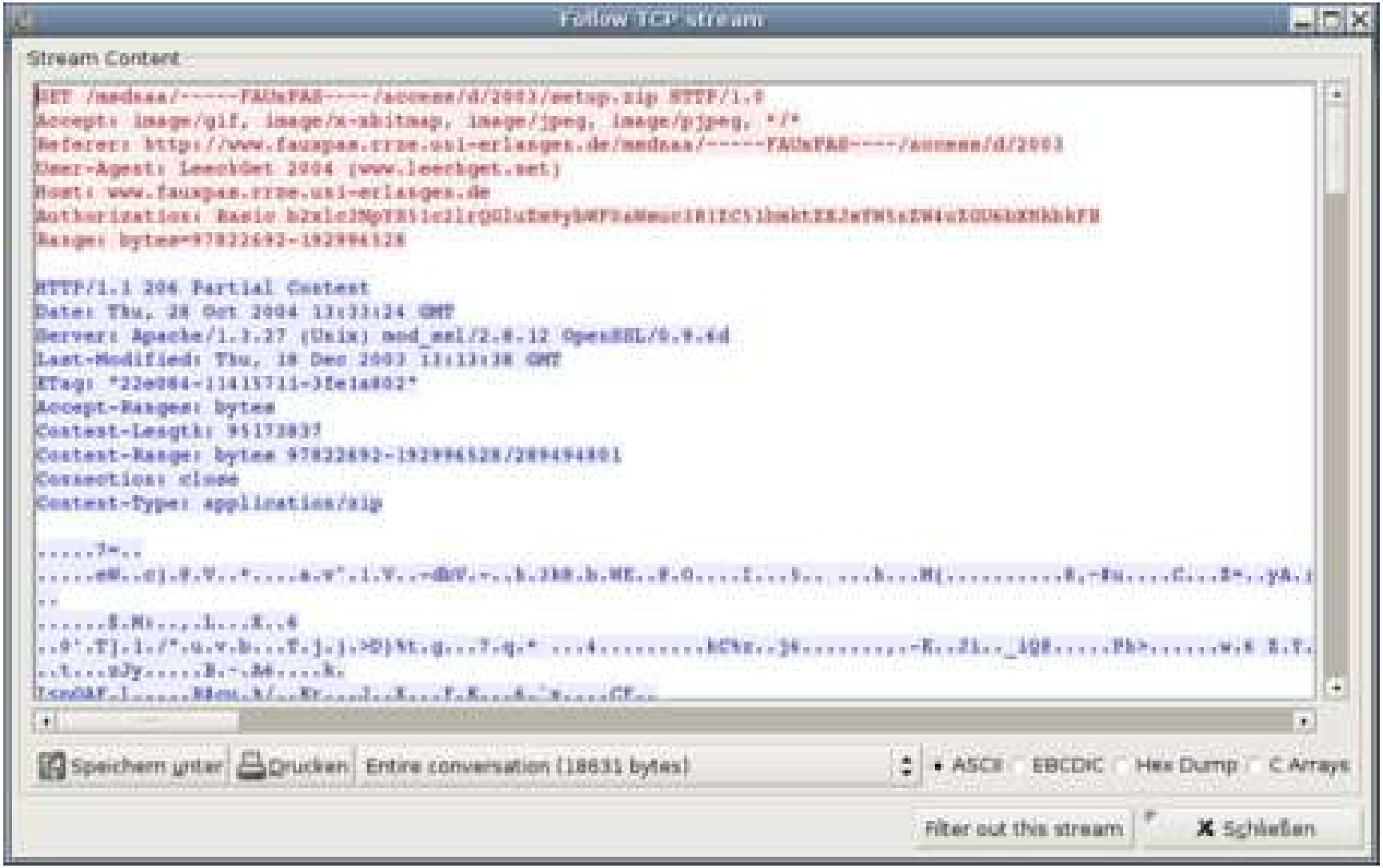
So sicher wie drahtgebunden? Leider nicht.

