



POLITECNICO
MILANO 1863



Fondamenti di Internet e Reti

Antonio Capone, Achille Pattavina,
Francesco Musumeci, Matteo Cesana



POLITECNICO
MILANO 1863



2 – Wireless sniffing

**Antonio Capone, Achille Pattavina,
Francesco Musumeci, Matteo Cesana**

Introduzione

- Al giorno d'oggi, siamo completamente circondati da reti wireless:
 - WiFi
 - 3G e LTE
 - Bluetooth, Zigbee, SigFOX.....
- Caratteristica comune a tutte le reti wireless è l'essere **broadcast** per definizione
- Qualsiasi ricevitore nel raggio di comunicazione di un trasmettitore può ascoltare la comunicazione



Sniffing

- Si definisce **sniffing (eavesdropping)** l'attività di intercettazione passiva dei dati che transitano in una rete
- E' possibile effettuare sniffing wireless semplicemente con un ricevitore sintonizzato sulla stessa frequenza del trasmettitore e conoscendo il protocollo di comunicazione
- Chiaramente, la maggior parte dei protocolli wireless è protetta tramite crittografia (solo chi ha la giusta chiave può decodificare i messaggi) – (WPA, WPA2, AES...)



WiFi sniffing

- Il protocollo IEEE 802.11 (Wi-Fi) prevede alcuni **messaggi in chiaro**, tra cui:
 - **Beacons**: inviati dagli access points, contengono informazioni sulla rete (SSID, rate supportati, ecc...)
 - **Probe requests** (inviati dai terminali per eseguire active scanning delle reti disponibili)
- Entrambi i messaggi contengono informazioni interessanti sul trasmettitore e possono essere “sniffati” senza difficoltà!



Obbiettivi del laboratorio

- Imparare a fare sniffing WiFi
- Analizzare i dati ottenuti dallo sniffing per rispondere alle seguenti domande:
 - Quanti device WiFi sono presenti in questa stanza?
 - Di che marca sono?
 - Dove si trovano?
- Strumenti utilizzati:
 - tcpdump / wireshark
 - PyCharm
 - Python matplotlib

