

/Root@d[🔒] 2013



Why
iOS (Android & others)
Fail
inexplicably?

 **Taddong**

www.taddong.com

[@taddong](#)

Raúl Siles

raul@taddong.com

March 9, 2013



Wi-Fi Challenges Today?





Outline



Pen-test:

- Wi-Fi ((mobile) client) security
- Wi-Fi mobile clients behavior & The PNL
 - iOS recent Wi-Fi updates
- Wi-Fi network impersonation
 - Attacking Wi-Fi (personal) clients
 - Attacking Wi-Fi enterprise clients
- Post-MitM Wi-Fi exploitation
- Conclusions & References

Mapping

Exploitation

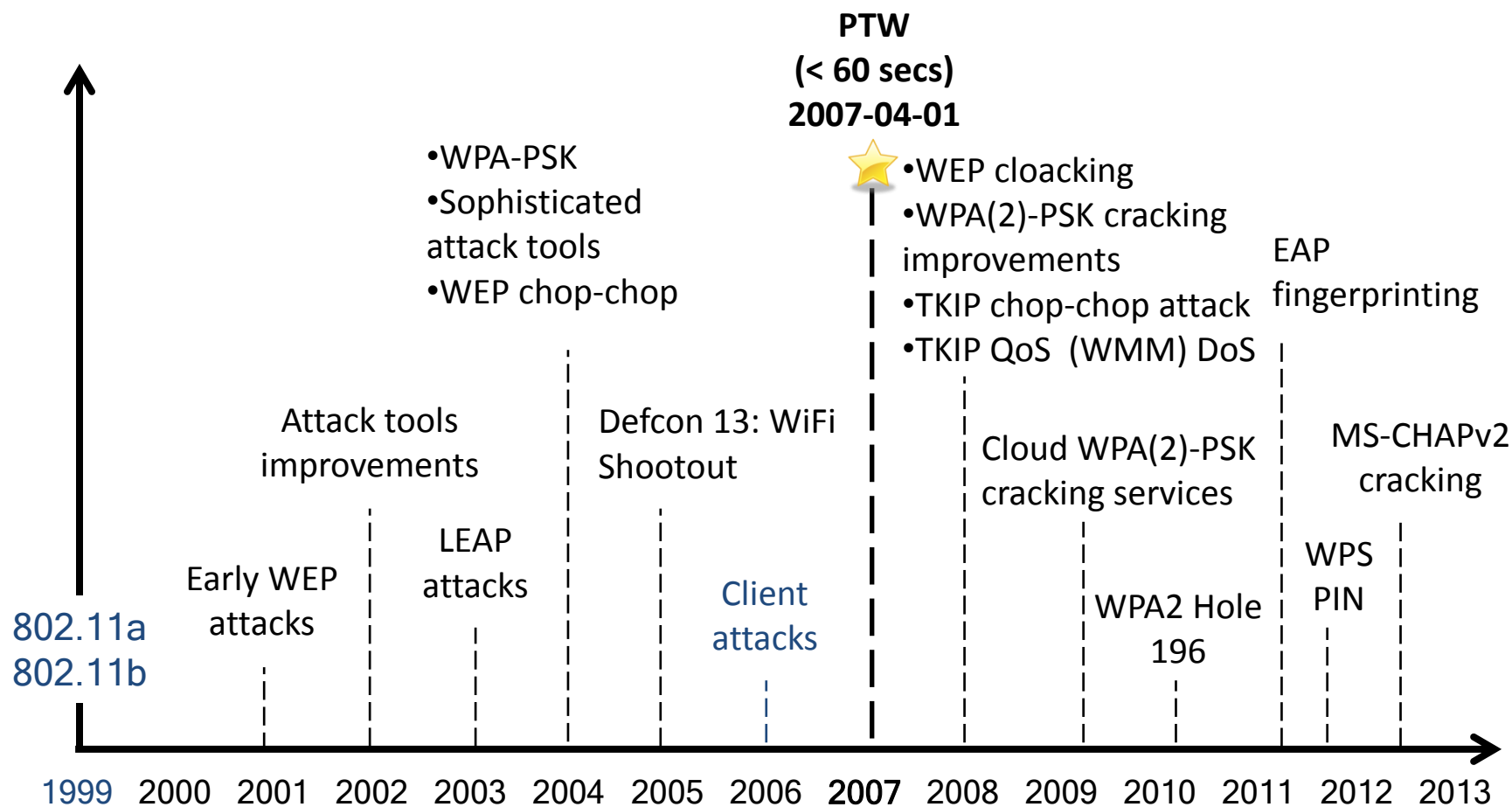
Post...

Mainly iOS & Android (+90% market share Feb'2013), but others...



Wi-Fi Security

State-of-the-Art During the Last Decade (or More...)