- Problem an WEP: durch Zufall werden sogenannte „schwache“ Initialisierungsverktoren (IVs) zur Verschlüsselung benutzt
- Die Verschlüsselungstabelle der „schwachen“ Pakete enthält eine starke Korrelation mit einigen Schlüsselbytes
- Schwäche von RC4 schon seit 1995 bekannt
- Fatale Scheinsicherheit, WEP gebrochen

Ethereal

- Capturen von Netzwerkverkehr
- Detaillierte Analyse des Verkehrs möglich
- Kann bequem für die „Vorverarbeitung“ des Netzverkehrs benutzt werden, um die Daten dann z.B. mit Ettercap auszuwerten


TCPDump004.pcap - Ethereal
File Edit View Go Capture Analyze Statistics Help
Filter: http.request.method == "GET"
Expression... Leeren
No. Time Source Destination Protocol Info
2619 20. 10.21.5.234 82.212.218.43 HTTP GET /cgi-bin/irw/CP/
2703 20. 10.21.6.123 217.160.218.168 HTTP GET /game/galaxy.php
2840 21. 10.21.5.234 131.188.3.16 HTTP GET /favicon.ico HTTP
2978 23. 10.21.5.16 212.204.60.1 HTTP GET /phus-pls/Seiten
2987 23. 10.21.5.16 212.204.60.1 HTTP GET /phus-pls/Bilder
3570 27. 10.21.5.234 131.188.3.81 HTTP GET /RRZE/rrze.css H
3726 28. 10.21.5.234 131.188.3.81 HTTP GET /rrze.css HTTP/1
3746 28. 10.21.5.234 131.188.3.81 HTTP GET /navigation/fehl
4042 30. 10.21.5.234 131.188.3.81 HTTP GET /RRZE/framing.%
Frame 3078 (501 bytes on wire, 501 bytes captured)
IEEE 802.11
Logical-Link Control
Internet Protocol, Src Addr: 10.21.6.114 (10.21.6.114), Dst Addr: 131.188.3.11 (131.188.
Transmission Control Protocol, Src Port: 1312 (1312), Dst Port: www (80), Seq: 1, Ack: 1
0000 08 01 02 01 00 e0 63 82 1d 10 00 0c f1 3a e2 05
0010 00 50 e8 00 17 79 c0 92 aa aa 03 00 00 00 08 00
0020 45 00 01 d5 2f a5 40 00 80 06 32 30 0a 15 06 72
0030 83 bc 05 0b 00 50 bf 01 10 05 50 f1 65 05
File: TCPDump004.pcap 5957 KB 00:01:56
P: 16449 D: 98 M: 0


Follow TCP stream
Stream Content
GET /mednaa/-----FAUnFAU----/accmnn/d/2003/setup.zip HTTP/1.0
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Referer: http://www.faunpan.rrze.uni-erlangen.de/mednaa/-----FAUnFAU----/accmnn/d/2003
User-Agent: LeechGet 2004 (www.leechget.net)
Host: www.faunpan.rrze.uni-erlangen.de
Authorization: Basic b2xlc3Mgt3B5leZlrQGluZm4yb3WFraMmucIR1ZC5lbmktE8JaFW5sZW4uZGUkbXNkbkFB
Range: bytes=97822692-192994528
HTTP/1.1 206 Partial Content
Date: Thu, 28 Oct 2004 13:33:24 GMT
Server: Apache/1.3.27 (Unix) mod_ssl/2.8.12 OpenSSL/0.9.6d
Last-Modified: Thu, 18 Dec 2003 13:13:38 GMT
ETag: "22e084-11415711-3fe1a892"
Accept-Ranges: bytes
Content-Length: 95173833
Content-Range: bytes 97822692-192994528/289494801
Connection: close
Content-Type: application/zip
Speichern unter
Drucken
Entire conversation (18631 bytes)
ASCII
EBCDIC
Hex Dump
C Arrays
Filter out this stream
Schließen

Kismet

- Captured Netzwerkverkehr, filtert schwache IVs zur späteren Benutzung sofort aus
- Findet Netze (ESSID Broadcast) und markiert diese via GPS
- Zuhause kann das WEP in Ruhe geknackt werden, dann kann man das Netz „wiederfinden“
- Grafische Auswertung integriert