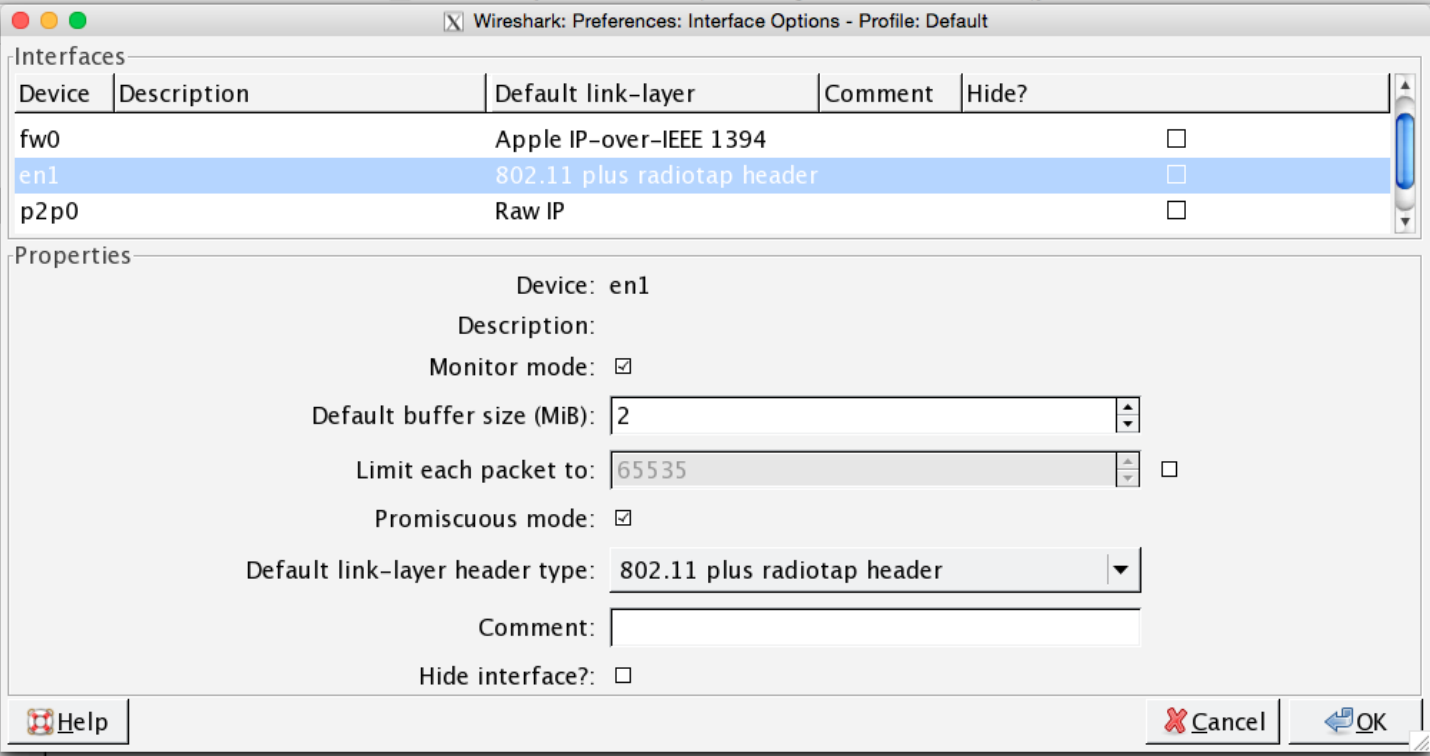


La modalità “monitor”

- E' la modalità di utilizzo dell'interfaccia WiFi con cui è possibile ascoltare il traffico su un determinato canale, **senza** essere associati a una particolare rete
- Per attivare la modalità monitor è solitamente necessario avere i privilegi di amministratore:
 - Linux (interfaccia wireless “wlan0” su canale 6)
 - `sudo ifconfig wlan0 down/up`
 - `sudo iwconfig wlan0 mode monitor chan 6`
 - Mac OS X (interfaccia wireless “en1” su canale 6)
 - `sudo airport en1 sniff 6`
 - E' anche possibile attivare la modalità direttamente da wireshark (se eseguito con privilegi da amministratore)



Monitor mode in wireshark



The image shows the 'Wireshark: Preferences: Interface Options - Profile: Default' dialog box. The 'Interfaces' table lists three interfaces: fw0, en1, and p2p0. The 'en1' interface is selected and highlighted in blue. Below the table, the 'Properties' section for the selected interface 'en1' is shown. The 'Monitor mode' checkbox is checked, and the 'Default link-layer header type' is set to '802.11 plus radiotap header'. Other settings include 'Default buffer size (MiB): 2', 'Limit each packet to: 65535', 'Promiscuous mode: checked', 'Comment: (empty)', and 'Hide interface?: unchecked'. Buttons for 'Help', 'Cancel', and 'OK' are visible at the bottom of the dialog.

Device	Description	Default link-layer	Comment	Hide?
fw0		Apple IP-over-IEEE 1394		<input type="checkbox"/>
en1		802.11 plus radiotap header		<input type="checkbox"/>
p2p0		Raw IP		<input type="checkbox"/>

Properties

Device: en1

Description:

Monitor mode:

Default buffer size (MiB): 2

Limit each packet to: 65535

Promiscuous mode:

Default link-layer header type: 802.11 plus radiotap header

Comment:

Hide interface?:



WiFi sniffing: tcpdump

- Sui sistemi Unix (linux e Mac OS X), è possibile usare il programma tcpdump per fare sniffing (windump su windows)
- Passi da seguire (Linux):
 - Mettere l'interfaccia wifi (es. "wlan0") in modalità monitor
 - Da linea di comando, digitare:

```
sudo tcpdump -i wlan0 -n -e -s 256 type mgt subtype probe-req > out.txt
```

