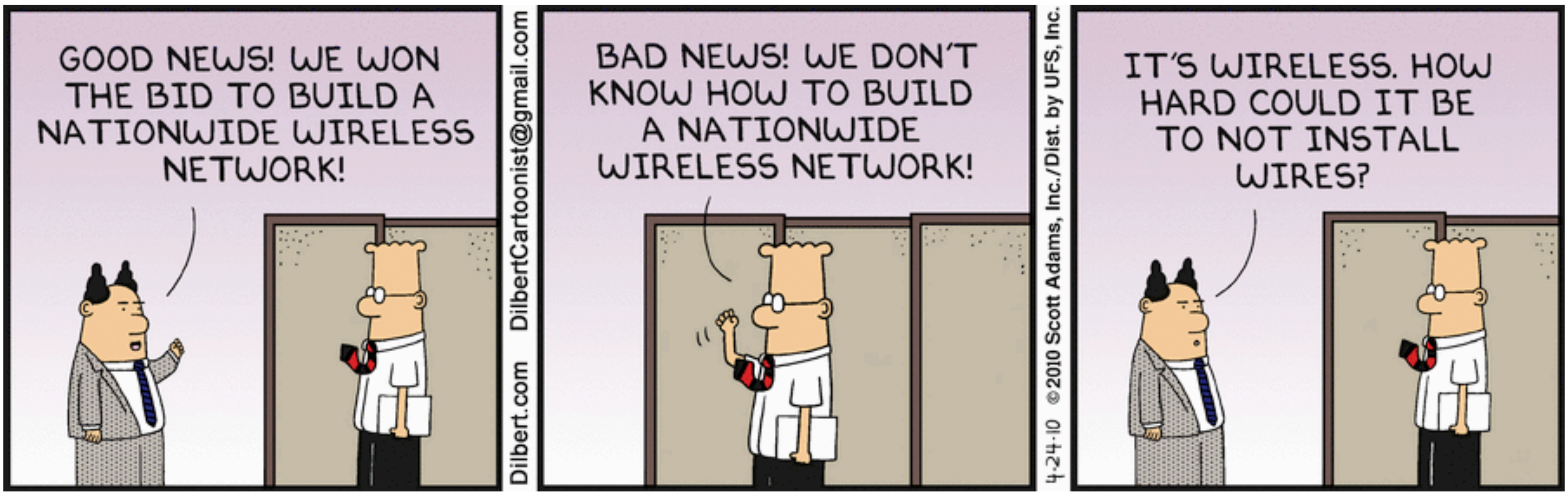
Advanced Computer Networking

CYBR 230 – Jeff Shafer – University of the Pacific

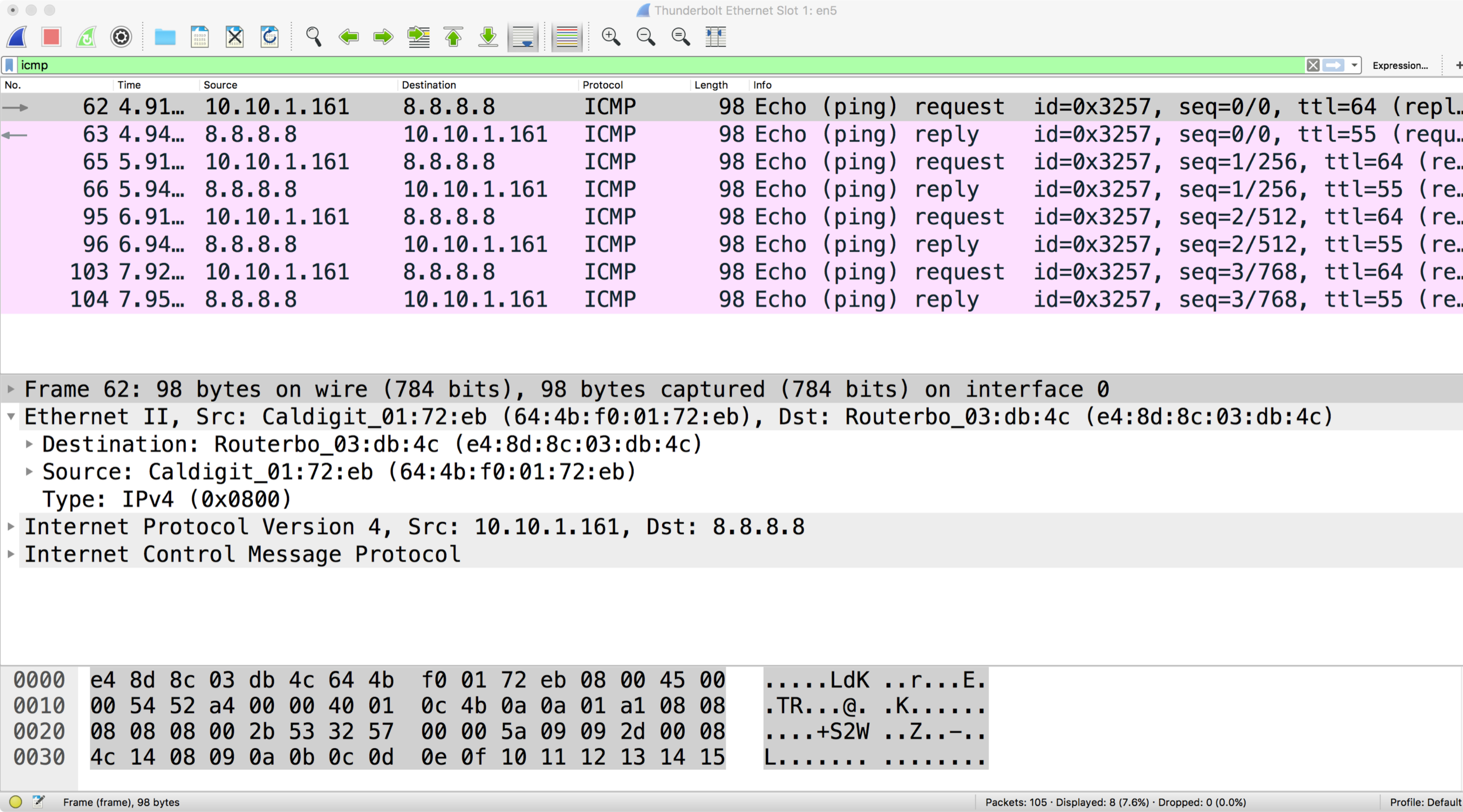
IEEE 802.11


GOOD NEWS! WE WON THE BID TO BUILD A NATIONWIDE WIRELESS NETWORK!
Dilbert.com DilbertCartoonist@gmail.com
BAD NEWS! WE DON'T KNOW HOW TO BUILD A NATIONWIDE WIRELESS NETWORK!
4-24-10 © 2010 Scott Adams, Inc./Dist. by UFS, Inc.
IT'S WIRELESS. HOW HARD COULD IT BE TO NOT INSTALL WIRES?

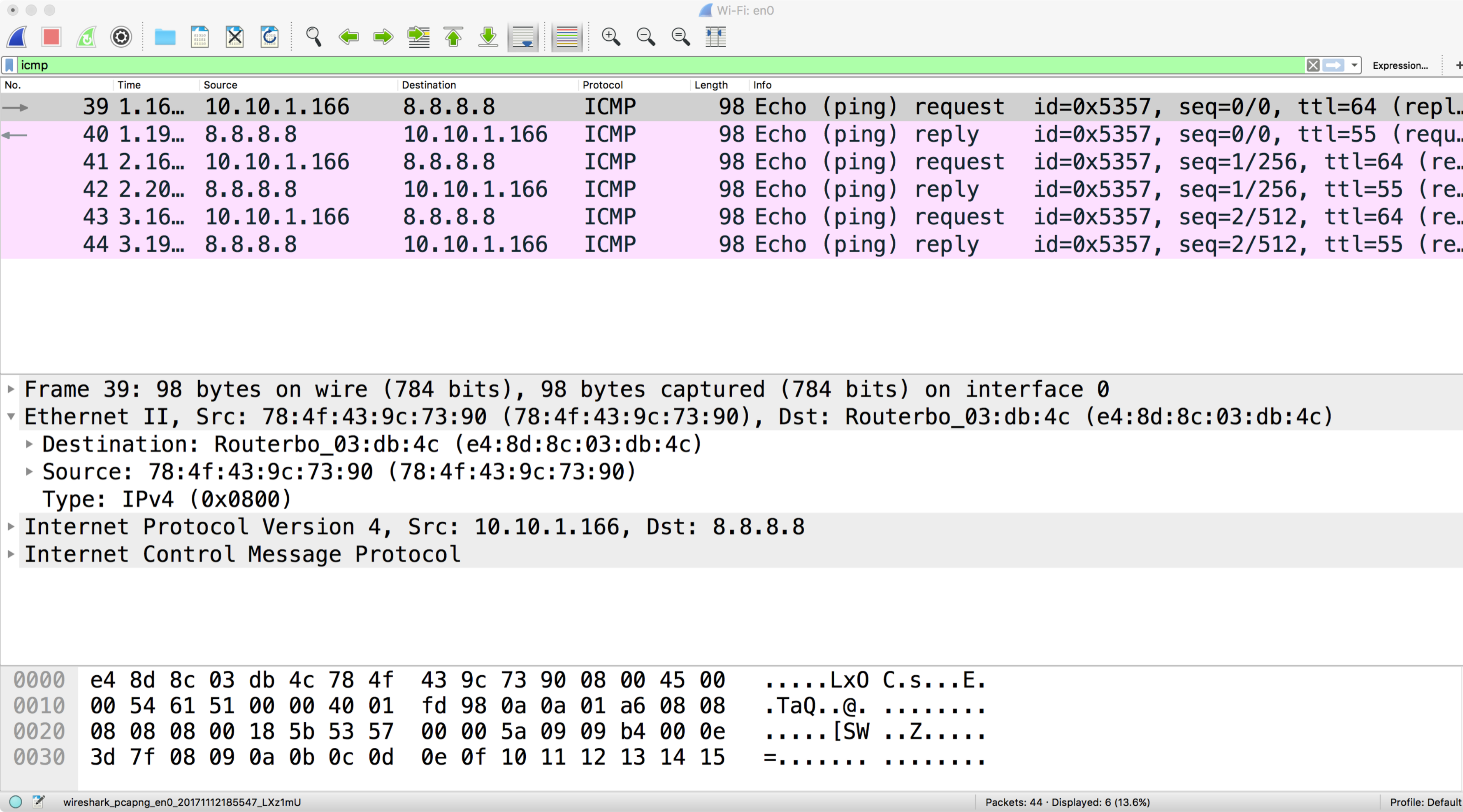
Classic Network Model

(Not ISO model, but as actually implemented)

	Application Layer	HTTP, ….
Layer 4	Transport Layer	TCP, ….
Layer 3	Network Layer	IP (IP addresses, routers, ….)
Layer 2	Link Layer	Ethernet (MAC addresses, switches, …)
Layer 1	Physical Layer	Bits on a wire…