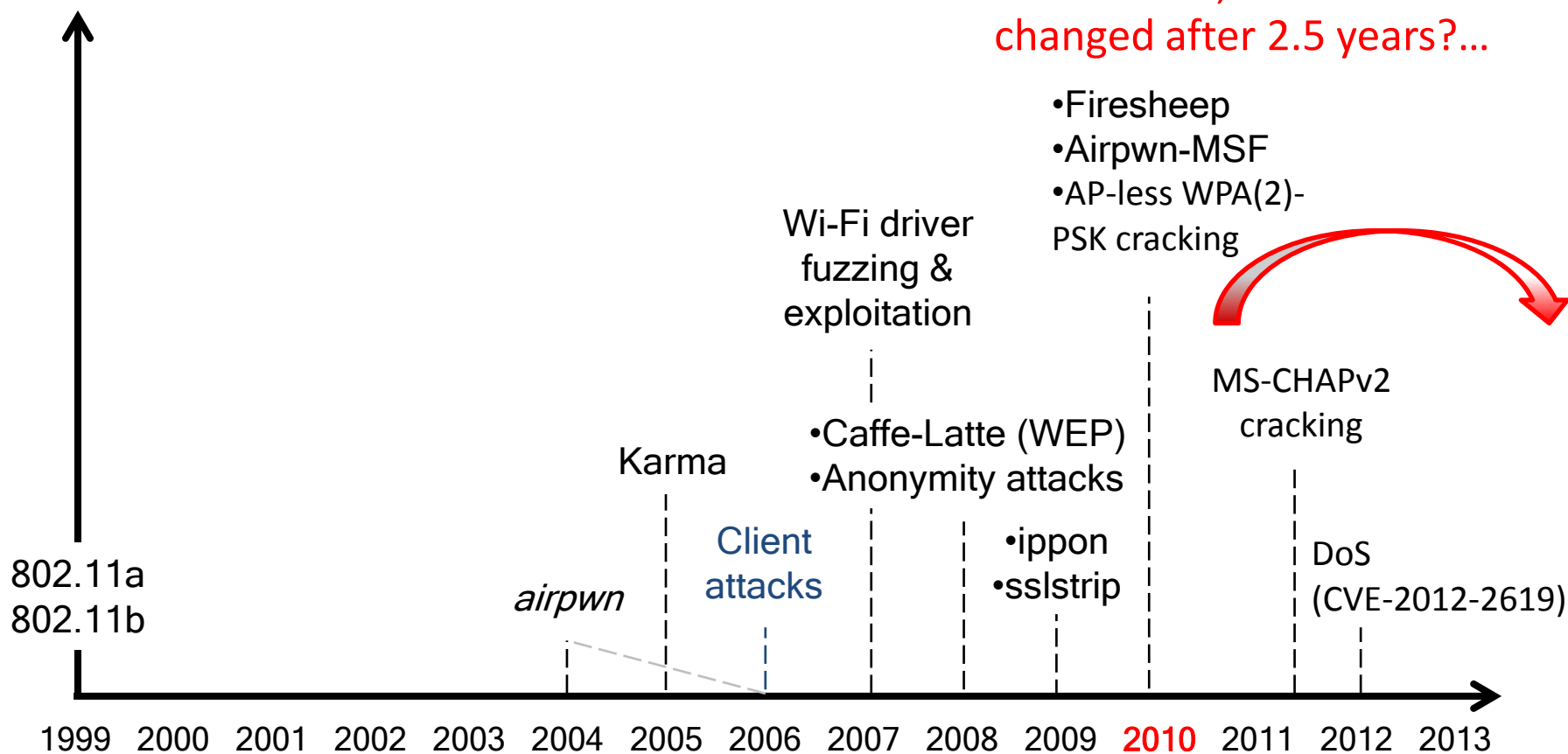


Target: Wi-Fi Infrastructure vs. Wi-Fi Clients



Wi-Fi Clients Security

My last Wi-Fi presentation was in late 2010, so... what has changed after 2.5 years?...



“Wi-Fi (In)Security” (Raul Siles, GOVCERT & CCN-CERT, Nov 2010)



Wi-Fi Mobile Clients Behavior



How Wi-Fi Clients Work?

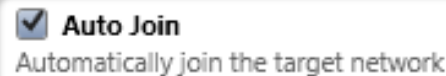
- Users connect to Wi-Fi networks by...
 1. Selecting them from the list of currently available networks in the area of coverage
 2. Adding them manually to the Wi-Fi client
- Security settings are mandatory (if any)
 - Open, WEP, WPA(2)-Personal & WPA(2)-Enterprise
- Networks are remembered and stored for future connections: list of known networks

The Preferred Network List (PNL)



Mobile Clients Standard Behavior to Connect to Wi-Fi Networks

- Automatically connect to known Wi-Fi networks
 - Cannot be disabled or configured per network easily
 - E.g. iOS configuration profile (if “Auto join” is disabled)
- If multiple known Wi-Fi networks are available...
 - iOS connects to the last-used network
 - Android: supplicant, driver, API... (e.g. Wi-Fi Ruler)
 - Windows Phone: signal strength?
 - BlackBerry: priority based on the list order (PNL)
- Network identification is based on...
 - SSID (network name) and security settings

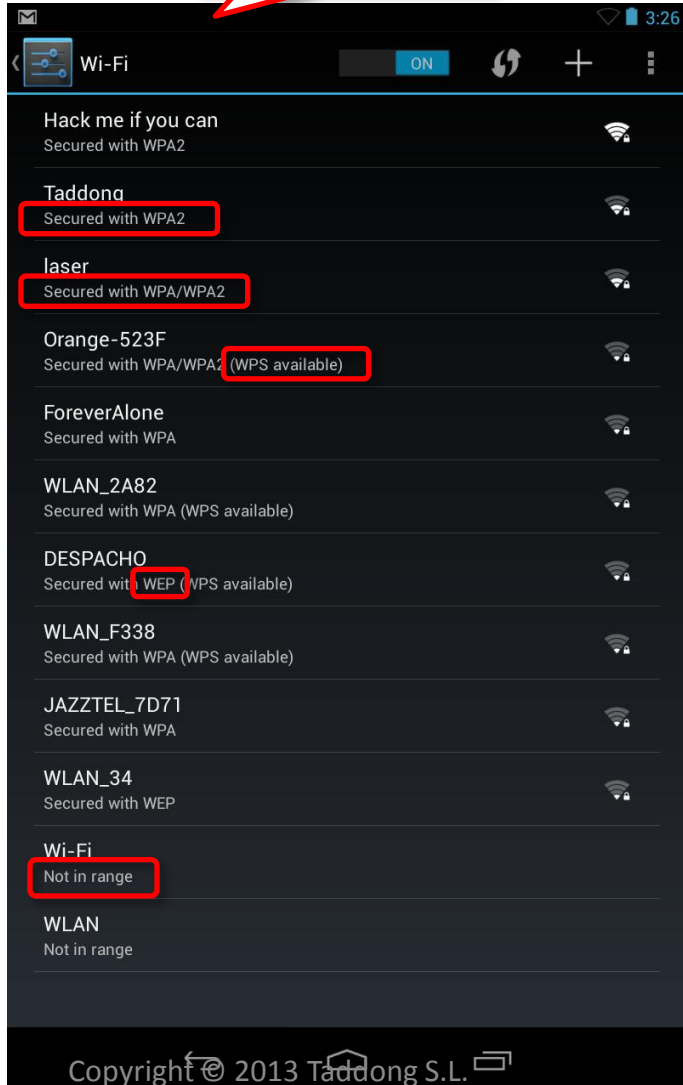




What Type of Network Are You Connecting To?



Find the diffs...



- Add the network manually to be able to verify and set all the security settings, but...



It has a lock, so it must be secure! 😊



Steve Jobs' (Apple) Minimalism





Managing the PNL in Mobile Clients

- Why the user cannot manually select the **priority** for each network in the PNL? (except in BB)
 - As in the traditional clients (e.g. Windows 7 & 8)
- Can the user **view** the list of known networks?
- Can the user **manage** (add, delete, edit...) the list of known networks?
 - Android 4.x (from the live network list – turn on Wi-Fi first)
 - BlackBerry 7.x (“Saved Wi-Fi Networks”)
 - Windows Phone 7.x & 8.x (“Advanced – Known networks”)
 - iOS minimalism...



Managing the PNL in iOS Reality



- Wi-Fi networks are easily added to the PNL...
 - ... but cannot be easily removed from the PNL
- “Forget this Network” is only available when the Wi-Fi network is in range
 - User needs to be in the area of coverage of the Wi-Fi network
 - WTF (Without Traveling Faraway)
 - Good excuse to travel for business reasons: “I have to improve the security of our Wi-Fi network and mobile devices...”



There is even a CVE-2011-4003 (check my *preso* from 2010)



Managing the PNL in iOS Solutions

- We thought about publishing a new iOS app
 - Show PNL entries
 - SSID, BSSID, security, hidden, channel, current network...
 - Manage the PNL (add, delete, edit... entries)
- iOS SDK API
 - Apple removed from AppStore all stumbler-like apps (in 2010)
 - Public API: “You can only get the SSID of the network your device is currently connected to” ☹
 - Private API: Apple80211 framework → MobileWiFi framework
- Jailbroken devices (Cydia):
 - “WiFi Passwords”: View networks and passwords in the PNL
 - “NetworkKnowledge” (\$0.99): Delete networks in the PNL
 - `/private/var/Keychains/keychain-2.db`
 - Only SSID and password (not for open networks)