Opzione	
-i <iface>	Specifica l'interfaccia di cattura
-n	Non converte indirizzi in nomi
-e	Stampa gli header del livello data-link
-s <len>	Cattura fino a <len> bytes per pacchetto
type (mgt ctl data) [subtype <subtype>]	Filtra i pacchetti sulla base del tipo

La stringa "> out.txt" reindirige l'output del programma sul file "out.txt"



WiFi sniffing: tcpdump

- Passi da seguire (Mac OS X)

```
sudo tcpdump -I en1 -e -s 256 type mgt subtype probe-req > out.txt
```

- Su Mac OS X, l'opzione `-I` mette automaticamente l'interfaccia selezionata in modalità monitor!
- E' possibile cambiare subtype e filtrare solo "beacon"



Analizziamo l'output di tcpdump

- Il file out.txt conterrà tante righe (una per ogni pacchetto sniffato), con (tra le altre) le seguenti informazioni:

1.0 Mb/s 2412 MHz 11g -83dB signal BSSID:ff:ff:ff:ff:ff:ff DA:ff:ff:ff:ff:ff:ff SA:bc:67:78:4d:13:e6

- E' quindi possibile leggere in chiaro il MAC address di chi ha trasmesso la probe request e la potenza del segnale!
- La potenza del segnale ricevuto è inversamente proporzionale alla distanza: si può anche stimare a che distanza è il trasmettitore!



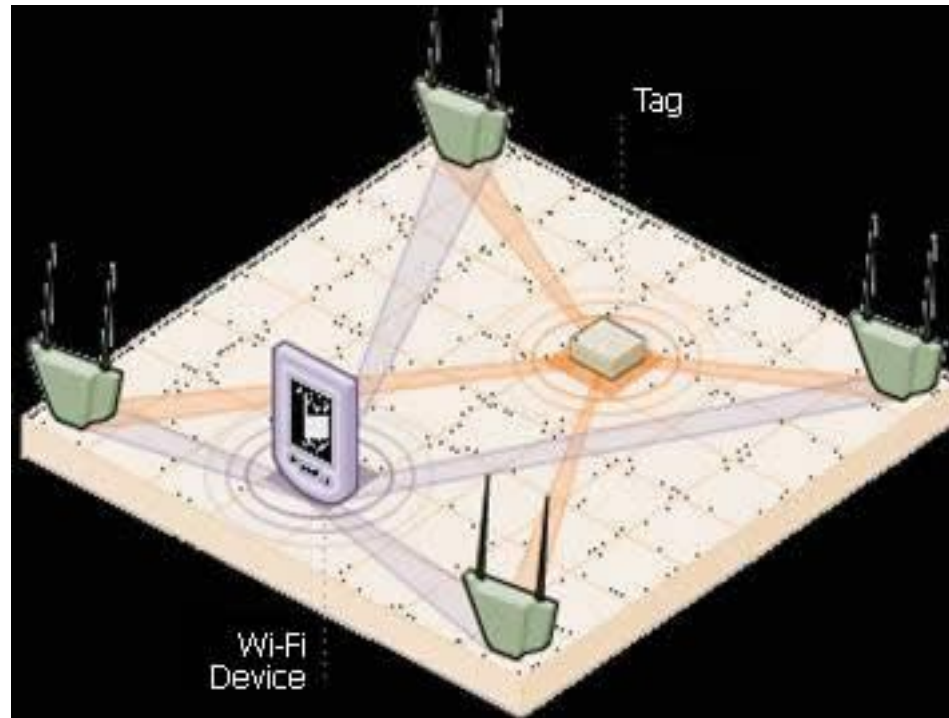
Un piccolo esempio in python...

- Vediamo che informazioni è possibile ricavare facendo sniffing in quest'aula...



Localizzazione tramite WiFi sniffing

- Abbiamo visto che la potenza a cui si riceve il segnale è inversamente proporzionale alla distanza
- E' possibile usare questa informazione per capire la propria posizione?