Wireshark capture of *wired* Ethernet



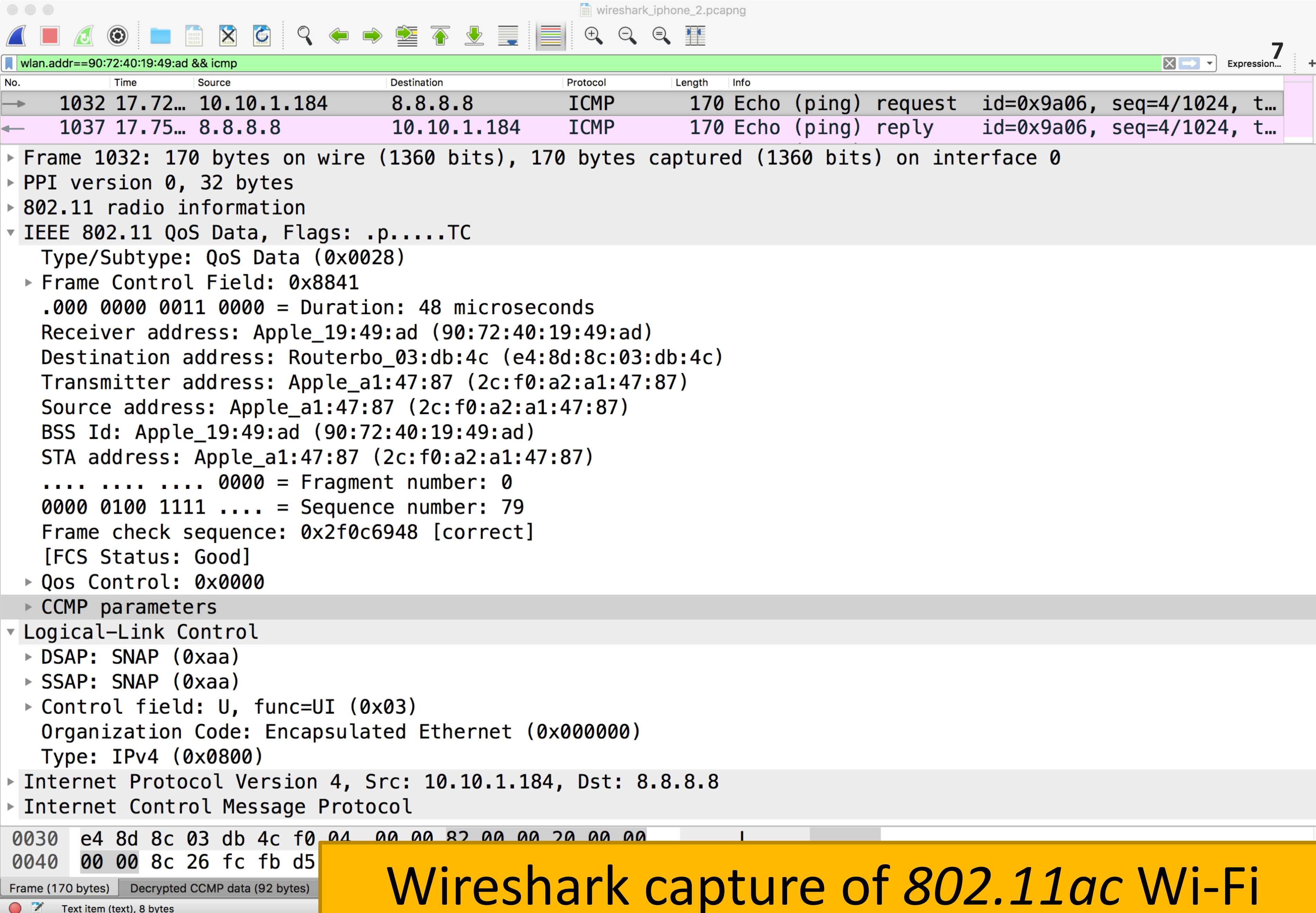
Wireshark capture of *802.11ac* Wi-Fi

Looks like wired Ethernet, so lecture over, right?

802.11

802.11 looks like Ethernet

… but only at the network layer and above



Wireshark capture of *802.11ac* Wi-Fi

With station in *monitor mode*

Network Model

	Application Layer	HTTP, ….
Layer 4	Transport Layer	TCP, ….
Layer 3	Network Layer	IP (IP addresses, routers, ….)
Layer 2	Link Layer	802.2 Logical Link Control (LLC) 802.11 MAC header *(identical a,b,g,n,…)*
Layer 1	Physical Layer	802.11 PLCP header *(varies a,b,g,n,…)* Physical media (DSSS, OFDM, …)

IEEE 802.11 Physical Layer

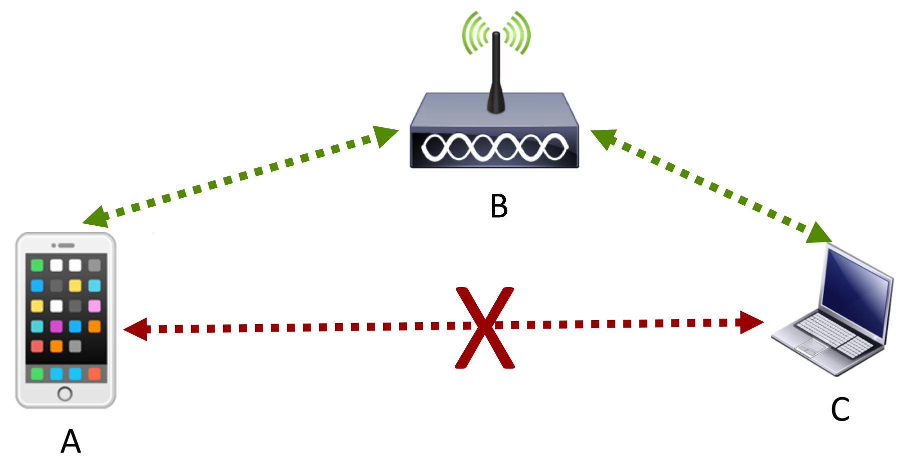
IEEE 802.11 Physical Layer

Physical Layer (PHY)

- Purpose: Transmit raw bits over a physical link
  - Copper wire, optical cable, **wireless**
- Challenges
  - Convert input bitstream into symbols/code words?
  - Frequencies to transmit on?
  - Modulation scheme?
- **Layer 1**

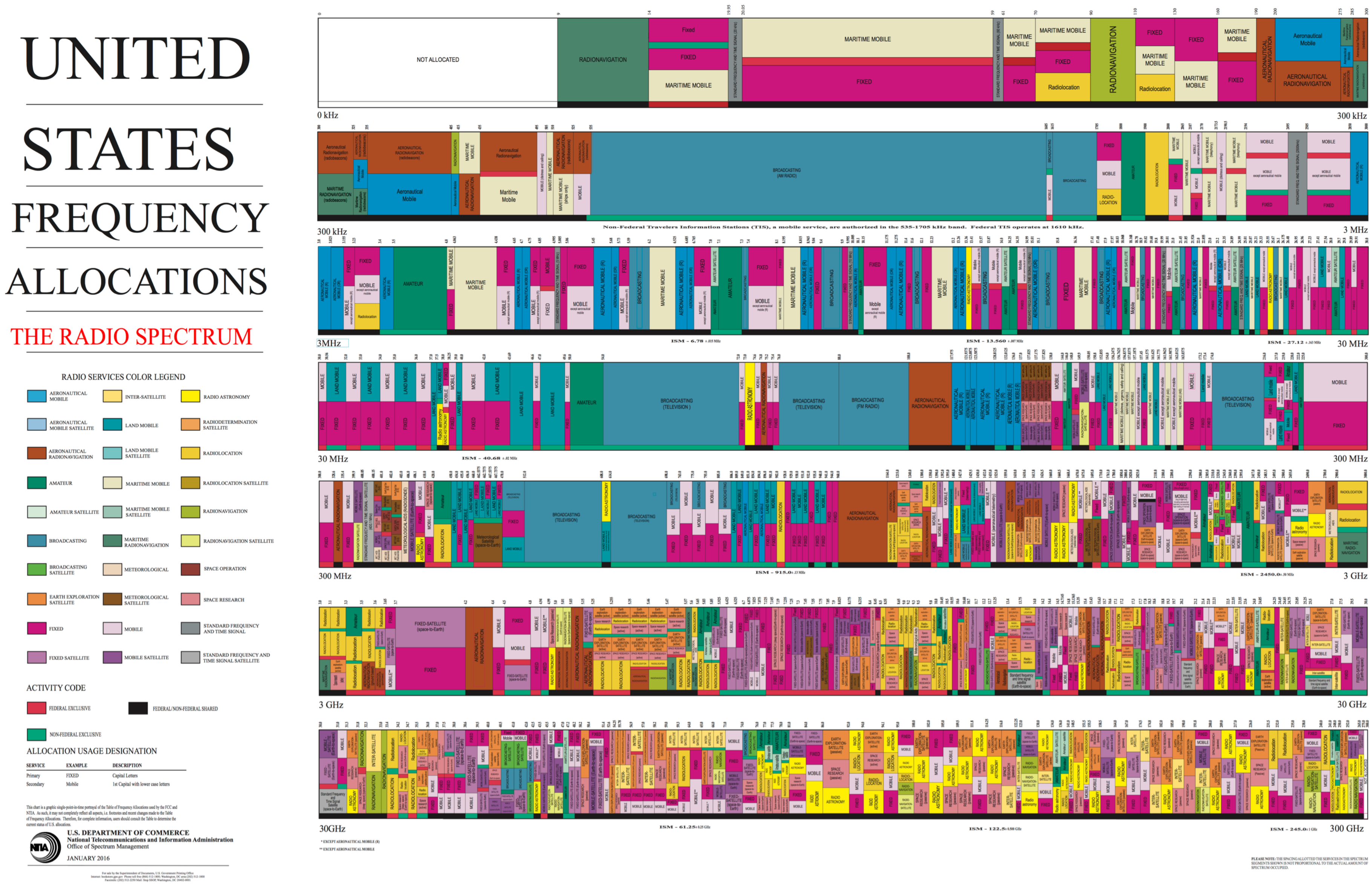
Physical Layer Challenges

- Stations can move
  - Changes propagation delays and signal strength
- Non-transitive reception
  - A can hear B
  - B can hear C
  - A cannot hear C
- No collision detection
  - Must detect unsuccessful transmission by absence of acknowledgement



Physical Layer Challenges

- Range of network limited by transmission power
  - Limits end-to-end propagation delay
- Radio Frequency (RF) spectrum usage limited by law and treaty
  - 802.11 uses 2.4 GHz and 5 GHz bands
    - Industrial, Scientific, Medicine (ISM) bands
    - Unlicensed National Information Infrastructure (U-NII)
  - Must use spread spectrum technology to minimize interference with other devices



https://www.ntia.doc.gov/page/2011/united-states-frequency-allocation-chart [Last Update: 2016]

It sure *looks* fast....