Managing the PNL in iOS



- iStupid (v0.9)
 - indescreet SSID Tool (for the) Unknown PNL (on) iOS Device
- Generates beacons frames for one or multiple SSIDS (dictionary and brute force - *future*)
 - Multiple configuration options
 - Canal, SSID, BSSID, interval, rates, security settings...
- Allows to select the security settings
 - Open, WEP, WPA(2)-Personal & WPA(2)-Enterprise
 - WPA or WPA2 & TKIP or AES-CCMP (not relevant for iOS)
 - Loop



iStupid in Action

```
root@bt: ~/WiFi/iStupid
File Edit View Terminal Help

root@bt:~/WiFi/iStupid# ./iStupid.py -h
usage: iStupid.py [-h] [-c CHANNEL] [-s SSID] [-b BSSID] [-i INTERVAL]
                  [-t RATES]
                  [--wep | --wpa | --wpa2 | --wpa-enterprise | --wpa2-enterprise]
                  [--loop | --cve-2012-2619]
                  [-V]
                  interface

iStupid (v0.9):
indiscreet SSID tool (forthe)

Copyright (c) 2013 Taddong SL

Tool that creates fake Wi-Fi networks

positional arguments:
  interface              local Wi-Fi interface

optional arguments:
  -h, --help              show this help message and exit
  -c CHANNEL, --channel CHANNEL
                        Wi-Fi network channel (default = 1)
  -s SSID, --ssid SSID   Wi-Fi network name (SSID) (default = random)
  -b BSSID, --bssid BSSID
                        Wi-Fi network address (BSSID) (default = random)
  -i INTERVAL, --interval INTERVAL
                        Wi-Fi beacon interval (ms) (default = 100)
  -t RATES, --rates RATES
                        Wi-Fi network rates: 11b or 11g (default = 11g)
  --wep                  create a WEP Wi-Fi network (default = off)
  --wpa                  create a WPA-Personal Wi-Fi network (default = off)
  --wpa2                 create a WPA2-Personal Wi-Fi network (default = off)
  --wpa-enterprise       create a WPA-Enterprise Wi-Fi network (default = off)
  --wpa2-enterprise      create a WPA2-Enterprise Wi-Fi network (default = off)
  --loop                loop through the different network types (default = off)
  --cve-2012-2619        OPEN, WEP, WPA(2)-Personal, WPA(2)-Enterprise
                        CVE-2012-2619: Broadcom chipsets DoS (default = off)
  -V, --version          show version information and exit

Make those faraway Wi-Fi networks show up in the air!
root@bt:~/WiFi/iStupid#
```

```
root@bt: ~/WiFi/iStupid
File Edit View Terminal Help

root@bt:~/WiFi/iStupid# ./iStupid.py -c 6 -s WLANCORP --wpa2-enterprise mon0
Interface: mon0 [100 ms (0.1 secs) (privacy: WPA2-Enterprise) (rates: 11g)]
SSID: WLANCORP, BSSID: fe:d6:5d:cb:38:8d, Channel: 6
.....
.....^C

Stopping AP...
root@bt:~/WiFi/iStupid#
```



Disclosing the PNL for Free

- Hidden Wi-Fi networks (cloaked or non-broadcast)
 - Still today a very common security best practice...
 - ... with relevant security implications for the Wi-Fi clients
 - Beacon frames do not contain the SSID (empty)
- Visible (or broadcast) Wi-Fi networks include the SSID in their beacon frames
 - Wi-Fi clients need to know the SSID to connect to the network
- So how Wi-Fi clients connect to hidden Wi-Fi networks?
 - Wi-Fi clients have various networks (SSIDs) in their PNL
- Wi-Fi clients have to specifically ask for the hidden Wi-Fi networks in their PNL by sending probe requests containing the SSID
 - As a result they disclose their PNL !!

PNL was disclosed by Wi-Fi client in the past (2005; Win XP fix in 2007)

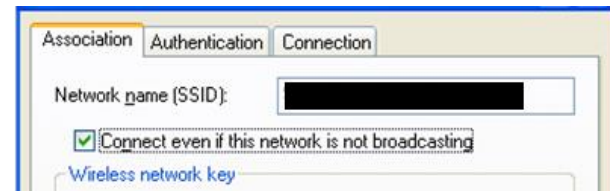
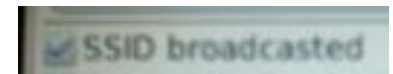


Do Mobile Devices Disclose the PNL?

- Always (not in Windows Phone 7.x or 8)
 - Windows Mobile 6.5 (TAD-2010-003)
- When networks are manually added (hidden)
 - Android 2.x – 4.x (TAD-2011-003) *Not Fixed Yet*
 - iOS ?.x – 6.x (TAD-2013-001) *NEW*
 - BlackBerry 7.x (TAD-2013-002) *NEW*
 - Can be changed afterwards through advanced settings:
- Why there is no option to indicate if a network in the PNL is hidden or not?
 - As in the traditional clients



TAKE 1



“That’s one small step for a *user*, a giant leap for *security*”



Security Risks of Disclosing the PNL

- An attacker can impersonate the various Wi-Fi networks in the PNL
 - Different methods based on the security settings
- People didn't pay enough attention to this...
 - ... because there was no name for it!



War Standing or War “Statuing” (Statue)




War Standing Risks






Tools for War Standing Activities

- Significant limitations on current Wi-Fi security, hacking, and pen-testing tools
 - Network or AP focused
 - Very limited client details
- Examples
 - Kismet(-ng)
 - Probe Networks
 - Autogroup Probe
 - Airodump-ng
 - Look at the bottom, if you can...
 - Stumbler-like tools ignore clients...



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	
QQF93	00:1F:90:F2:CB:C2	A W	1	2412	1	0B	---	---	1	ActiontecE	US	wlan0	
landscapers	00:14:BF:07:2F:84	A N	6	2437	2	0B	10%	-86	1	Cisco-Link	---	wlan0	
linksys_SES_45997	00:16:B6:1B:E4:FF	A 0	6	2447	2	0B	---	---	1	Cisco-Link	---	wlan0	
linksys	00:1A:70:D9:BC:13	A N	6	2437	2	0B	---	---	1	Cisco-Link	---	wlan0	
MPA41	00:1F:90:E6:E0:84	A W	11	2462	3	0B	---	---	1	ActiontecE	---	wlan0	
TSC	00:00:5B:07:00:03	A N	---	---	4	0B	---	---	1	Netgear	---	wlan0	
Autogroup Probe	00:13:E8:92:3F:CB	P N	---	---	5	0B	---	0	1	IntelCorpo	---	wlan0	
meskas	00:18:01:F3:05:E1	A 0	11	2462	7	0B	10%	-87	1	ActiontecE	US	wlan0	
6S103	00:1F:90:FA:F4:CB	A W	---	2412	8	0B	---	---	1	ActiontecE	---	wlan0	
Xu Chen	00:18:01:F9:70:F0	A N	6	2442	9	0B	0%	-75	1	ActiontecE	US	wlan0	
7J4R0	00:1F:90:E6:04:F1	A W	11	2462	14	0B	---	-79	1	ActiontecE	---	wlan0	
TK421	00:18:01:FE:68:77	A 0	6	2437	14	0B	---	-82	1	ActiontecE	---	wlan0	
Elina-PC-Wireless	00:24:B2:0E:E6:E2	A 0	11	2462	14	0B	0%	-31	1	Netgear	---	wlan0	
Pickles	00:1F:33:F3:C5:4A	A 0	2	2422	17	0B	---	---	1	Netgear	---	wlan0	
38c8	00:16:CE:07:60:77	A W	6	2447	38	0B	---	-76	1	NonHuiPrec	---	wlan0	



```
(wlan.fc.type_subtype == 0x04) && !(wlan_mgt.ssid == "")
```




“Faltan las Palabras”



problema.

(Del lat. *problēma*, y este del gr. πρόβλημα).