Localizzazione

- In quest'aula ci sono 2 access points noti che periodicamente inviano dei messaggi (beacon) sul ch 1
- “Sniffando” i beacon posso stimare la distanza a cui mi trovo rispetto a ciascun access point

```
sudo tcpdump -i wlan0 -n -e -s 256 type mgt subtype beacon and \
(ether host aa:aa:aa:aa:aa:aa or ether host bb:bb:bb:bb:bb:bb) > beacon.txt
```

- Un algoritmo di **triangolazione** mi permette di stimare la mia posizione!
- Vediamo un esempio in python...



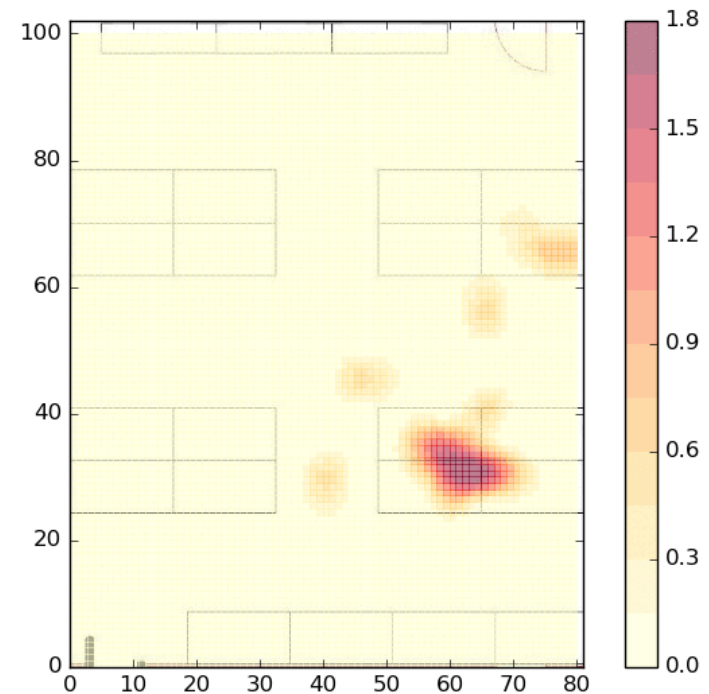
Localizzazione tramite probe request

- Posso anche pensare di invertire il processo di localizzazione:
 - Ascolto le probe request di un MAC address in punti diversi tramite sniffing
 - Uso la triangolazione per localizzare un particolare MAC address!!!



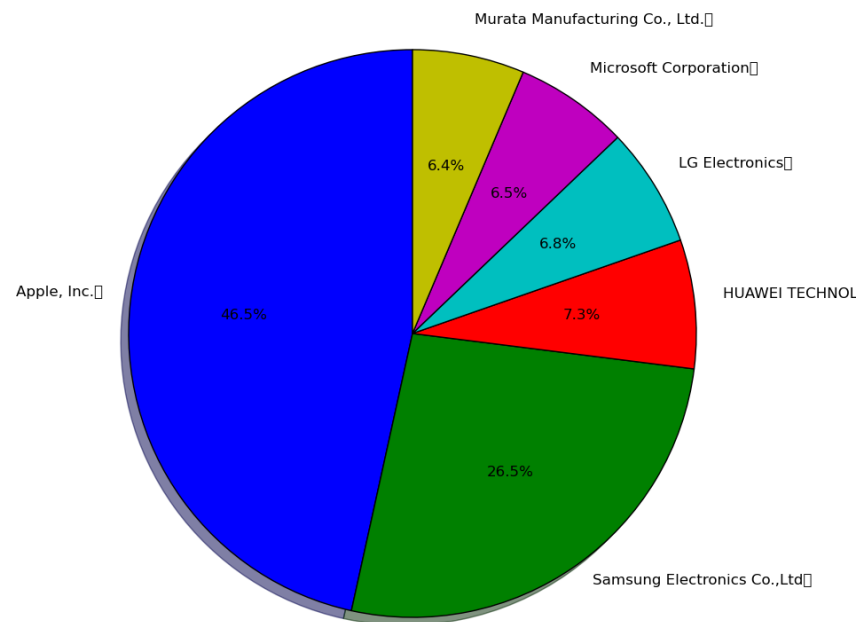
Altre applicazioni

- Tramite sniffing di probe request è possibile implementare diverse interessanti applicazioni:
- Quali sono le zone più visitate in:
 - Supermercati / fiere
 - Centri commerciali
 - Concessionari
 - etc...
- Quanti studenti ci sono in un'aula?
- Altro?