802.11 Physical Layer Standards

802.11 Protocol	Release date	Frequency (GHz)	Bandwidth (MHz)	Stream data rate (Mbit/s)	Allowable MIMO streams	Modulation	Approximate range Indoor	Approximate range Outdoor
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	35 m (115 ft)	120 m (390 ft)
		3.7						5,000 m (16,000 ft)
b	Sep 1999	2.4	22	1, 2, 5.5, **11**	N/A	DSSS	35 m (115 ft)	140 m (460 ft)
g	Jun 2003	2.4	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	38 m (125 ft)	140 m (460 ft)
n	Oct 2009	2.4/5	20	Up to **288.8**	4	MIMO-OFDM	70 m (230 ft)	250 m (820 ft)
			40	Up to 600				
ac	Dec 2013	5	20	Up to **346.8**	8	MIMO-OFDM	35 m (115 ft)	
			40	Up to 800				
			80	Up to 1733.2				
			160	Up to 3466.8				

https://en.wikipedia.org/wiki/IEEE_802.11

Frequency

2.4 GHz

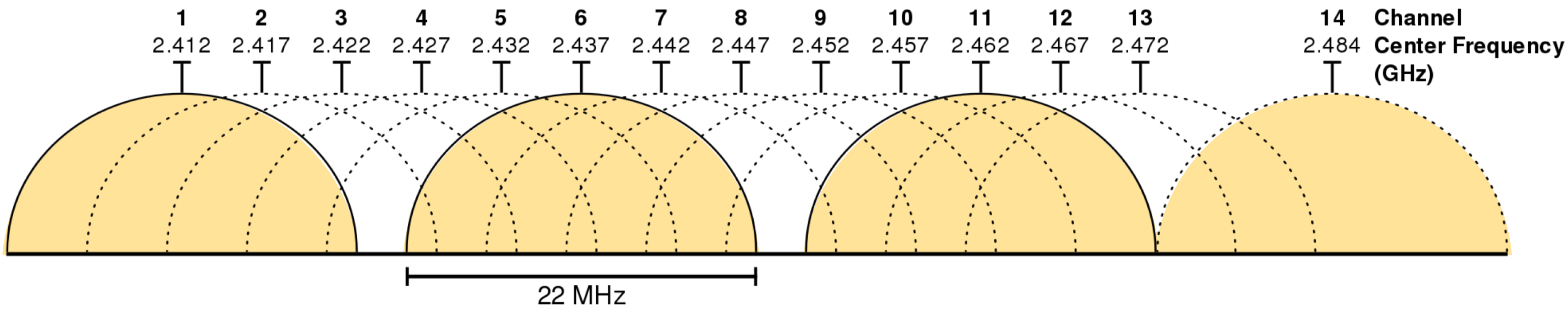
- Longer range
- Lower data rate
- Increased penetration of walls and floors
- Particularly crowded
  - Used by many other devices besides WiFi (cordless phones, Bluetooth, wireless microphones, …)
  - Subject to interferences (microwave ovens)

5 GHz

- Shorter range
- Higher data rate due to higher frequency
- Attenuated more severely by walls and floors

Each increment in **channel number** is +5MHz

802.11 2.4 GHz Channels



2.4 GHz: Channels 1-11 valid in North America
Only 3 non-overlapping channels! (Or 4 in Japan)

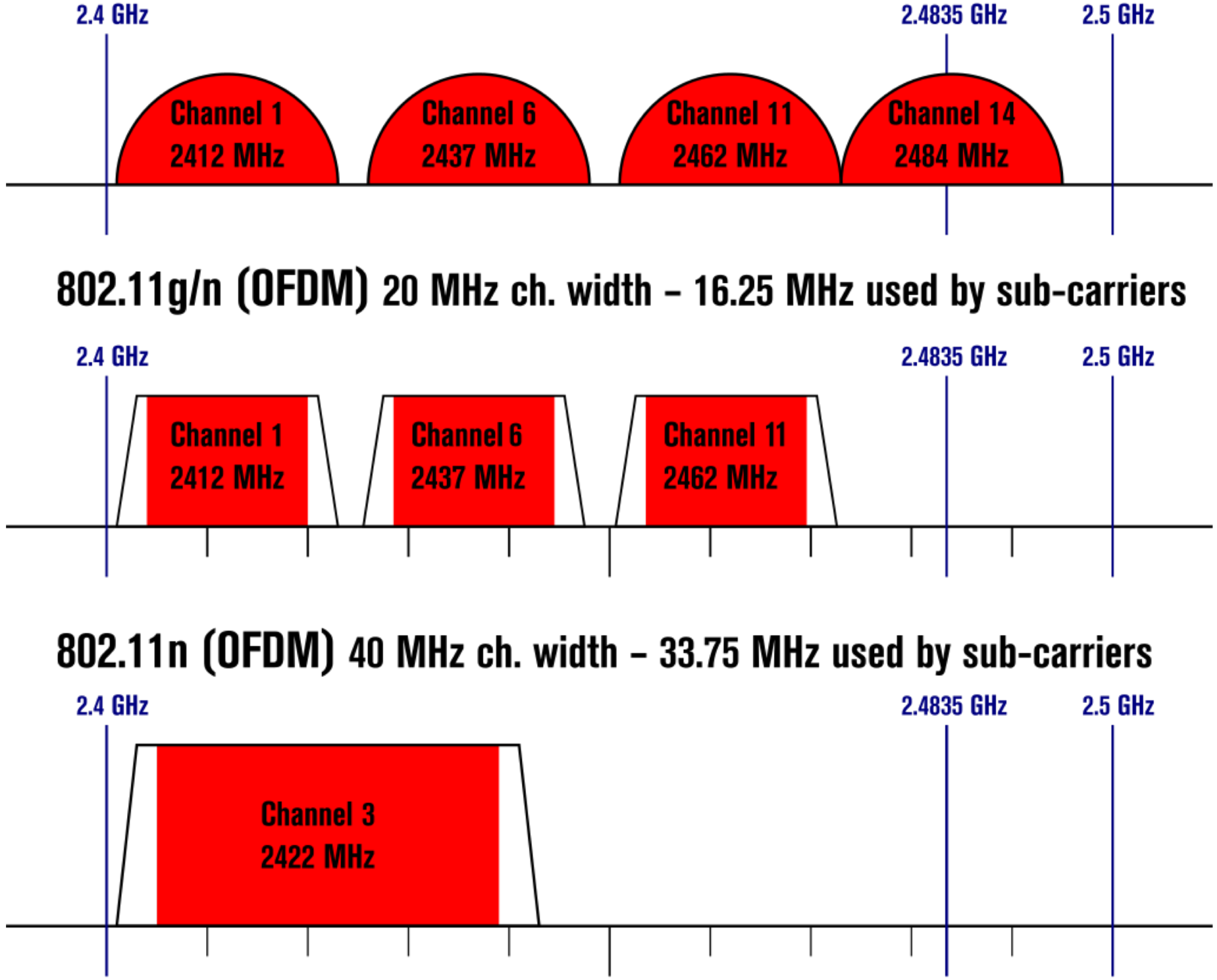
https://en.wikipedia.org/wiki/List_of_WLAN_channels

Non-Overlapping Channels for 2.4 GHz WLAN

802.11b (DSSS) channel width 22 MHz

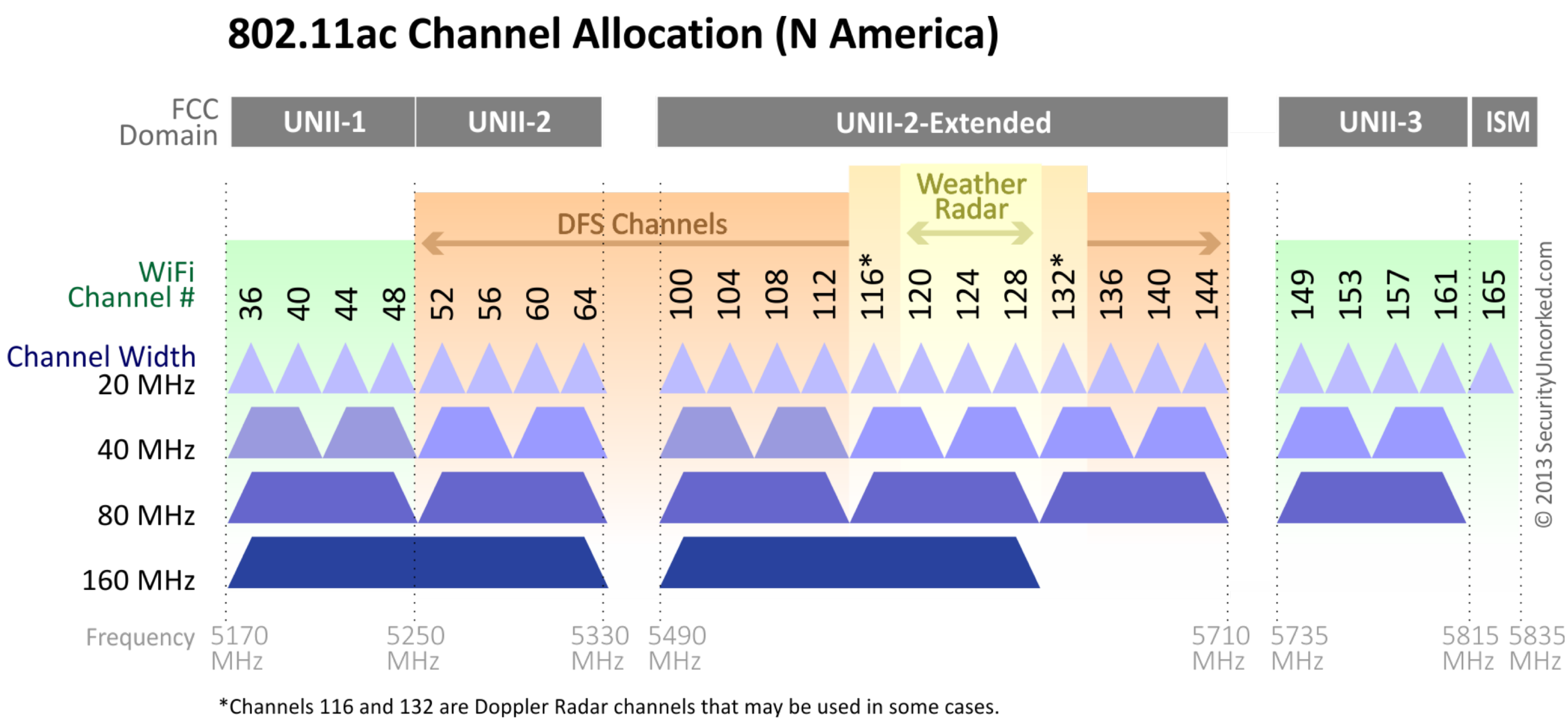
802.11g/n (OFDM) 20 MHz ch. width – 16.25 MHz used by sub-carriers

802.11n (OFDM) 40 MHz ch. width – 33.75 MHz used by sub-carriers



https://en.wikipedia.org/wiki/List_of_WLAN_channels

802.11 5GHz Channels



http://securityuncorked.com/2013/11/the-best-damn-802-11ac-channel-allocation-graphics/