3. m. Conjunto de hechos o circunstancias que dificultan la consecución de algún fin.

problemón *m.* Un problema relevante o de notable importancia.

probremón *m.* Un problema relevante o de notable importancia, que se repite a lo largo del tiempo.

probremon.py

- probe
- request
- monitor

“Yo propongo...”



La letra pequeña (Luis Piedrahita)



probremmon.py



Coming Soon...



שנה פרט!
הבנק המרכזי של ישראל
הפך לפרטיות
David luz

davidluz2000.deviantart.com

Ask the Tarasco Bro's (Tarlogic) for GUI: "Wireless Auditing Framework"



SSID Selection & Contradictions

- Change the default SSID and select a unique one
 - WPA cracking
 - $PMK = PBKDF2(PSK, \text{SSID} + \text{length}, 4096 \text{ hmac-sha1})$
 - AP impersonation
 - Avoid guessing, dictionary, or brute force attacks on SSID
 - Top SSIDs, Top Wi-Fi Hotspots, WiGLE...
- Identifying a unique (set of) SSID(s) allows...
 - Associate network name to location (WiGLE...)
 - Anonymity attacks (personal privacy implications)
 - Targeted attacks (unique client fingerprint)



Wi-Fi PNL Conclusions

- Vendors do not read Taddong's Security Blog
 - Adding Wi-Fi networks manually == hidden network
- Why don't why make hidden Wi-Fi networks disappear in 802.11_ technologies?
- AP shouldn't provide an option to configure the Wi-Fi network as hidden
- Wi-Fi clients should never allow users to add a Wi-Fi network as hidden
 - No need then to have an option to indicate if a network in the PNL is hidden or not (minimalism)

Wi-Fi clients would not disclose their PNL... Right?



Do Mobile Devices Disclose the PNL?



Difficult to consistently reproduce them... ☹



The full PNL of iOS is disclosed in iOS 5.x & 7.x (sometimes...)

iOS Recent Wi-Fi Updates: Analysis



iOS 6 & 6.0.1

Software (SW) vs. Security (SE) Updates

- iOS 6: (Sep 19, 2012)
 - SW: <http://support.apple.com/kb/DL1578>
 - SE: <http://support.apple.com/kb/HT5503>
 - 197 security fixes
 - Wi-Fi: iOS discloses MAC of hosts of previous networks (DNAv4)
- iOS 6.0.1: (Nov 1, 2012)
 - SW: <http://support.apple.com/kb/DL1606>
 - “Improves reliability of iPhone 5 and iPod touch (5th generation) when connected to encrypted WPA2 Wi-Fi nets”
 - SE: <http://support.apple.com/kb/HT5567> (4 CVEs)



iOS 6.0.2 (Dec 18, 2012)

- SW: <http://support.apple.com/kb/DL1621>

Screenshot of the Apple Support page for iOS 6.0.2 Software Update (DL1621).

The page shows the update details, including the version (6.0.2), post date (Dec 18, 2012), and download ID (DL1621). The system requirements section is highlighted with a red box, listing iPhone 5 and iPad mini.

The "About iOS 6.0.2 Software Update" section is also highlighted with a red box, stating: "Fixes a bug that could impact Wi-Fi." This is followed by a red arrow pointing to the text: "Fixes a bug." and "Security Update?"

Available via OTA

Forums speculations:
"iOS prioritizing open networks over secure networks..."

<http://blog.taddong.com/2013/01/apples-skimpy-software-update.html>



iOS 6.1

- iOS 6.1: (Jan 28, 2013)
 - SW: <http://support.apple.com/kb/DL1624>
 - SE: <http://support.apple.com/kb/HT5642>
 - 27 CVEs + root CA's updates
 - CVE-2012-2619: “A remote attacker on the same WiFi network may be able to temporarily disable WiFi”
 - DoS condition (CVE-2012-2619)
 - October 2012 (Ekoparty - CoreLabs)
 - Broadcom Wi-Fi chipsets (iOS and others)
 - iPad 1 (iOS 5.1.1)




iOS 6.1.1 (Feb 11, 2013)

- SW: <http://support.apple.com/kb/DL1631>

<https://support.apple.com/kb/DL1631>

iOS 6.1.1 Software Update



About iOS 6.1.1 Software Update
This update fixes an issue that could impact cellular performance and reliability for iPhone 4S.

For information on the security content of this update, please visit this website:
<http://support.apple.com/kb/HT1222>

System Requirements
iPhone 4S

Supported Languages
Български, Čeština, Slovenčina, Tiếng Việt, العربية (مصر), Ελληνικά, Hrvatski, Ukrainian, Bahasa Indonesia, Deutsch, English, Français, 日本語, Español, Italiano, Nederlands, Dansk, Norsk Bokmål, Polski, Português, Português (Brasil), Русский, Suomi, Svensk, 简体中文, 繁體中文, 한국어, Română, Türkçe, Magyar, British English, עברית, Thai

Languages
English

Security Update?

<http://support.apple.com/kb/HT1222>



HT1222 up to iOS 6.1.2

support.apple.com/kb/HT1222

Security updates		
Name and information link	Released for	Release date
OS X Server v2.2.1	OS X Mountain Lion v10.8 or later	04 Feb 2013
Java for Mac OS X v10.6 Update 12	Mac OS X v10.6.8	01 Feb 2013
Apple TV 5.2	Apple TV 2nd generation and later	28 Jan 2013
iOS 6.1	iPhone 3GS and later, iPod touch (4th generation) and later, iPad 2 and later	28 Jan 2013
Apple TV 5.1.1	Apple TV 2nd generation and later	28 Nov 2012
QuickTime 7.7.3	Windows 7, Vista, XP SP2 or later	05 Nov 2011
Safari 6.0.2	OS X Lion v10.7.5, OS X Mountain Lion v10.8.2	01 Nov 2012
iOS 6.0.1	iPhone 3GS and later, iPod touch (4th generation) and later, iPad 2 and later	01 Nov 2012
Java for OS X 2012-006 and Java for Mac OS X 10.6 Update 11	Mac OS X v10.6.8, OS X Lion v10.7 or later, OS X Mountain Lion v10.8 or later	16 Oct 2012

Apple consistent and uniform updates are something from the past



Conclusions



BE UNIQUE

Apple — ~~Society~~ will tell you to do things a certain way
Don't listen to them



Wi-Fi Network Impersonation (MitM)



Attacker's Main Goal

- When some entries in the PNL are disclosed by Wi-Fi clients...
- ... force the victims to (silently) connect to the attacker's Wi-Fi network (Karma-like attacks)
 - AP impersonation (or fake AP): anywhere in the world
 - Evil-twin: area of coverage of the legitimate network
 - Strongest signal wins (*or less battery drawing network*)
- Prerequisites
 - Open: None
 - WEP & WPA(2)-PSK: Pre-shared key
 - WPA(2)-Enterprise: Certificates or... none
 - Additionally you can obtain valid user credentials



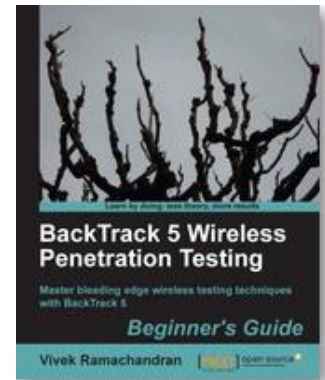
Attacking Wi-Fi (Personal) Clients



Attacking Wi-Fi Clients

*A book in Spanish
is coming soon...*

- Create a fake AP using airbase-ng
 - Impersonate the legitimate network
 - SSID & security settings
 - Obtain the pre-shared key
 - Setup full network connectivity & MitM
- “BackTrack 5 Wireless Penetration Testing Beginner’s Guide” (Vivek Ramachandran)
 - Chapter 6: “Attacking the Client” (2011-09)
 - <http://www.packtpub.com/article/backtrack-5-attacking-the-client>



