Lösungsansätze für das Wave-LAN Security Desaster

Ein Drama

Ruediger Weis, Jens Ohlig

Chaos Computer Club

Hackerethik I

- Der Zugang zu Computern und allem, was einem zeigen kann, wie diese Welt funktioniert, sollte unbegrenzt und vollständig sein.
- Alle Informationen müssen frei sein.
- Misstraue Autoritäten - fördere Dezentralisierung.

Hackerethik II

- Beurteile einen Hacker nach dem, was er tut und nicht nach üblichen Kriterien wie Aussehen, Alter, Rasse, Geschlecht oder gesellschaftlicher Stellung.
- Man kann mit einem Computer Kunst und Schönheit schaffen.
- Computer können dein Leben zum Besseren verändern.

Hackerethik III

- Mülle nicht in den Daten anderer Leute.
- Öffentliche Daten nützen, private Daten schützen.

WLAN Realitätsabgleich

Wavelan ist Zukunftstechnologie.

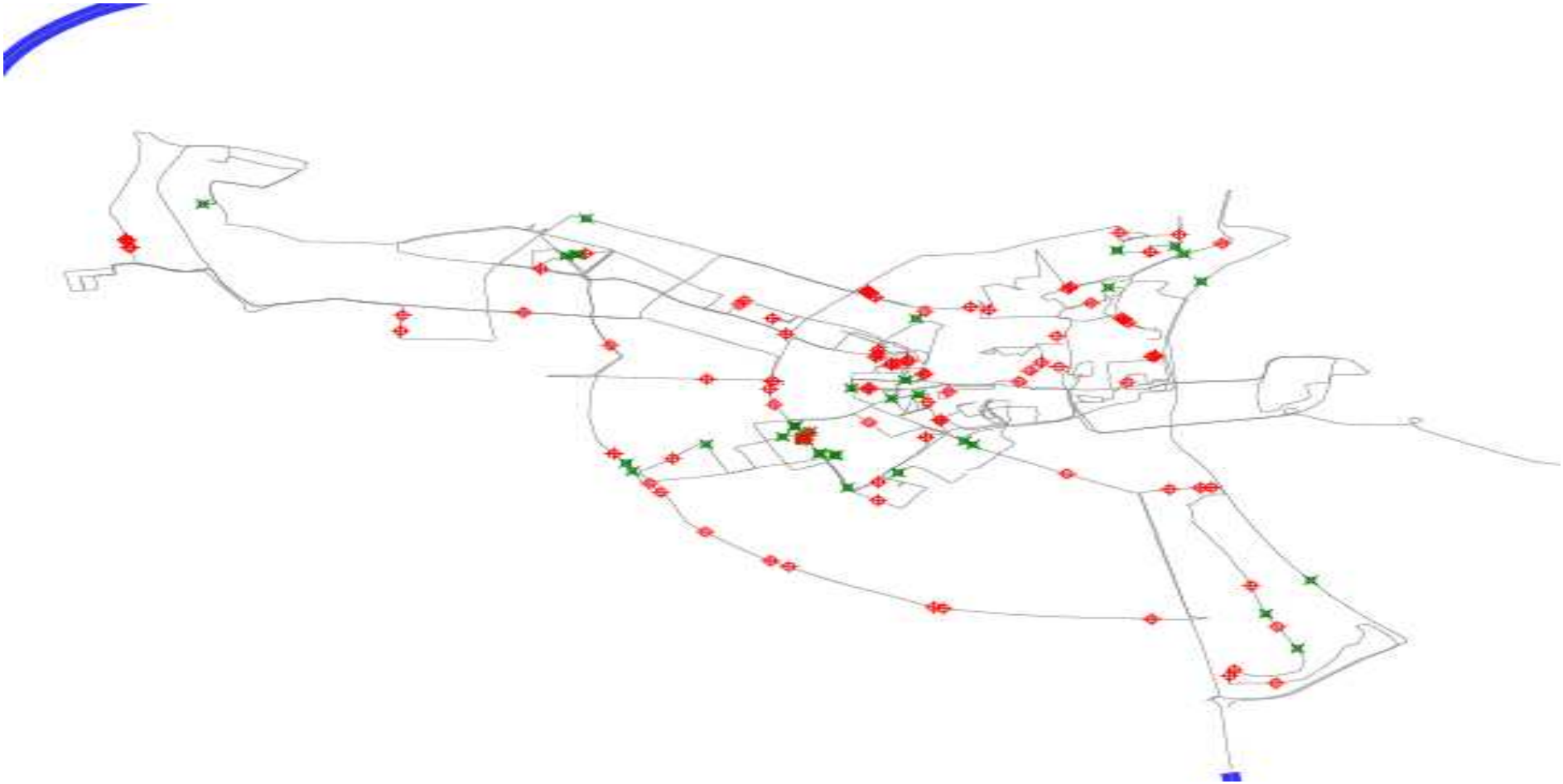
- "Über 80% der Netze klicken nicht mal auf Sicherheitsemulation." - CCC 2001
- Tiroler Zeitung, Oktober 2002
  - 40-50 Netze in Rum und Innsbruck.
  - “Etwa drei Viertel” offen