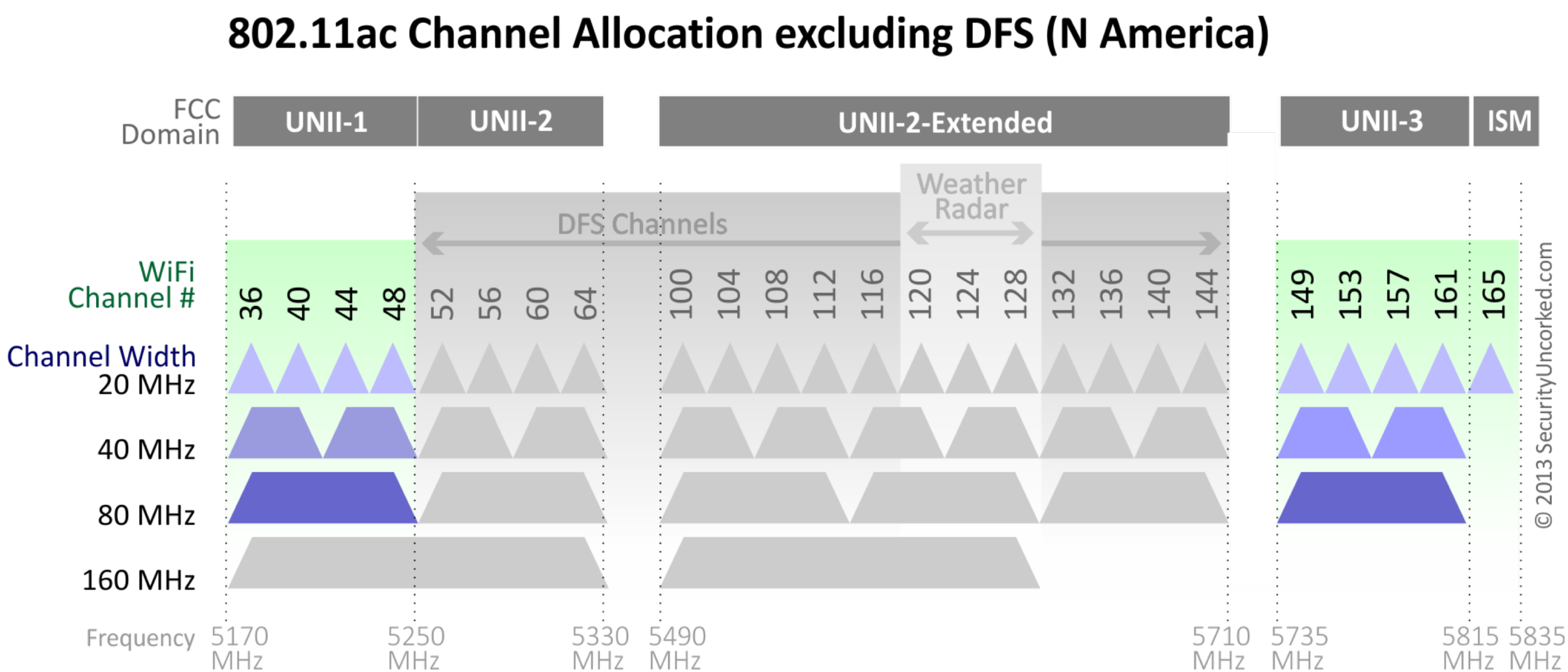
Dynamic Frequency Selection (DFS)

- **Regulatory requirement**: If your wireless device (access point, station, etc..) wants to use certain licensed 5GhZ frequencies, it must listen for <u>**and avoid**</u> interference
  - i.e. Your unlicensed device can only use the frequency in the *absence* of any licensed users
- Licensed users
  - Doppler weather **radar**
  - Civilian aviation **radar**
  - Military **radar**
  - Satellite **radar**

Dynamic Frequency Selection (DFS)

- IEEE 802.11(h) standardized DFS
  - Access point (and clients) specify a Quiet Period in the Beacon frame to silence clients
  - Access point listens in Quiet Period for transmitting radar
  - Radar detected?
    - AP will block further transmissions, broadcast a channel switch announcement, disassociate remaining clients, and randomly select a different channel. (If new channel is DFS-required, AP will scan for radar signals for 60 seconds before enabling beacons and accepting client associations)
- Suggestion: Avoid DFS channels (or at least verify radar interference is non-issue in your location)

802.11 5GHz Channels



http://securityuncorked.com/2013/11/the-best-damn-802-11ac-channel-allocation-graphics/

802.11 5GHz Channels

802.11ac Channel Availability (N America)

Channel Width	Number of channels available	
	Using DFS	DFS Excluded
40 MHz	9-10*	4
80 MHz	4-5*	2
160 MHz	1	0

*Channels 116 and 132 are Doppler Radar channels that may be used in some cases.

http://securityuncorked.com/2013/11/the-best-damn-802-11ac-channel-allocation-graphics/

802.11 Physical Layer Standards

<table>
<tr><th rowspan="2">802.11 Protocol</th><th rowspan="2">Release date</th><th>Frequency</th><th>Bandwidth</th><th>Stream data rate</th><th rowspan="2">Allowable MIMO streams</th><th rowspan="2">Modulation</th><th colspan="2">Approximate range</th></tr>
<tr><th>(GHz)</th><th>(MHz)</th><th>(Mbit/s)</th><th>Indoor</th><th>Outdoor</th></tr>
<tr><td rowspan="2">a</td><td rowspan="2">Sep 1999</td><td>5</td><td rowspan="2">20</td><td rowspan="2">6, 9, 12, 18, 24, 36, 48, 54</td><td rowspan="2">N/A</td><td rowspan="2">OFDM</td><td>35 m (115 ft)</td><td>120 m (390 ft)</td></tr>
<tr><td>3.7</td><td></td><td>5,000 m (16,000 ft)</td></tr>
<tr><td>b</td><td>Sep 1999</td><td>2.4</td><td>22</td><td>1, 2, 5.5, 11</td><td>N/A</td><td>DSSS</td><td>35 m (115 ft)</td><td>140 m (460 ft)</td></tr>
<tr><td>g</td><td>Jun 2003</td><td>2.4</td><td>20</td><td>6, 9, 12, 18, 24, 36, 48, 54</td><td>N/A</td><td>OFDM</td><td>38 m (125 ft)</td><td>140 m (460 ft)</td></tr>
<tr><td rowspan="2">n</td><td rowspan="2">Oct 2009</td><td rowspan="2">2.4/5</td><td>20</td><td>Up to 288.8</td><td rowspan="2">4</td><td rowspan="2">MIMO-OFDM</td><td rowspan="2">70 m (230 ft)</td><td rowspan="2">250 m (820 ft)</td></tr>
<tr><td>40</td><td>Up to 600</td></tr>
<tr><td rowspan="4">ac</td><td rowspan="4">Dec 2013</td><td rowspan="4">5</td><td>20</td><td>Up to 346.8</td><td rowspan="4">8</td><td rowspan="4">MIMO-OFDM</td><td rowspan="4">35 m (115 ft)</td><td rowspan="4"></td></tr>
<tr><td>40</td><td>Up to 800</td></tr>
<tr><td>80</td><td>Up to 1733.2</td></tr>
<tr><td>160</td><td>Up to 3466.8</td></tr>
</table>

https://en.wikipedia.org/wiki/IEEE_802.11

Bandwidth

- Tradeoffs
  - Smaller bandwidth (e.g. 20MHz)
    - Lower data rate
    - Lower risk of interference from APs on neighboring channels
  - Larger bandwidth (e.g. 40, 80MHz)
    - Higher data rate
    - Higher risk of interference from APs on neighboring channels
- Higher bandwidth channels (80MHz, 160MHz) difficult to use in enterprise settings due to interference

802.11 Physical Layer Standards

802.11 Protocol	Release date	Frequency (GHz)	Bandwidth (MHz)	Stream data rate (Mbit/s)	Allowable MIMO streams	Modulation	Approximate range: Indoor	Approximate range: Outdoor
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	35 m (115 ft)	120 m (390 ft)
		3.7						5,000 m (16,000 ft)
b	Sep 1999	2.4	22	1, 2, 5.5, **11**	N/A	DSSS	35 m (115 ft)	140 m (460 ft)
g	Jun 2003	2.4	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	38 m (125 ft)	140 m (460 ft)
n	Oct 2009	2.4/5	20	Up to **288.8**	4	MIMO-OFDM	70 m (230 ft)	250 m (820 ft)
			40	Up to 600				
ac	Dec 2013	5	20	Up to **346.8**	8	MIMO-OFDM	35 m (115 ft)	
			40	Up to 800				
			80	Up to 1733.2				
			160	Up to 3466.8				

https://en.wikipedia.org/wiki/IEEE_802.11

Stream Data Rate

- Manufactures advertise gross bit rate at Layer 2
  - Megabits per second, inclusive of all signaling and control overhead, and with zero interference

802.11a	802.11b	802.11g	802.11n	802.11ac
54	11	54	288	346

- Rates in practice vary widely
  - Distance, obstructions, interference?
- *Customers care about application-layer throughput*

802.11 Physical Layer Standards

802.11 Protocol	Release date	Frequency (GHz)	Bandwidth (MHz)	Stream data rate (Mbit/s)	Allowable MIMO streams	Modulation	Approximate range Indoor	Approximate range Outdoor
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	35 m (115 ft)	120 m (390 ft)
		3.7						5,000 m (16,000 ft)
b	Sep 1999	2.4	22	1, 2, 5.5, **11**	N/A	DSSS	35 m (115 ft)	140 m (460 ft)
g	Jun 2003	2.4	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	38 m (125 ft)	140 m (460 ft)
n	Oct 2009	2.4/5	20	Up to **288.8**	4	MIMO-OFDM	70 m (230 ft)	250 m (820 ft)
			40	Up to 600				
ac	Dec 2013	5	20	Up to **346.8**	8	MIMO-OFDM	35 m (115 ft)	
			40	Up to 800				
			80	Up to 1733.2				
			160	Up to 3466.8				

https://en.wikipedia.org/wiki/IEEE_802.11

MIMO

- Transmitting/Receiving more than one data signal simultaneously over the same radio channel by exploiting multipath propagation
- Added in 802.11n
- Specification: **NxM** system
  - **N** = Number of transmitter antennas
  - **M** = Number of receiver antennas