<http://www.packtpub.com/backtrack-5-wireless-penetration-testing-beginners-guide/book> (Sample Chapter: 6)



Attacking Wi-Fi Clients: Open



“I never, ever, connect to an open Wi-Fi network!” Right? 😊



Attacking Wi-Fi Clients: WEP

- Caffe Latte (ToorCon 2007)

- Broadcast ARP request & flip bits & ICV

- http://www.aircrack-ng.org/doku.php?id=airbase-ng#how_does_the_caffe_latte_attack_work

```
# airbase-ng -c 1 -a 00:01:02:0a:0b:0c -e "Taddong" -W 1 -L mon0
# airodump-ng -c 1 --bssid 00:01:02:0a:0b:0c --write CaffeLatte mon0
```

- Hirte (2008)

- Fragmentation attack

- http://www.aircrack-ng.org/doku.php?id=airbase-ng#how_does_the_hirte_attack_work

```
# airbase-ng -c 1 -a 00:01:02:0a:0b:0c -e "Taddong" -W 1 -N mon0
# airodump-ng -c 1 --bssid 00:01:02:0a:0b:0c --write Hirte mon0
```

Strength of WEP key is irrelevant

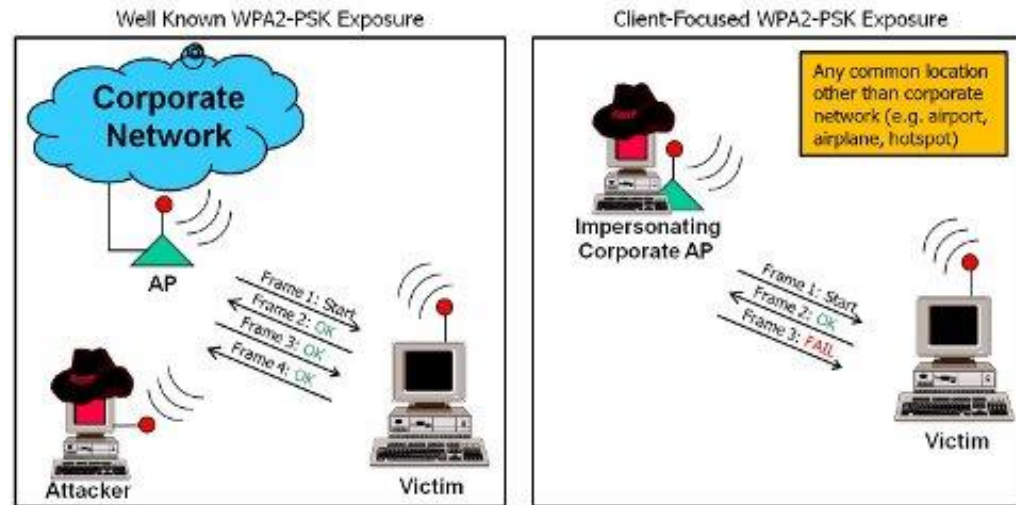
Attacking Wi-Fi Clients: WPA(2)-PSK

- AP-less WPA(2)-PSK cracking

- Out of the range of the target network
- Only requires the first two frames of the 4-way WPA(2) handshake

- E.g. cowpatty (+v4.5): “-2”

- <http://www.willhackforsushi.com/?p=284>



PMK = PBKDF2(PSK, SSID + length, 4096 hmac-sha1)
PTK = PRF-512(PMK, text, AP@, STA@, anonc, snonc)



airbase-ng

- airbase-ng
 - <http://www.aircrack-ng.org/doku.php?id=airbase-ng>
 - WEP (-W1): set the privacy bit
 - SKA: Shared Key Authentication (-s)
 - WPA (-z) & WPA2 (-Z)
 - TKIP (2) & AES (4)
 - WPA/TKIP (-z 2)
 - WPA/AES (-z 4)
 - WPA2/TKIP (-Z 2)
 - WPA2/AES (-Z 4)

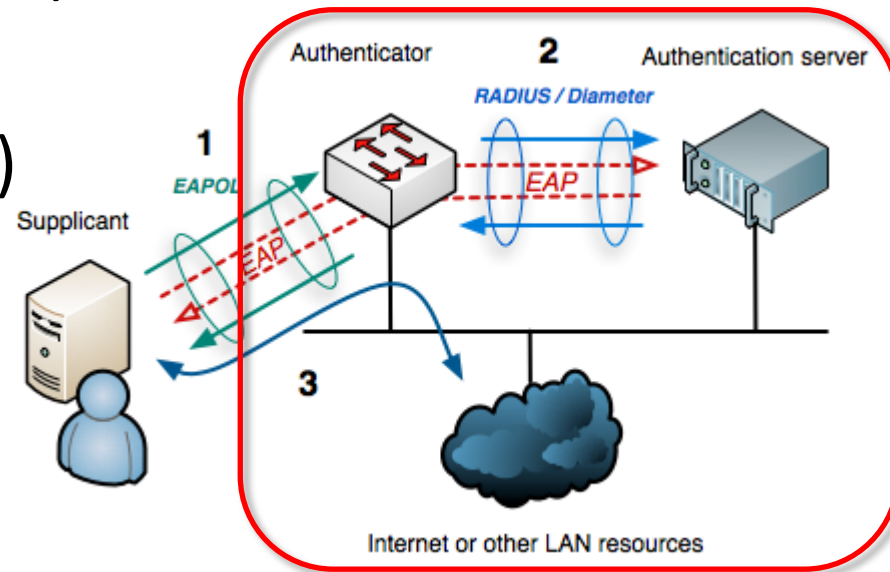
Strength of the WPA(2) PSK – 63 random characters



Attacking Wi-Fi Enterprise Clients

Wi-Fi Enterprise Networks

- Wi-Fi client, access point (AP), and RADIUS server
- Multiple user credentials allowed (802.1X/EAP types)
- How to verify the RADIUS server certificate?
 - CN, CA, expiration, revocation & purpose
 - There is no URL like in the web browsers ☹ (X.509 CN)
 - SSID (max. 32 chars) vs DNS hostname
 - Revocation (CRL & OCSP): no connection yet...
 - OCSP Stapling & 802.11u & Open Secure Wireless (OSW) & Secure Open Wireless Networking (SOWA – SOWN, e.g. XSSID)





FreeRADIUS-WPE



- FreeRADIUS-Wireless Pwnage Edition (WPE)
 - SchmooCon 2008: Joshua Wright & Brad Antoniewicz
- Attacker impersonates the full Wi-Fi network infrastructure (AP + RADIUS server)
- PEAP & TTLS
 - Inner authentication: MS-CHAPv2 (or others)
 - Username + Challenge/Response

http://www.shmoocon.org/2008/presentations/PEAP_Antoniewicz.pdf

http://www.willhackforsushi.com/?page_id=37

<http://blog.opensecurityresearch.com/2011/09/freeradius-wpe-updated.html>

<https://github.com/brad-anton/freeradius-wpe>



MS-CHAPv2 Cracking



- asleap (+v2.1) - Joshua Wright
 - Crack challenge (-C) and response (-R)
 - <http://www.willhackforsushi.com/Asleap.html>
 - Dictionary attack (DES x 3)
- genkeys
 - Precomputed MD4 hashes (indexed list of passwords)
 - Indexed by the last two bytes of MD4 hash (brute force)
 - Challenge (8-byte) & MD4 hash (16-byte) \approx Response (24-bytes)
- MS-CHAPv2 cloud cracking
 - Defcon 20 (2012): Moxie Marlinspike & David Hulton
 - <https://www.cloudcracker.com/blog/2012/07/29/cracking-ms-chap-v2/>
 - Brute force attack ($2^{56} \approx$ DES) – FPGA box: \sim 12-24h
 - www.cloudcracker.com & chapcrack (100% success rate = \$200)