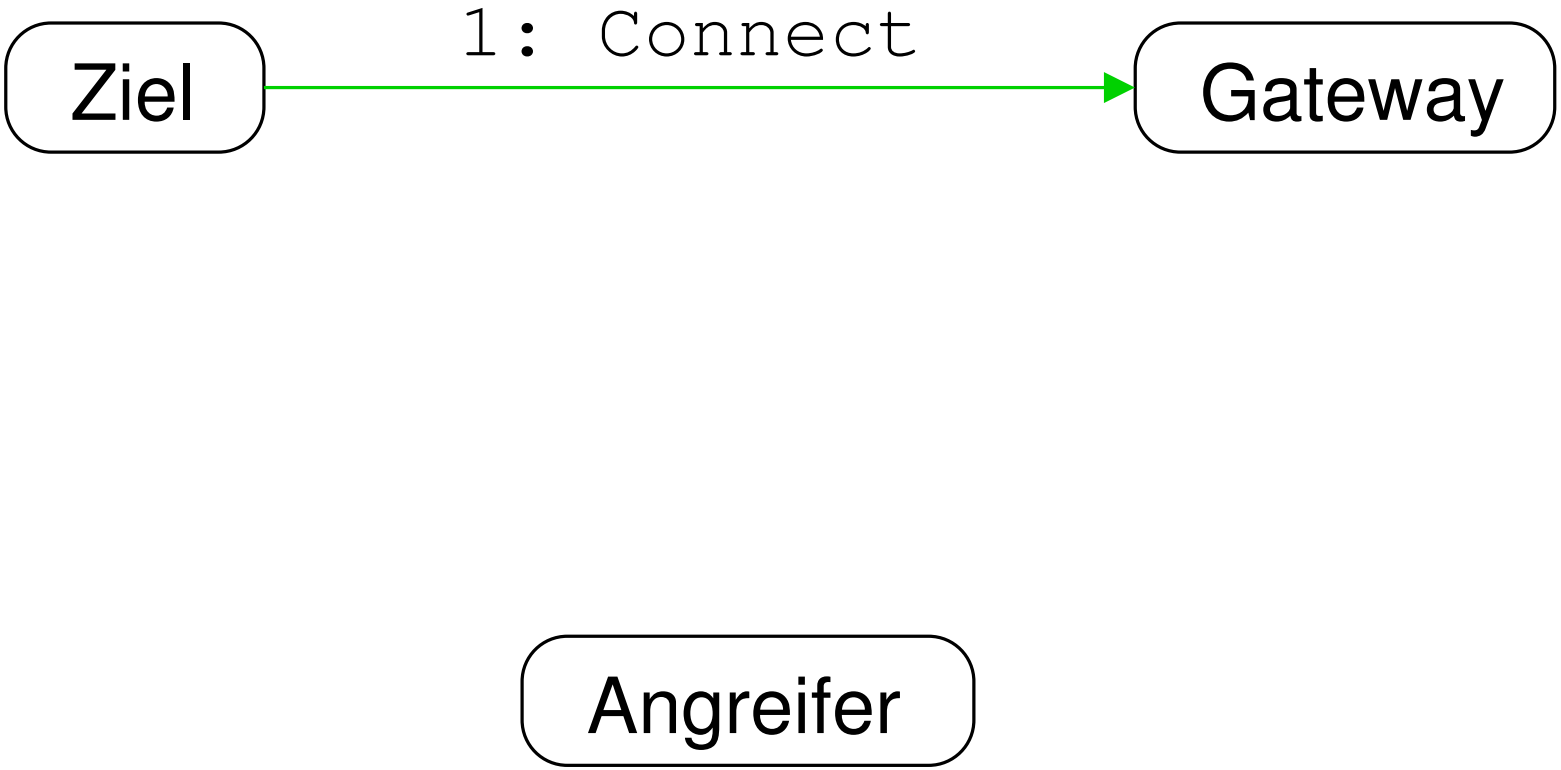

'ConnectionPoint'
'NETGEAR'
'WLAN'
'eurospot'
'Wireless'