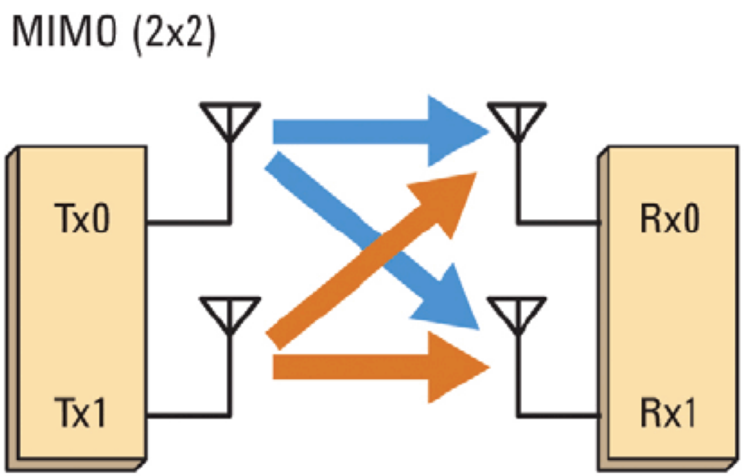
MIMO

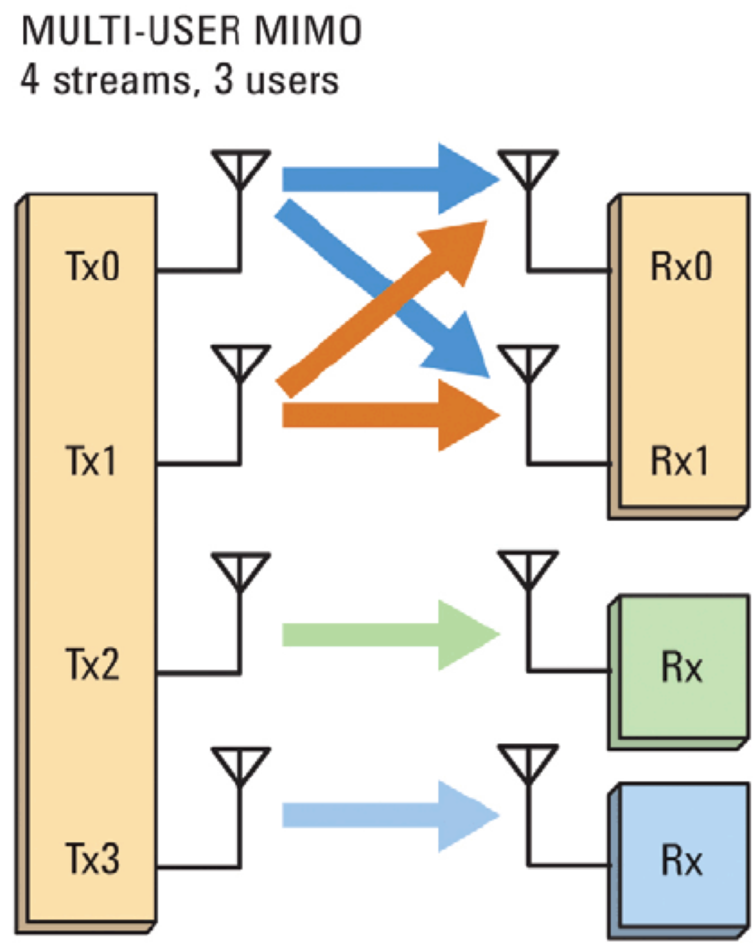
- **Spatial multiplexing**
  - High data rate signal is divided into multiple lower-rate streams
  - Each stream is transmitted from a different transmit antenna, but in the same frequency channel
  - Receiver (with multiple antennas) can separate these streams and reassemble original signal
- Limited by number of antennas at transmitter or receiver

MIMO

- 2x2 : 1 stream
  - 2 transmitting antennas
  - 2 receiving antennas
  - 1 stream of data
- 2x2 : 2 streams
  - 2 transmitting antennas
  - 2 receiving antennas
  - 2 streams of data
- Cannot have 2x2 : 3 (number of streams exceeds number of antennas)

MIMO, MU-MIMO


MIMO (2x2)
Tx0
Tx1
Rx0
Rx1


MULTI-USER MIMO
4 streams, 3 users
Tx0
Tx1
Tx2
Tx3
Rx0
Rx1
Rx
Rx

802.11 Physical Layer Standards

802.11 Protocol	Release date	Frequency (GHz)	Bandwidth (MHz)	Stream data rate (Mbit/s)	Allowable MIMO streams	Modulation	Approximate range: Indoor	Approximate range: Outdoor
a	Sep 1999	5	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	35 m (115 ft)	120 m (390 ft)
		3.7						5,000 m (16,000 ft)
b	Sep 1999	2.4	22	1, 2, 5.5, **11**	N/A	DSSS	35 m (115 ft)	140 m (460 ft)
g	Jun 2003	2.4	20	6, 9, 12, 18, 24, 36, 48, **54**	N/A	OFDM	38 m (125 ft)	140 m (460 ft)
n	Oct 2009	2.4/5	20	Up to **288.8**	4	MIMO-OFDM	70 m (230 ft)	250 m (820 ft)
			40	Up to 600				
ac	Dec 2013	5	20	Up to **346.8**	8	MIMO-OFDM	35 m (115 ft)	
			40	Up to 800				
			80	Up to 1733.2				
			160	Up to 3466.8				

https://en.wikipedia.org/wiki/IEEE_802.11

Modulation

- DSSS
  - Direct Sequence Spread Spectrum
- OFDM
  - Orthogonal Frequency-Division Multiplexing
- MIMO-OFDM
  - Multiple Input Multiple Output Orthogonal Frequency-Division Multiplexing

Modulation and Coding Scheme (MCS)

MCS index	Spatial Streams	Modulation type	Coding rate	Data rate (in Mbit/s)							
				20 MHz channels		40 MHz channels		80 MHz channels		160 MHz channels	
				800 ns GI	400 ns GI	800 ns GI	400 ns GI	800 ns GI	400 ns GI	800 ns GI	400 ns GI
0	1	BPSK	1/2	6.5	7.2	13.5	15	29.3	32.5	58.5	65
1	1	QPSK	1/2	13	14.4	27	30	58.5	65	117	130
2	1	QPSK	3/4	19.5	21.7	40.5	45	87.8	97.5	175.5	195
3	1	16-QAM	1/2	26	28.9	54	60	117	130	234	260
4	1	16-QAM	3/4	39	43.3	81	90	175.5	195	351	390
5	1	64-QAM	2/3	52	57.8	108	120	234	260	468	520
6	1	64-QAM	3/4	58.5	65	121.5	135	263.3	292.5	526.5	585
7	1	64-QAM	5/6	65	72.2	135	150	292.5	325	585	650
8	1	256-QAM	3/4	78	86.7	162	180	351	390	702	780
9	1	256-QAM	5/6	N/A	N/A	180	200	390	433.3	780	866.7

Added in 802.11n and 802.11ac instead of specifying data rates

Index continues incrementing for multiple spatial streams

https://en.wikipedia.org/wiki/IEEE_802.11ac

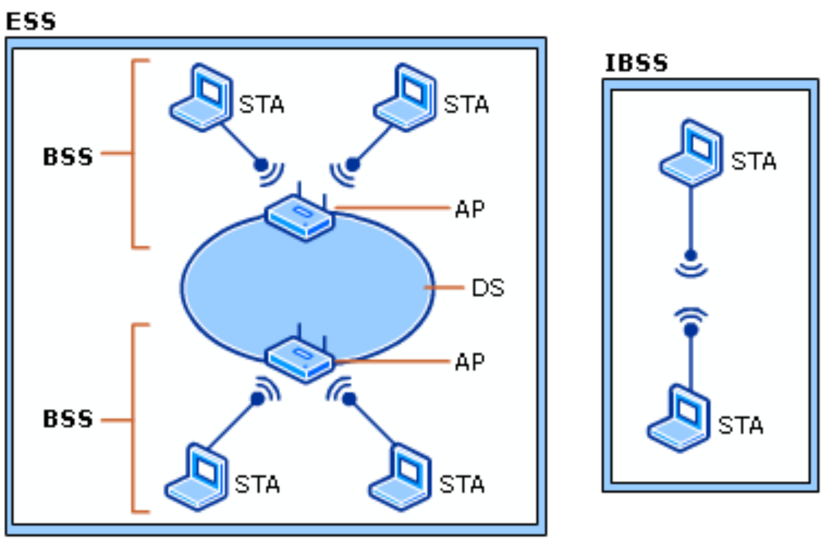
Modulation and Coding Scheme (MCS)

- Modulation type
  - More complex modulation = Higher data rate
  - More complex modulations require better conditions (less interference, line of sight, …)
- Coding rate
  - How much of the data stream is used to transmit payload data (as opposed to encodings)
  - Most efficient rate is 5/6 or 83.3% of the data stream being used
- Guard Interval (GI)
  - Short pause between packet transmission to allow for any false information to be ignored
  - Longer Guard Intervals increase reliability

IEEE 802.11 Link Layer

Link Layer Terminology

- Station (**STA**)
  - Laptop, desktop, phone (and access point)
- Access Point (**AP**)
- Basic Service Set (**BSS**)
  - Set of stations controlled by common coordination function (decides who can transmit)
- Distribution System (**DS**)
  - Connects BSS and LANs together to form ESS
- Extended Service Set (**ESS**)



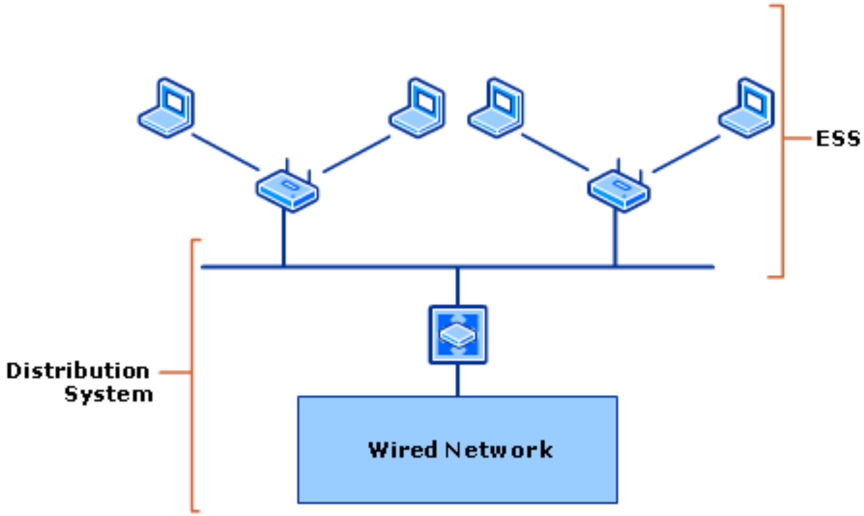
- Independent Basic Service Set (**IBSS**)
  - Ad-hoc network (no AP)

https://technet.microsoft.com/en-us/library/cc757419%28v=ws.10%29.aspx

Link Layer Terminology

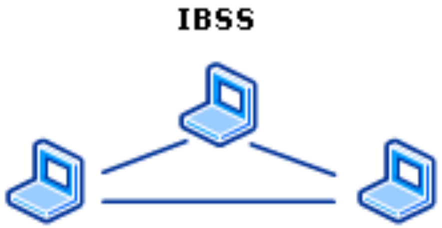
Infrastructure Mode

- One client (station) + One AP



Ad-Hoc Mode

- Clients (stations) communicate directly with each other (no AP)



802.11 MAC Frame

2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	6 bytes	6 bytes	0-2312 bytes	4 bytes
Frame Control	**Duration**	**Addr 1**	**Addr 2**	**Addr 3**	**Sequence Ctrl**	**Addr 4**	**Frame Body**	**FCS**

- Frame Control *(Bitfield)*
  - Protocol Version
  - Type/Subtype
  - To DS / From DS (Distribution System, i.e. LAN)
    - From STA to DS via an AP
    - From DS to STA via AP
    - *Determines meaning of all the address fields!*
  - More Fragments
  - Power Management
  - Retry (in case ACK was not received)
  - Protected (encrypted)
  - …

802.11 MAC Frame

2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	6 bytes	6 bytes	0-2312 bytes	4 bytes
Frame Control	**Duration**	**Addr 1**	**Addr 2**	**Addr 3**	**Sequence Ctrl**	**Addr 4**	**Frame Body**	**FCS**