Das *CEBIT-Problem*

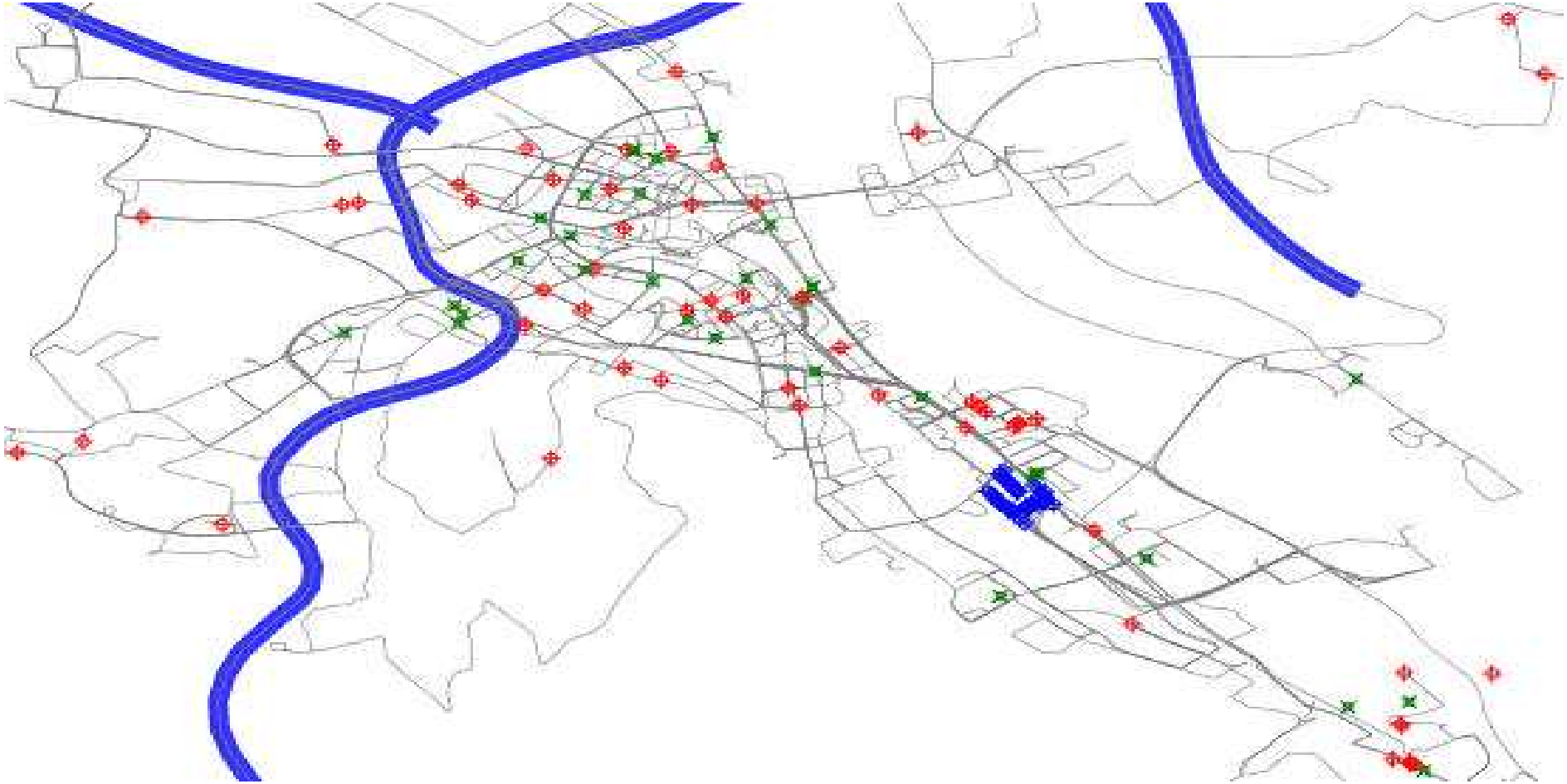
Köln/Bonn

- 283 Netze in Köln/Bonn
- 150 Netze in Bonn
- 23% WEP
- 21% hidden

Karte Köln



Karte Bonn



Berlin

- Krankenhäuser
- Firmenzentralen

Prolog 1: Grundschutz

- Netz verstecken
- MAC Adressen festlegen
- WEP einschalten

Ansonsten kann jeder *"legal"* mithören.

Wireless Encryption Placebo

WEP

- 1 Shared Secret
- Verschlüsselung: RC4
- Initialvektor: 24 bit

Bausteine: Problematisch bis katastrophal

Zur Erinnerung:

Sicherheit der Bausteine ist NOTWENDIG aber NICHT hinreichend

für sichere Protokolle.

Prolog 2: 40 bit Schlüssel

Angreifer:

- Netzwerk: Passives Abören
- IQ: Gering
- Rechenpower: Mittel

"Sollte jeder Informatiksudent als Übungsaufgaben bewältigen können."

1.Akt: Protokolldesign ist schwierig ...

und wenn man Stromchiffrierer verwendet wird es *richtig* schwierig.

Stromchiffrierer Verschlüsselung

$$\text{Ciphertext} = \text{Klartext} \oplus \text{Schlüsselstrom(Key)}$$

Stromchiffrierer Entschlüsselung

$$\text{Klartext} = \text{Ciphertext} \oplus \text{Schlüsselstrom(Key)}$$

Trivialitäten