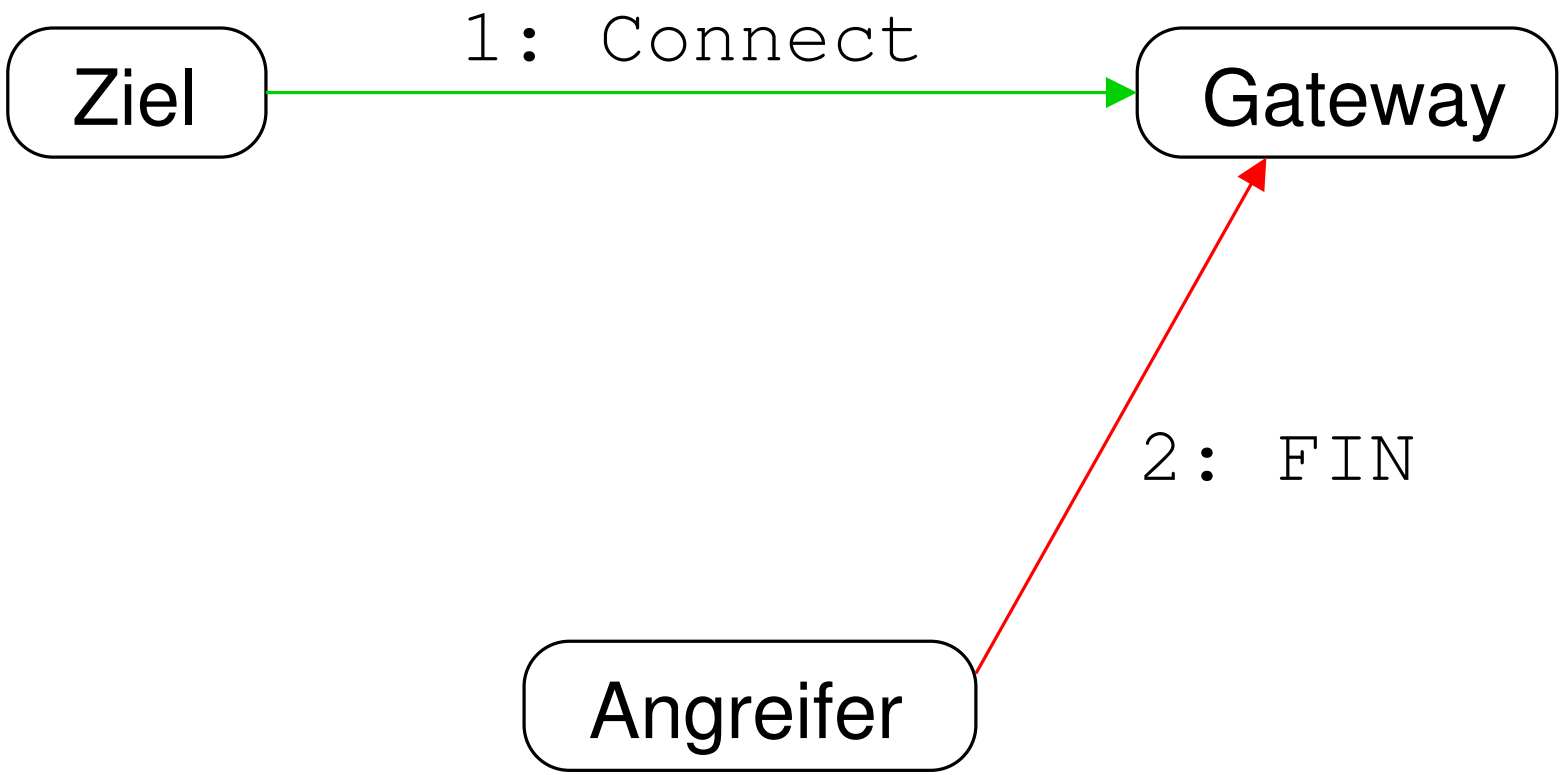
Ettercap

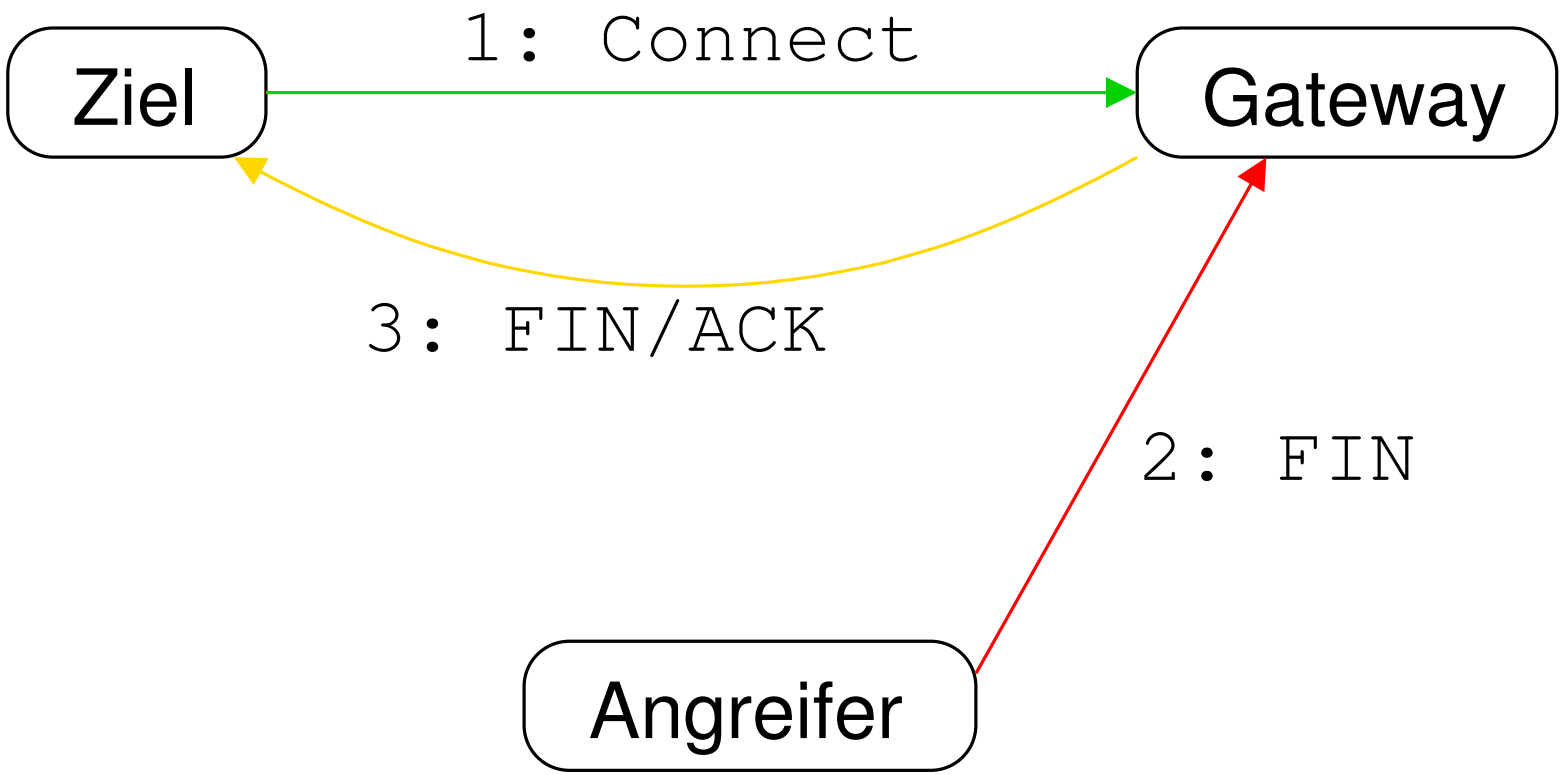
- Dediziertes Programm zum Ausschlachten des mitgelauschten Verkehrs
- Unterstützt zahllose Protokolle (FTP, HTTP, POP, IMAP, SMTP, SNMP, NNTP, Telnet...)
- Völlig automatisches Sniffen und Dissektion des Verkehrs möglich

DoS mit Ettercap

- Ettercap unterstützt „TCP Connection killing“
- Sobald das Ziel eine Verbindung aufbaut, sendet Ettercap ein gefälschtes Paket mit der Mac/IP-Adresse des Ziels, in dem das FIN-Flag gesetzt ist
- Der AP denkt, das Ziel will die Verbindung beenden und sendet ein FIN/ACK
  $\longrightarrow$ Die Verbindung ist abgebrochen (Socket geschlossen)


Ziel
1: Connect
Gateway
Angreifer


Ziel
1: Connect
Gateway
2: FIN
Angreifer


Ziel
1: Connect
Gateway
3: FIN/ACK
2: FIN
Angreifer

Wepattack und John

Wörterbuchattacke

- *John the Ripper* erzeugt aus verschiedensten Passwortlisten „wahrscheinliche“ Passwörter
- *Wepattack* liest diese Passwörter und probiert, ob sie zu einem der Netze passen
- Meist innerhalb weniger Minuten die ersten Netze aufgebrochen
- Schuld des Benutzers: Wahl eines schwachen Passworts

MAC-Spoofing

Theorie...