- Duration
  - Duration needed to receive <u>next</u> frame transmission in *microseconds*
  - i.e. Everyone else should stay quiet for this time!
- Field can also be *association ID*

802.11 MAC Frame

2 bytes	*2 bytes*	*6 bytes*	*6 bytes*	*6 bytes*	*6 bytes*	*6 bytes*	*0-2312 bytes*	*4 bytes*
Frame Control	**Duration**	**Addr 1**	**Addr 2**	**Addr 3**	**Sequence Ctrl**	**Addr 4**	**Frame Body**	**FCS**

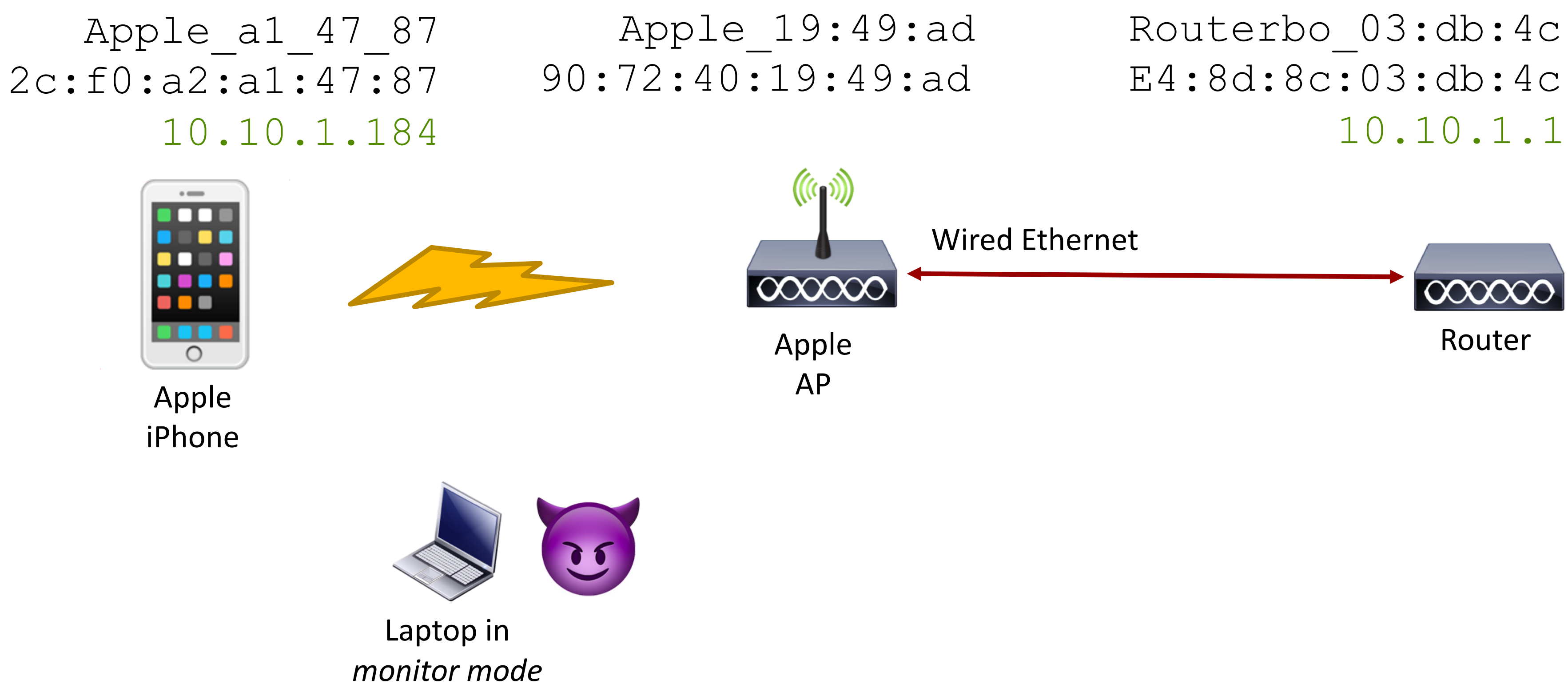
- 4 MAC address fields – will have some combination of:
  - Destination Address (DA) – Final destination to receive frame
  - Source Address (SA) – Original source that created and transmitted frame
  - Receiver Address (RA) – Address of next station on wireless medium to receive frame
  - Transmitter Address (TA) – MAC address of station that transmitted frame onto wireless medium
  - Basic Service Set Identifier (BSSID)
    - In infrastructure mode, BSSID is MAC address of access point

802.11 MAC Frame

2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	6 bytes	6 bytes	0-2312 bytes	4 bytes
Frame Control	**Duration**	**Addr 1**	**Addr 2**	**Addr 3**	**Sequence Ctrl**	**Addr 4**	**Frame Body**	**FCS**

- Frame Body = Payload
- FCS = Frame Check Sequence
  - Cyclic Redundancy Check (CRC) over all fields in MAC header and frame body

Example



Apple_a1_47_87
2c:f0:a2:a1:47:87
10.10.1.184
Apple_19:49:ad
90:72:40:19:49:ad
Routerbo_03:db:4c
E4:8d:8c:03:db:4c
10.10.1.1
Apple iPhone
Apple AP
Wired Ethernet
Router
Laptop in *monitor mode*

wireshark_iphone_2.pcapng

Apply a display filter … <⌘/>   Expression…   +

No.	Time	Source	Destination	Protocol	Length	Info
1030	17.72…	Apple_a1:47:87 …	Apple_19:49:ad …	802.11	52	Request-to-send, Flags=........C
1031	17.72…		Apple_a1:47:87 …	802.11	46	Clear-to-send, Flags=........C
1032	17.72…	10.10.1.184	8.8.8.8	ICMP	170	Echo (ping) request id=0x9a06, seq=4/10…
1033	17.72…	Apple_19:49:ad …	Apple_a1:47:87 …	802.11	64	802.11 Block Ack, Flags=........C

Ping from iPhone to Google

```
Frame 1032: 170 bytes on wire (1360 bits), 170 bytes captured (1360 bits) on interface 0
PPI version 0, 32 bytes
802.11 radio information
IEEE 802.11 QoS Data, Flags: .p.....TC
  Type/Subtype: QoS Data (0x0028)
  Frame Control Field: 0x8841
  .000 0000 0011 0000 = Duration: 48 microseconds
  Receiver address: Apple_19:49:ad (90:72:40:19:49:ad)
  Destination address: Routerbo_03:db:4c (e4:8d:8c:03:db:4c)
  Transmitter address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
  Source address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
  BSS Id: Apple_19:49:ad (90:72:40:19:49:ad)
  STA address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
  .... .... .... 0000 = Fragment number: 0
  0000 0100 1111 .... = Sequence number: 79
  Frame check sequence: 0x2f0c6948 [correct]
  [FCS Status: Good]
  Qos Control: 0x0000
  CCMP parameters
Logical-Link Control
  DSAP: SNAP (0xaa)
  SSAP: SNAP (0xaa)
```

(1) Receiver Addr (RA): Access Point

(3) Destination Addr (DA): Router

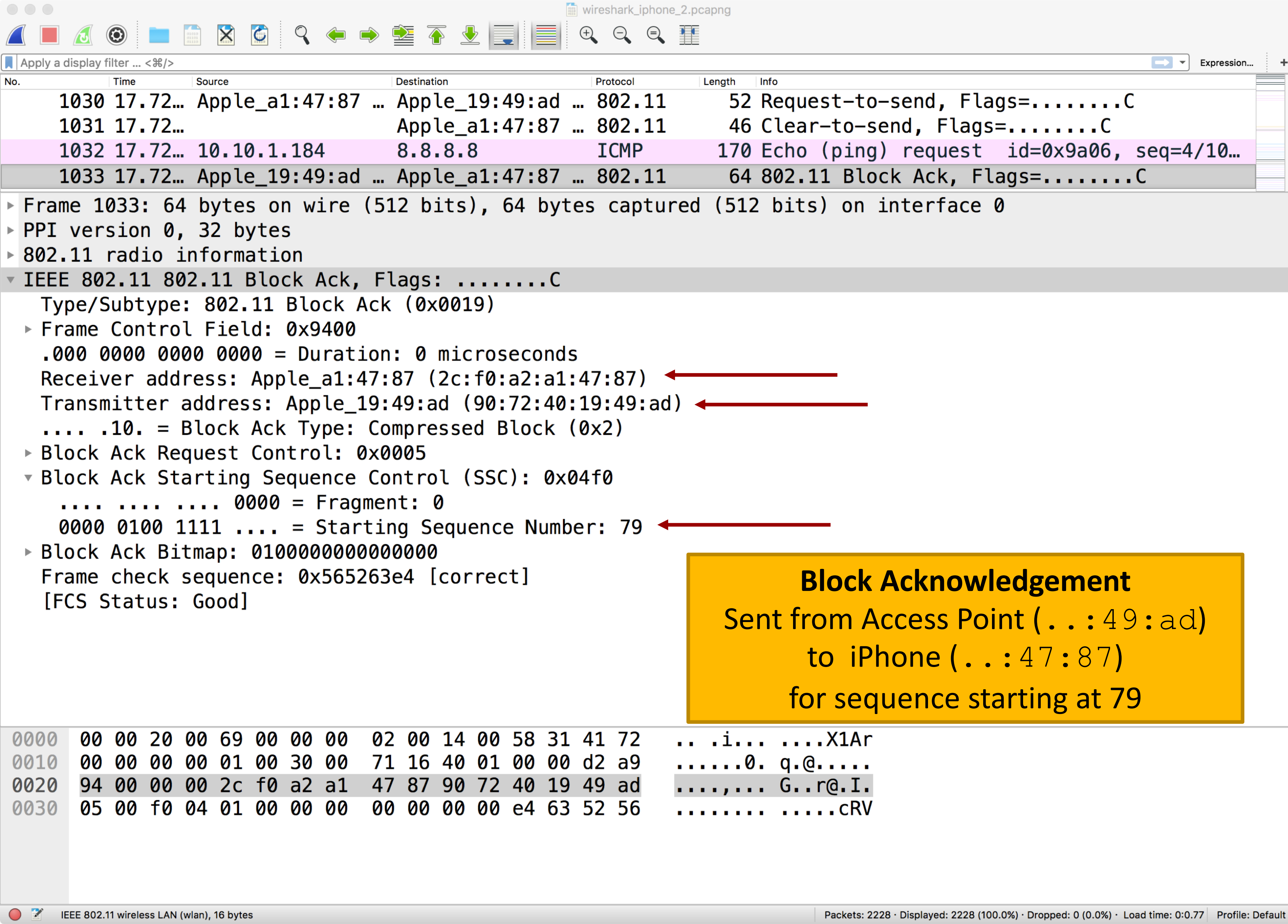
(2) Source Addr (RA): iPhone

(Wireshark labels same field with two names)

```
0020  88 41 30 00 90 72 40 19  49 ad 2c f0 a2 a1 47 87   .A0..r@. I.,...G.
0030  e4 8d 8c 03 db 4c f0 04  00 00 82 00 00 20 00 00   .....L.. ..... ..
0040  00 00 8c 26 fc fb d5 60  1b 4f 6e 24 bf 0d 52 ff   ...&...` .On$..R.
0050  cd 7d 4f 12 c4 cd 51 81  f2 68 9c c7 ee 7d bb c5   .}O...Q. .h...}..
0060  80 20 fd 70 93 06 c8 67  c8 dd 4c 58 25 aa a0 82   . .p...g ..LX%...
0070  06 06 60 8f 09 44 fa 2f  6a 87 f6 40 d5 4e 6f 35   ..`..D./ j..@.No5
```