Zwei Dinge sind recht offensichtlich.

- Erstens hängt der Schlüsselstrom nur vom Schlüssel und in keiner Weise vom Klartext ab.
- Wenn man 2 Nachrichten mit dem selben Schlüssel verschlüsselt hat man ein Problem.

Ganz schwere Mathematik

Kennt man nun den Klartext$_1$, so kann man auch den Klartext$_2$ lesen denn:

$$
\begin{aligned}
& & \text{Ciphertext}_2 \oplus & \\
\text{Ciphertext}_1 \oplus \text{Klartext}_1 & = & (\text{Klartext}_2 \oplus \text{Schlüsselstrom(Key)}) & \\
& & \oplus(\text{Klartext}_1 \oplus \text{Schlüsselstrom(Key)}) & \\
& & \oplus\text{Klartext}_1 & \\
& = & \text{Klartext}_2 &
\end{aligned}
$$

und noch mehr Mathematik

und es kommt sogar noch übler, denn es gilt

$$\begin{aligned} \text{Ciphertext}_1 &= \text{Klartext}_1 \oplus \text{Schlüsselstrom(Key)} \\ \text{Ciphertext}_2 &= \text{Klartext}_2 \oplus \text{Schlüsselstrom(Key)} \\ \hline \text{Ciphertext}_1 \oplus \text{Ciphertext}_2 &= \text{Klartext}_1 \oplus \text{Klartext}_2 \end{aligned}$$

Schlüsselwechsel ”recommended”

Trivialer Weise ist also eine Wiederholung des Schlüssels für einen Stomchiffrierer unbeding zu vemeiden.