Strength of user passphrase... not any more ☹



Attack Opportunities

- If the CA is not verified by the Wi-Fi client...
 - Attacker can build his own private CA...
 - Issue a certificate to impersonate target network
 - Both CA and server certs can mimic all fields from legitimate certs except fingerprint
 - Not needed most times: Wi-Fi client warnings?
- If the CA is verified by the Wi-Fi client...
 - Purchase a valid certificate from a public CA
 - Silently accepted by Wi-Fi client without extra checks
 - If the RADIUS server name (or subject) is verified, or if a private CA is used ... attack will fail

Are mobile clients vulnerable to FreeRADIUS-WPE?...



FreeRADIUS-WPE in Action

The image displays three terminal windows from a Kali Linux system, illustrating the setup and execution of the asleap tool for password recovery.

Top Left Terminal (root@mobisec-desktop): Shows the installation of FreeRADIUS. The user runs `sudo apt-get install freeradius`, which prompts for a password and confirms the installation of FreeRADIUS version 2.1.12. Subsequent commands include `sudo systemctl enable freeradius` and `sudo systemctl start freeradius`. The user then edits the `/etc/freeradius/3.0/users` file to add a user named 'raulsiles' with attributes `username = raulsiles`, `challenge = 0b:6b:c2:10:5b:5c:e2:35`, and `response = 3c:30:16:96:72:5c:34:d1:f2:8f:2e:f4:80:6d:e9:9b:e4:d8:17:df:c6:6d:8d:72`. Finally, the user runs `sudo systemctl restart freeradius`.

Top Right Terminal (root@mobisec-desktop): Shows the user running `sudo tail -f /usr/local/var/log/radius/freeradius-server-wpe.log` to monitor the logs. The logs show the successful authentication of the 'raulsiles' user.

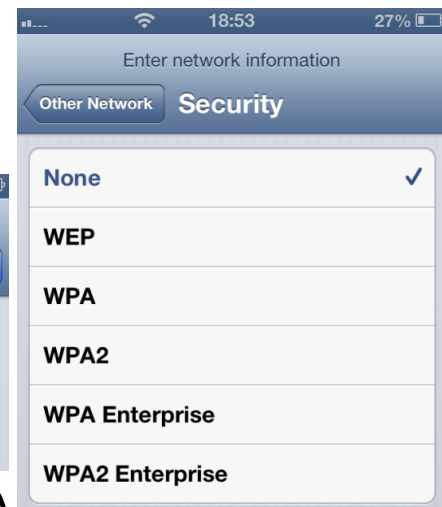
Bottom Left Terminal (root@mobisec-desktop): Shows the user editing the `/etc/freeradius/3.0/sites-enabled/default` file. The user adds a new listen block for the 'auth' type, specifying the IP address `127.0.0.1` and port `18120`. The user then runs `sudo systemctl restart freeradius`.

Bottom Right Terminal (mobisec@mobisec-desktop): Shows the user running the `asleap` tool. The user runs `./asleap-2.2/asleap -f rockyou_22+.dat -n rockyou_22+.idx -C 0b:6b:c2:10:5b:5c:e2:35 -R 3c:30:16:96:72:5c:34:d1:f2:8f:2e:f4:80:6d:e9:9b:e4:d8:17:df:c6:6d:8d:72`. The tool outputs the recovered password: `esternocleidomastoideo`.



iOS & FreeRADIUS-WPE User Interface (UI)

- iOS 5.0 - 6.1.2 (iOS ?.x)
- Default CA: None
 - Prompts user to validate certificate at first connect
 - Both to legitimate and attacker's Wi-Fi Enterprise networks
 - Does not validate CA or server certificate
- Default server name: None
- Extra (minimalism)
 - Security mode: ...
 - Advanced Wi-Fi settings?
- Attack
 - Successful (except if user rejects to connect)



“Security in the hands of the end-user”



iOS & FreeRADIUS-WPE Configuration Profile



- iOS 5.0 - 6.1.2 (iOS ?.x)
- Default CA: Undefined (optional)
 - Full list of public trusted CA's available or import other CA's
- Default server name: Undefined (optional)
- Extra (advanced settings)
 - Attacker might need to obtain a valid cert from same CA
 - No user warnings
- Attack
 - Successful (except if private CA or the server name is defined)

Enterprise Settings

Configuration of protocols, authentication, and trust

Protocols Authentication **Trust**

Trusted Certificates

Certificates trusted/expected for authentication

[No certificates available -- use 'Credentials' tab to add]

Trusted Server Certificate Names

Certificate names expected from authentication server





Android & FreeRADIUS-WPE



- Android 2.x & 4.1-4.2.2 (Android ?.x)
- Default CA: Undefined (optional)
 - CA have to be imported manually
 - Good to avoid the full list of trusted CA's (if you know what you are really doing)
 - Bad as it is optional (and will end up empty most of the time)
- Default server name: None
 - Cannot be defined ☹
- Extra (advanced settings)
 - No user warnings
- Attack
 - Successful (except if private CA)

WLANCORP	
Security	802.1x EAP
EAP method	PEAP
Phase 2 authentication	None
CA certificate	(unspecified)
User certificate	(unspecified)
Identity	paulsiles
Anonymous identity	
Password	(unchanged)
<input type="checkbox"/> Show password	
<input type="checkbox"/> Show advanced options	
Cancel	Save



WP 7.x & FreeRADIUS-WPE



- Windows Phone 7.5
- Default CA: None
- Default server name: None
- Extra (consumer device)
 - Lack of advanced settings: CA, server, EAP type...
 - First time it connects (to the legitimate Wi-Fi Enterprise network) it generates a warning 😊
 - Location: Legitimate area of coverage?
 - Not afterwards, when it connects to the attacker's network
- Attack
 - Successful (worst case scenario for WPE)

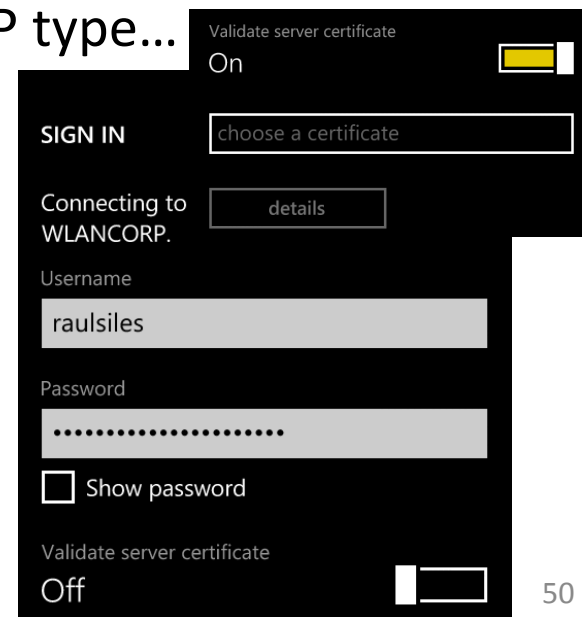




WP 8 & FreeRADIUS-WPE



- Windows Phone 8
- Default CA: Off (“Validate server certificate”)
 - Full list of public trusted CA’s available or import other CA’s
- Default server name: None
- Extra (corporate device)
 - Attacker might need to obtain a valid cert from same CA
 - Lack of advanced settings: server name, EAP type...
 - Same warning as WP 7.x (location)
 - A single authentication failure generates an unstable client state (self imposed DoS?)
- Attack
 - Successful (except if private CA)