Frame (170 bytes)   Decrypted CCMP data (92 bytes)

IEEE 802.11 wireless LAN (wlan), 34 bytes   Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.38   Profile: Default



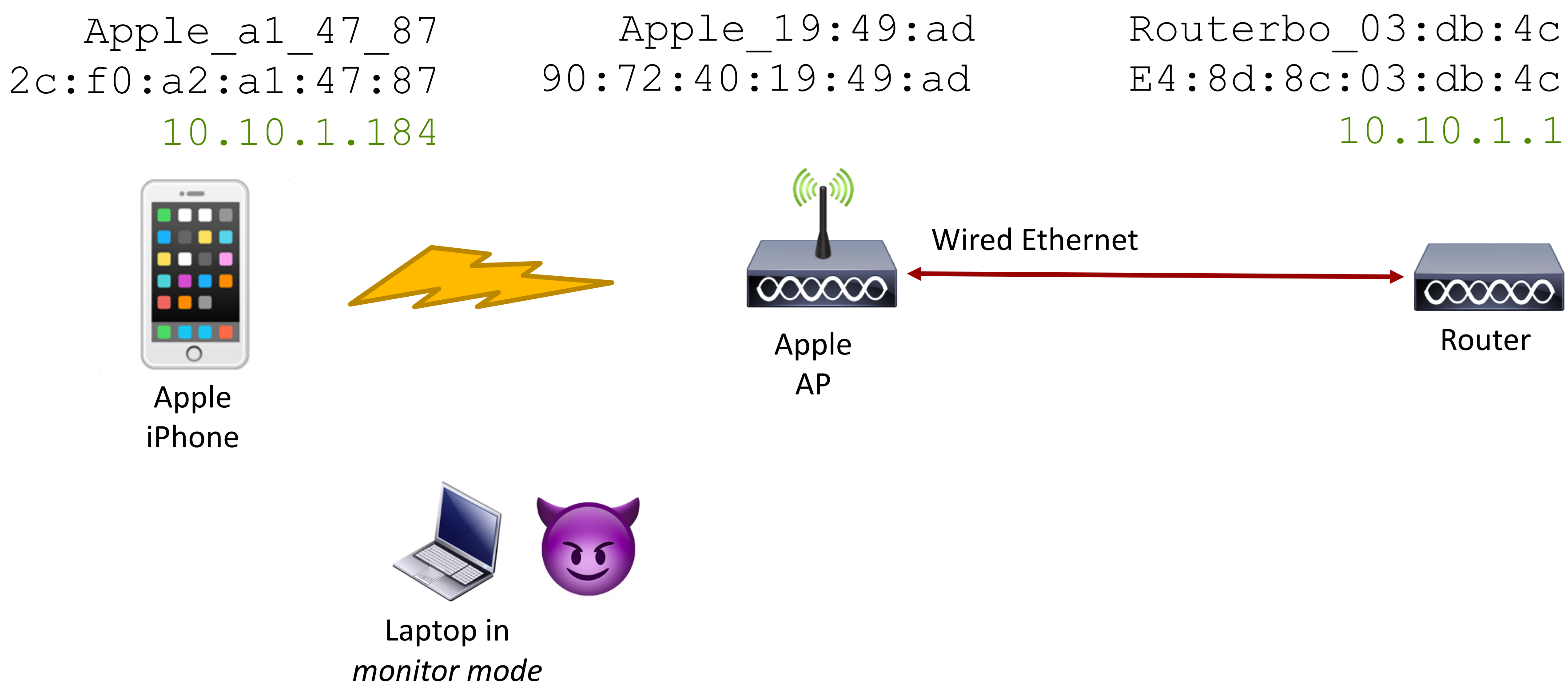
Block Acknowledgement

Sent from Access Point (..:49:ad)

to iPhone (..:47:87)

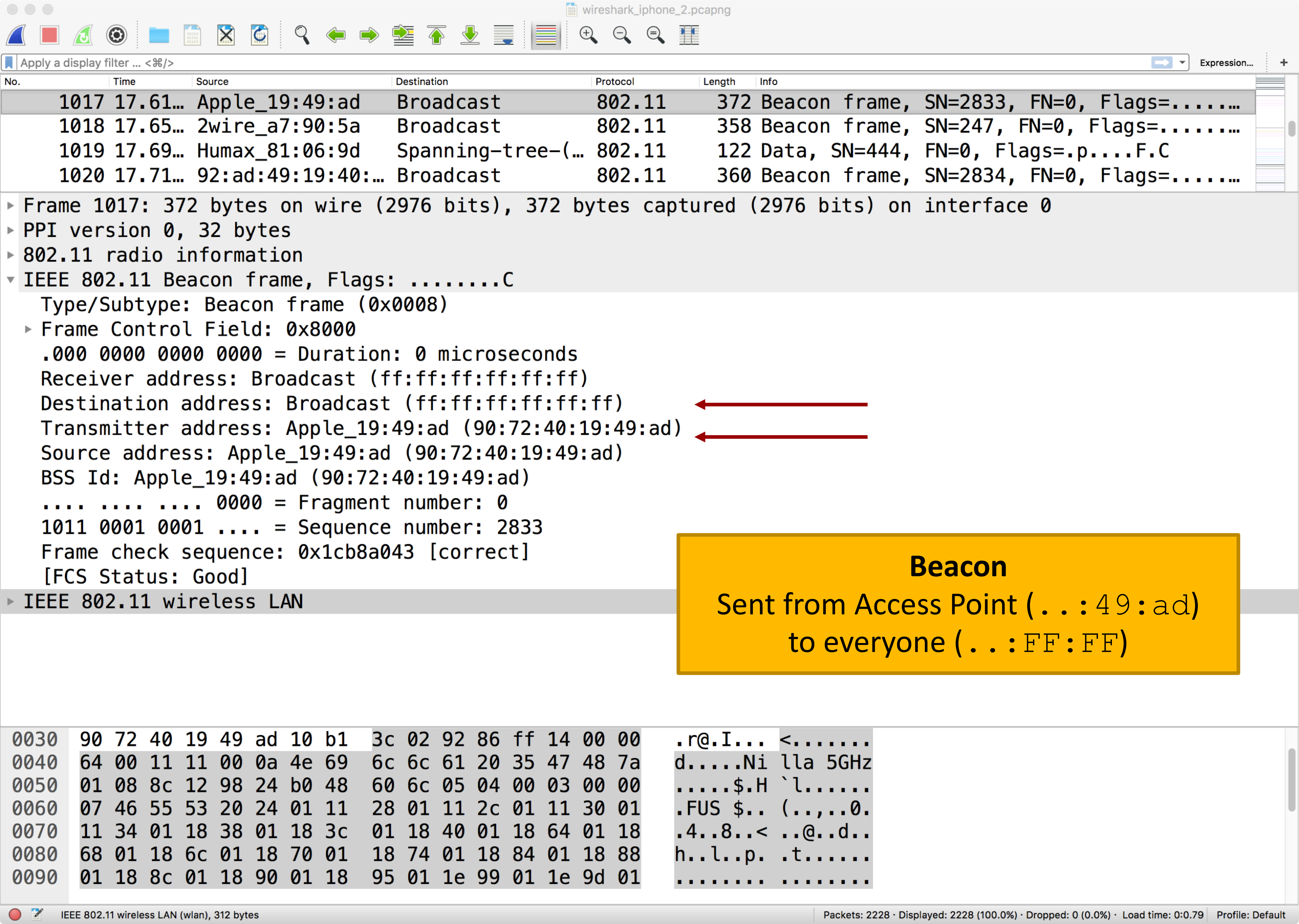
for sequence starting at 79

(Same) Example



Beacons and Probes

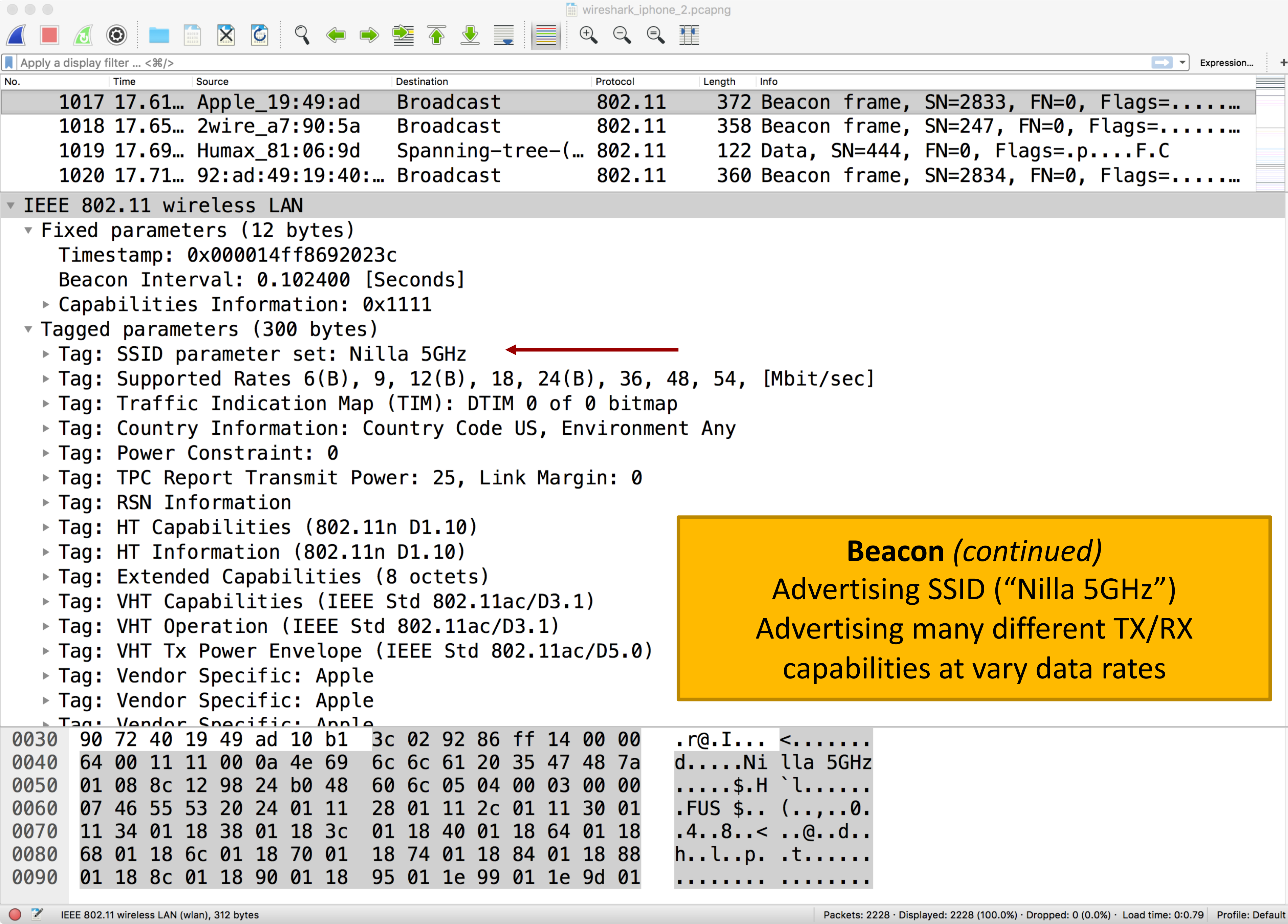
- **Beacon** Frames
  - Broadcast periodically by APs
  - Contains SSID, AP address, Beacon Frame interval, supported data rates, other capabilities
- **Probe Request** Frames
  - Stations can solicit information from APs instead of waiting for beacon
  - Reply from AP sent in **Probe Response** frames