- Der Standard schreibt dies *nicht* vor.

Dies bedeutet, dass auch eine Implementierung, welche immer den selben Schlüssel verwendet, vollständig standardkonform ist.

Baustein CRC32

- 32 Bit Ausgabe
- Linear
- Keine kryptographische Hashfunktion.

Linarität

- 24 bit IV
- 32 bit Kontrollsumme
- Linearität von RC4 und CRC32 ermöglicht nicht entdeckbare Manipulationen.

Die üblichen Verdächtigen

- Nikita Borisov (UC Berkeley),
- Dr. Ian Goldberg (Zeroknowledge),
- Prof. Dr. David Wagner (UC Berkeley)

Akt2: Kryptographie ist schwierig...

insbesondere Stromchiffrierer-Design.

Eine “Fehleinschätzung”:

”Übrigens, wenn das Protokoll insgesamt nicht so katastrophal schwach wäre, würde es sich vielleicht auch lohnen nachzusehen, ob die ersten Bits des Ausgabestromes von RC4 in verworfen werden. Die haben nämlich einige unschöne Eigenschaften.”

R.Weis, Datenschleuder Sommer 2001

Es lohnte sich...

- Roos, A Class of Weak Keys in the RC4 Stream Cipher, attachment to e-mail to cypherpunks@toad.com, September 22, 1995
- Weis, Security Problems of ARCfour, CCC Datenschleuder, Sommer 200

  http://cryptolabs.org/arcfour/WeisDatenschleuderSummer2000arcfour.txt
- Fuhler, Martin, Shamir, Weakness in the Keyshedule of RC4, SAC 2001.

Aktuelle Sicherheitslage

- Airsnort und Co
- Linux PDAs

Akt3: Hilfe naht?

Alles neu?! IEEE802.11i

Wi-Fi Protected Access (WPA):
Ankündigung Ende Oktober 2002

- Längere Initialization-Vector
- Re-Keying (TKIP)
- Message-Integrity-Check (Michael)

Heftpflaster

- Falsches Design.
- Falsche Algorithmen.
- Key Management nötig.
- Besser als nichts.

Migration zu AES später... IEEE802.11i

Zusammenfassung

Im Moment ist es möglich durch ein paar stündiges nicht bemerkbares Abhören mit preisgünstiger Standardhardware und frei erhältlicher GPL Software den geheimen Schlüssel zu erhalten und damit WEP vollständig zu komprimitieren!

Empfehlung

- Wavelan in DMZ.
- Nutze und ignoriere WEP.
- Praktische Lösung: IPsec.
- Infos: `cryptolabs.org/wep/`

”Vielleicht sollte man jemanden fragen, der sich damit auskennt...”

Epilog: All Your Basestations...

Are Belong To US

Nov 2002: Access Point Exploits

Verbreitete “Lösungen”

- Vermeidung SSID-Broadcast - Hilft nix!
- WEP-Verschlüsselung - Hilft nix!
- Firewall - Wavelan innen?
- MAC-Adress-Authentifizierung - Hilft nix!
- VPN - IPsec ist eine gute Idee.
- Proprietäre Lösungen - Gar keine gute Idee!!!

Kontakt

- Dr. Ruediger Weis, cryptolabs Amsterdam
  - ruedi@cryptolabs.org
  - www.cryptolabs.org
- Jens Ohlig, Sprecher Chaos Computer Club
  - jens@ccc.de
  - www.ccc.de

Produced with GPL software. Typesettig: LaTeX.