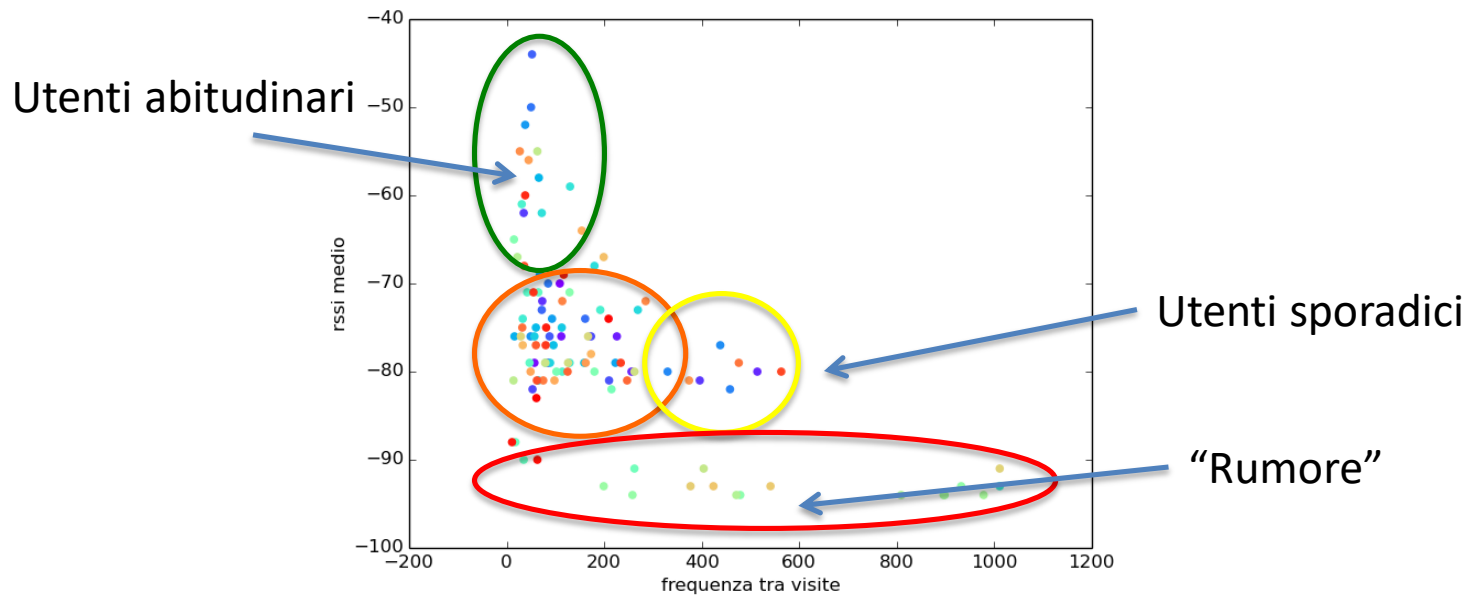
Altre applicazioni

- Qual è la fetta di mercato dei diversi produttori in una certa area target? (vendor analysis)



Altre applicazioni

- Caratterizzazione degli utenti
 - Quante volte ho visto un particolare MAC address?
 - Con che potenza del segnale?
 - Con che frequenza (oraria / giornaliera / settimanale)?



Altre applicazioni

- Caratterizzazione degli utenti
 - Le probe request contengono anche informazioni sulla provenienza/luoghi visitati dagli utenti!
 - Ogni access point cercato in una probe request può essere geolocalizzato!