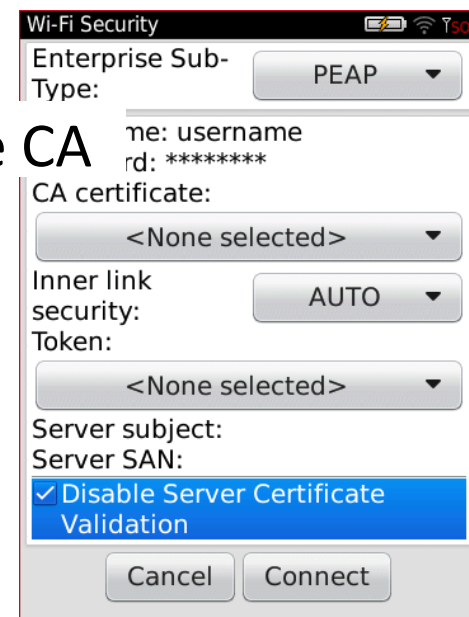




BlackBerry & FreeRADIUS-WPE



- BlackBerry 7.1.x
- Default CA: Undefined (mandatory)
 - Has to be set to be able to create the Wi-Fi profile
 - Full list of public trusted CA's available or import other CA's
- Default server name: Undefined (optional)
- Extra (advanced settings)
 - Attacker has to obtain a valid cert from same CA
 - No user warnings
 - “Disable Server Certificate Validation”
- Attack
 - Successful (except if private CA or the server name is defined)





(FreeRADIUS) EAP Dumb-Down

SANS SEC575

- Multiple EAP types available
 - Mobile devices seem to prefer to use PEAP (MS-CHAPv2) by default
- But in reality they use the preferred EAP method set by the RADIUS server
 - GTC-PAP: Log credentials in cleartext
 - Username and passphrase
- Additionally it might allow full Wi-Fi network impersonation (MitM)

Strength of user passphrase is irrelevant





EAP Dumb-Down in Action

```
root@mobisec-desktop: ~
File Edit View Terminal Help

root@mobisec-desktop:~#
root@mobisec-desktop:~#
root@mobisec-desktop:~# radiusd -X
FreeRADIUS Version 2.1.12, for host i686-pc-linux-gnu, built on Dec 10 2012 at 15:12:44
Copyright (C) 1999-2009 The FreeRADIUS server project
There is NO warranty; not even for MERCHANTABILITY or
PARTICULAR PURPOSE.
You may redistribute copies of FreeRADIUS under the
GNU General Public License v2.
Starting - reading configuration files ...
including configuration file /usr/local/etc/raddb/radiusd.conf
... adding new socket proxy address * port 34750
Listening on authentication address * port 1812
Listening on accounting address * port 1813
Listening on command file /usr/local/var/run/radiusd/radiusd.sock
Listening on authentication address 127.0.0.1 port 18120 as server inner-tunnel
Listening on proxy address * port 1814
Ready to process requests.
```

```
root@mobisec-desktop: ~
File Edit View Terminal Help

root@mobisec-desktop:~# sudo tail -f /usr/local/var/log/radius/freeradius-server-wpe.log
PAP: Wed Feb 27 17:30:55 2013

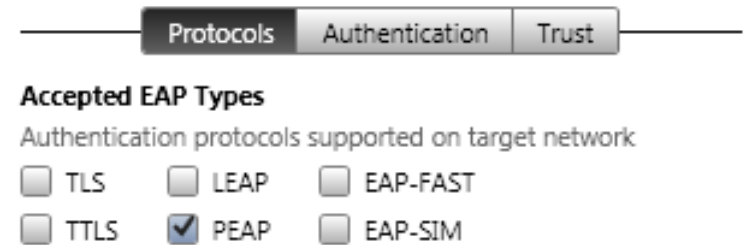
        username: raulsiles
        password: esternocleidomastoideo

PAP: Wed Feb 27 17:31:11 2013

        username: raulsiles
        password: esternocleidomastoideo
```



iOS & EAP Dumb-Down



- Default EAP type
 - Cannot be set from the UI:
 - None, WEP, WPA(2), WPA(2) Enterprise
 - Can be set via a configuration profile: PEAP
- Extra (configuration profile)
 - It is ignored, even if PEAP is the only EAP type set
- Attack
 - Same scenarios as in FreeRADIUS-WPE



Android & EAP Dumb-Down



- Default EAP type: PEAP
 - Can be set from the UI:
 - PEAP, TLS, TTLS, PWD

EAP method	PEAP
Phase 2 authentication	PEAP
CA certificate	TLS
User certificate	TTLS
Identity	PWD

- Extra
 - Can set “Phase 2 auth” (inner) in the profile
 - MS-CHAPv2 vs. None (default)

Phase 2 authentication	None
CA certificate	None
User certificate	PAP
Identity	MSCHAP
Anonymous identity	MSCHAPV2
Password	GTC

- Attack
 - Same scenarios as in FreeRADIUS-WPE
 - Except when “Phase 2 auth” is set: Not vulnerable



WP & EAP Dumb-Down



- Default EAP type: PEAP (Microsoft)
 - Cannot be set from the UI
 - PEAP is really enforced!
- Attack
 - WP 7.5 & 8 are not vulnerable by default 😊
 - Best case scenario for EAP dumb-down



BlackBerry & EAP Dumb-Down



- Default EAP type: PEAP
 - Can be set from the UI:
 - PEAP, LEAP, EAP-TLS, EAP-FAST, EAP-TTLS...
- Extra
 - Can set “Inner link security” in the profile
 - EAP-MS-CHAP v2 vs. AUTO (default)
- Attack
 - Same scenarios as in FreeRADIUS-WPE
 - Except when “Inner link security” is set: Not vulnerable



Wi-Fi Enterprise Clients

Conclusions

- Wi-Fi Enterprise is inherently “broken”
 - How to add a new RADIUS server?
 - Modify the config of all Wi-Fi clients in the organization
- Wi-Fi supplicants must always...
 - Trust only the specific CA used for the Wi-Fi network
 - Not a good idea to use the full list of public trusted CA's
 - Private CA's are a better option assuming an attacker cannot get a legitimate certificate from them
 - Define the specific (set of) RADIUS server(s) used (X.509 CN)
 - Do not provide options to disable certificate validation
 - Define and force the specific EAP type used
 - Define the inner authentication method (e.g. MS-CHAPv2)
 - Do not downgrade to other EAP types (dumb-down)

All vendors have been notified about the EAP dumb-down vulns



Wi-Fi Enterprise Clients

WPE & Dumb-Down Countermeasures

- iOS
 - Create a very strict and narrow configuration profile
 - Still “vulnerable” to EAP dumb-down (not if server is defined)
 - Still the standard UI allows adding Wi-Fi Enterprise networks
- Android
 - Import and define CA, and set inner authentication
 - Still vulnerable to WPE (server cannot be defined)
- Windows Phone
 - WP7: Fully vulnerable to WPE (not to EAP dumb-down)
 - WP8: Define CA (still vulnerable to WPE & DoS?)
- BlackBerry (manually or BES)
 - Define CA and server, and set inner authentication

Why are they vulnerable to WPE & EAP dumb-down by default?