- MAC-Addressen sollten fest „eingebrannt“ sein
- Realisierung jedoch vollständig in Software
- Fälschen durch einen einfachen ifconfig-Aufruf möglich
- MAC-Sperren bringen wiederum trügerische Scheinsicherheit

MAC-Spoofing

...und Praxis


```
joe [~]: ifconfig wlan0 down
joe [~]: ifconfig wlan0 hw ether ba:db:ad:c0:ff:ee
joe [~]: ifconfig wlan0 up 123.123.123.123
joe [~]: ifconfig wlan0
wlan0
   Protokoll:Ethernet
   Hardware Adresse BA:DB:AD:CO:FF:EE
   inet Adresse:123.123.123.123
   Bcast:123.255.255.255
   Maske:255.0.0.0
```

DoS-Attacke

Ein Netz komplett stilllegen mit nur 4 Befehlen

- Ziel durch belauschen des Verkehrs herausfinden (IP/Mac-Adresse)
- Bevorzugtes Ziel: Internet-Gateway
- Eigenem Netzwerkinterfache eben diese IP/Mac-Adresse zuweisen
- Interface „hoch“ bringen, Defaultroute setzen
  $\longrightarrow$ Das Netz ist „tot“

WLan und Windows-Rechner

- Für Windows-Rechner sind Wireless-LAN-Karten ganz „normale“ Netzwerkinterfaces
- Sämtliche Dateifreigaben gelten also auch für WLan
- Selbst wenn keine Freigaben erstellt worden sind, kann durch Erraten zu einfacher „Administrator“-Kennwörter an den gesamten Festplatteninhalt gelangt werden

Passwörter...

120783	ak47LSG	Cro4!Mex	ficken
14bada384	anke4534	default	figggn
15mausendorf	augenlos	dJ570?ui	figgn
18058977	baby2809	Drolan	firemail
2ganglion3	bacardi52	essal482	frank
4135327429	brochier	evui3D	ghost1312
4371ssbs	Bu313Fly	f1ck4l4t0r	glibber
4q538f	by5Dga	faui2k3	h4x0r
698695	christa0407	faui2k4	hurz
9se6Rvi2X5	connect	FAUxPAS	jhvip

Passwörter...

kemal	monte	robert	start
kemal1	mützli	rotstern	TanteJu
kerstin	Neuche	sAi257	thml1
konkurs	Nighthawk745	sa!N3draX5	tschau24
lexi11081978	oneisha	Schatzi22	Uwlee
loren	P2277	SHADOW	vidim
luBu99	Pappenheim	skript	vt091002
m20n84	papst	Skript	whisky
madcat	pkphil79	sonne1	wurstsuppe
Mahatma	renseb	sos2k3	zhu8hw

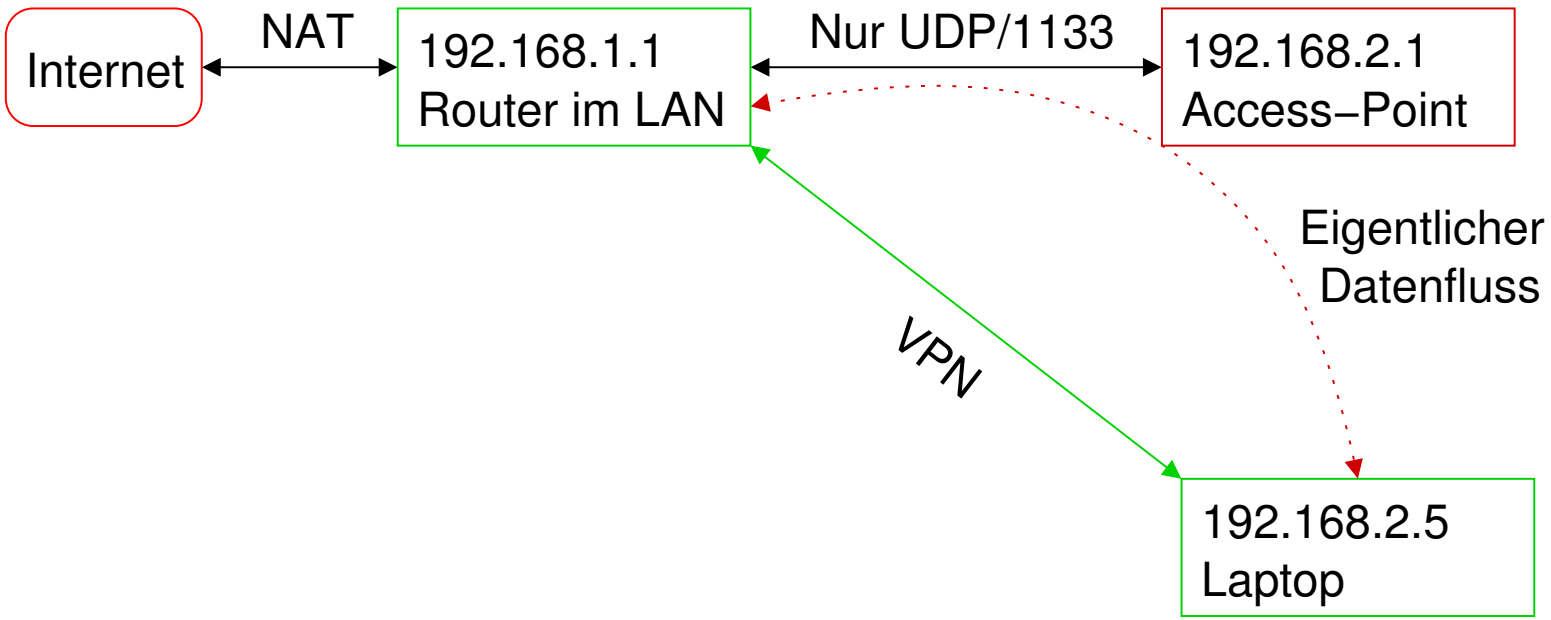
OpenVPN

- Virtual Private Network, d.h. ein virtuelles Netzwerkinterface
- Sämtlicher Verkehr wird durch das „unsicher“ Interface verschlüsselt gesandt
- Point-to-Point-Verbindung z.B. zu einem Router, der im LAN steht

OpenVPN

Generelle Funktionsweise

- Durch das unsichere (drahtlose) Netzwerk wird getunnelt
- Es entsteht ein neues Netzwerkinterface (tun0)
- Alles, was über das neue PtP-Interface übertragen wird, wird verschlüsselt gesendet


Internet
NAT
192.168.1.1
Router im LAN
Nur UDP/1133
192.168.2.1
Access−Point
Eigentlicher
Datenfluss
VPN
192.168.2.5
Laptop

OpenVPN

Topologie

- Lokales Netzwerk (LAN): 192.168.1.0/24
  - Gateway: 192.168.1.1
  - Ans Internet angebunden
- Funknetz (WLan): 192.168.2.0/24
  - Gateway (AP): 192.168.2.1
  - An das LAN angebunden

OpenVPN

Punkt-zu-Punkt Verbindung

- WLan-Access-Point erlaubt nur Routing zu 192.168.1.1, UDP Port 1194
- Auf 192.168.1.1 läuft OpenVPN im Servermodus
- Auf dem WLan-Client (192.168.2.x) läuft OpenVPN im Clientmodus

OpenVPN

Punkt-zu-Punkt Verbindung

- WLan-Client meldet sich zunächst am Funknetz an (z.B. als 192.168.2.5)
- Dann einzig mögliche Verbindung: OpenVPN zu 192.168.1.1
- Daten werden also doppelt verschlüsselt: WEP und OpenVPN
- Auf 192.168.1.1 wird mit ankommenden Paketen dann NAT (Masquerading/Routing) betrieben

OpenVPN

Probleme

- Erste Einrichtung etwas umständlich
- DoS durch IP/Mac-Spoofing des AP immernoch möglich
- Leicht verringerte Bandbreite
- Kleiner Rechner im lokalen Netz notwendig

OpenVPN

Vorteile

- Echte Authentifizierung (ersetzt die MAC-Sperre)
- Gute Verschlüsselung (ersetzt WEP)
- Kein Connection-Killing möglich (UDP ist „stateless“, d.h. keine FIN-Pakete)

Zusammenfassung

- AP mit Passwortschutz versehen
- Keinen AP, der ein Generalpasswort hat, akzeptieren
- Die Firmware des AP updaten, wenn sicherheitsrelevante Fehler darin behoben werden

Zusammenfassung

- WPA verwenden, wenn möglich
- Ansonsten: WEP mit höchster Schlüssellänge verwenden
- Schlüssel hexadezimal eingeben, häufig wechseln
- Kryptische ESSID wählen, nicht senden lassen
- SSL verwenden, wenn möglich
- **Virtual Private Network (z.B. OpenVPN) verwenden!**

Abschluß

- Danke für die Aufmerksamkeit!