Wi-Fi Enterprise Clients

iOS Suggestions



- iOS options: minimalism vs. advanced settings
 - Disable adding Wi-Fi Enterprise networks through the standard UI completely (minimalism)
 - Add full advanced settings for Wi-Fi Enterprise networks through UI (mandatory config profiles)

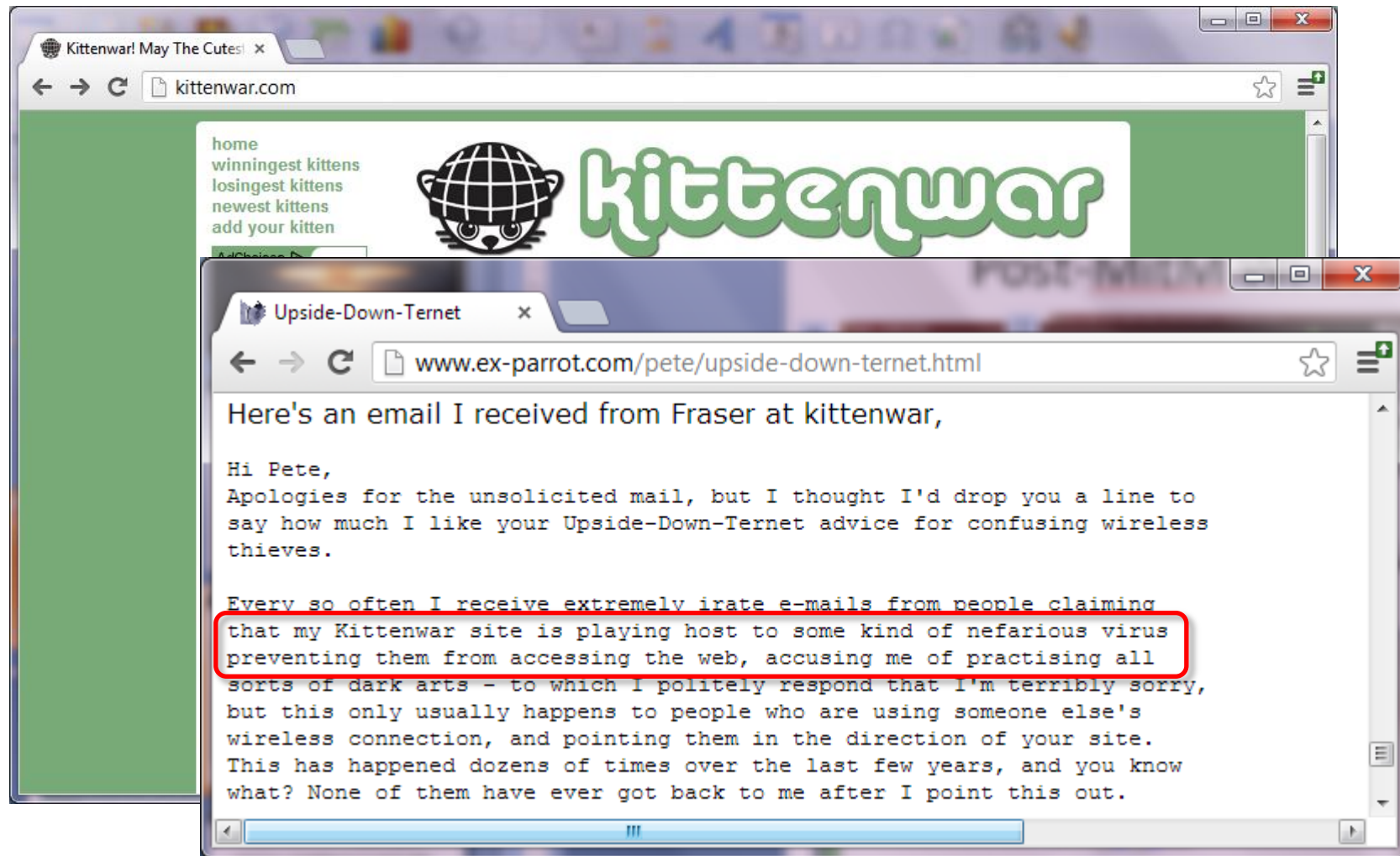




Post-MitM Wi-Fi Exploitation



Post-MitM Wi-Fi Exploitation For Fun





Post-MitM Wi-Fi Exploitation

- Wi-Fi driver vulnerabilities (if full connection state)
 - Remote code execution (ring 0) or DoS
- Mobile device fingerprinting
 - Traffic fingerprinting
 - Open ports (TCP/UDP) fingerprinting
- Traffic interception (Layer 2 and above)
 - HTTP (e.g. Firesheep, airpwn(-MSF), and others)
 - HTTPS (e.g. iOS untrusted certificate binding)
 - Other protocols
- Mobile client vulnerabilities
 - Karmetasploit



Impersonate any service (DHCP, DNS, mail...) and the whole Internet



Traffic Interception: HTTPS

- HTTPS & iOS untrusted certificate binding
 - Untrusted certs that are accepted by the user once, will remain on iOS forever (Safari Mobile)
 - RADIUS certificates? (1st time only)
 - Digital certificates cannot be managed
 - Configuration profile: Restrictions
 - Apple knows about it at least since iOS 5.x

Security and Privacy

Enforce security and privacy policies

- ☐ Allow diagnostic data to be sent to Apple
- ☐ Allow user to accept untrusted TLS certificates



Attacker's certificate is trusted by iOS forever after user accepts it

“You think that’s air you’re breathing now?”



Morpheus to Neo during the scene when he was teaching him in the virtual dojo on board the ship The Nebuchadnezzar





Thanks To...

- RootedCON
- Joshua Wright
 - FreeRADIUS-WPE
 - SANS SEC575: Mobile Device Security and Ethical Hacking
<https://www.sans.org/course/mobile-device-security-ethical-hacking>
- Mónica (testing, ideas & inspiration)
- Jorge Ortiz (iOS SDK APIs)
- Siletes & Tuno (WP 7.x)
- Mariana (BB)
- @omarbv (WP 8)
- David & José (challenges)
- Those who left us & Those that still are here



To all the vendors in advance for fixing all these things ☺



References

- “Wi-Fi (In)Security: All Your Air Are Belong To...”
 - [http://www.taddong.com/docs/Wi-Fi_\(In\)Security_GOVCERT-2010_RaulSiles_Taddong_v1.0_2pages.pdf](http://www.taddong.com/docs/Wi-Fi_(In)Security_GOVCERT-2010_RaulSiles_Taddong_v1.0_2pages.pdf)
- Taddong Security Advisories
 - <http://blog.taddong.com/p/security-advisories.html>
- Taddong’s Security Blog & Lab
 - blog.taddong.com & www.taddong.com/en/lab.html
- Raul Siles’ Wi-Fi Security
 - <http://www.raulsiles.com/resources/wifi.html>
- OWISAM (Tarlogic): www.owisam.org
 - OWISAM-TR-009, OWISAM-DI, OWISAM-CT...

Check advisories & tools publication on Taddong’s Security Blog & Lab



Thank You



Taddong

www.taddong.com

@taddong