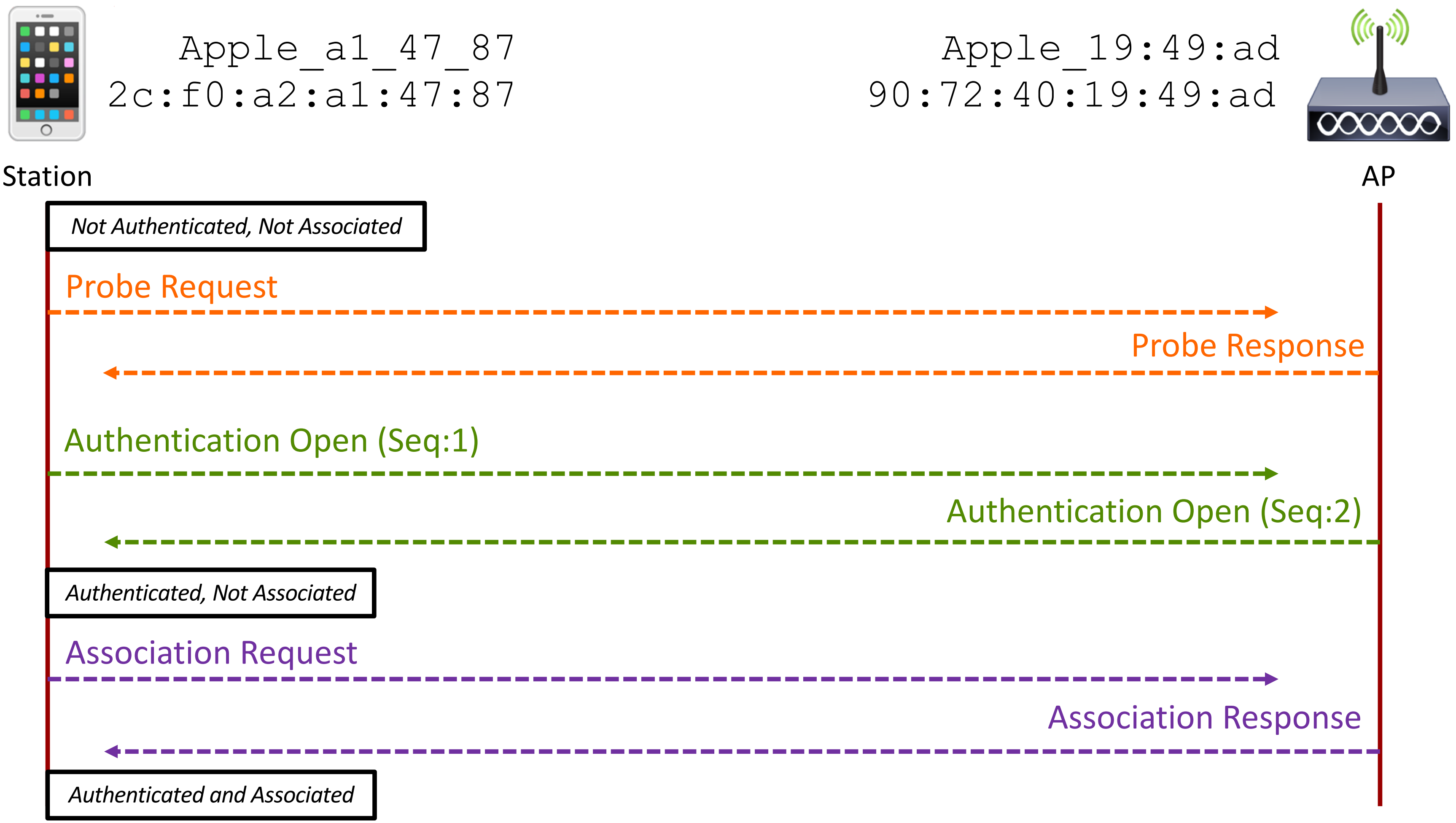
Beacon

Sent from Access Point (..:`49:ad`)

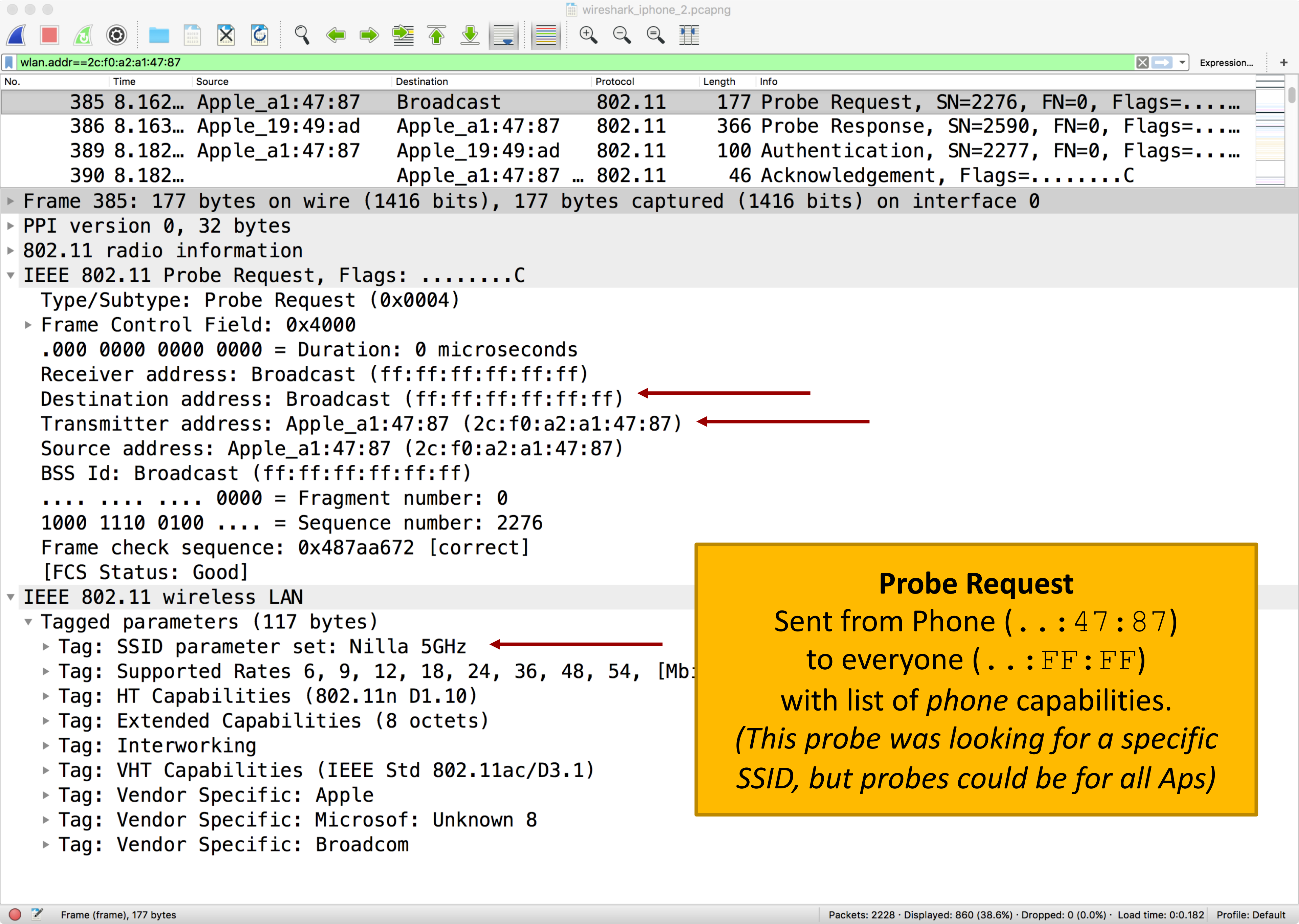
to everyone (..:`FF:FF`)


wireshark_iphone_2.pcapng
Apply a display filter ... <⌘/>
Expression...
No. Time Source Destination Protocol Length Info
1017 17.61… Apple_19:49:ad Broadcast 802.11 372 Beacon frame, SN=2833, FN=0, Flags=........
1018 17.65… 2wire_a7:90:5a Broadcast 802.11 358 Beacon frame, SN=247, FN=0, Flags=........
1019 17.69… Humax_81:06:9d Spanning-tree-(… 802.11 122 Data, SN=444, FN=0, Flags=.p....F.C
1020 17.71… 92:ad:49:19:40:… Broadcast 802.11 360 Beacon frame, SN=2834, FN=0, Flags=........
IEEE 802.11 wireless LAN
Fixed parameters (12 bytes)
Timestamp: 0x000014ff8692023c
Beacon Interval: 0.102400 [Seconds]
Capabilities Information: 0x1111
Tagged parameters (300 bytes)
Tag: SSID parameter set: Nilla 5GHz
Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48, 54, [Mbit/sec]
Tag: Traffic Indication Map (TIM): DTIM 0 of 0 bitmap
Tag: Country Information: Country Code US, Environment Any
Tag: Power Constraint: 0
Tag: TPC Report Transmit Power: 25, Link Margin: 0
Tag: RSN Information
Tag: HT Capabilities (802.11n D1.10)
Tag: HT Information (802.11n D1.10)
Tag: Extended Capabilities (8 octets)
Tag: VHT Capabilities (IEEE Std 802.11ac/D3.1)
Tag: VHT Operation (IEEE Std 802.11ac/D3.1)
Tag: VHT Tx Power Envelope (IEEE Std 802.11ac/D5.0)
Tag: Vendor Specific: Apple
Tag: Vendor Specific: Apple
0030 90 72 40 19 49 ad 10 b1 3c 02 92 86 ff 14 00 00 .r@.I... <.......
0040 64 00 11 11 00 0a 4e 69 6c 6c 61 20 35 47 48 7a d.....Ni lla 5GHz
0050 01 08 8c 12 98 24 b0 48 60 6c 05 04 00 03 00 00 .....$.H `l......
0060 07 46 55 53 20 24 01 11 28 01 11 2c 01 11 30 01 .FUS $.. (..,...0.
0070 11 34 01 18 38 01 18 3c 01 18 40 01 18 64 01 18 .4..8..< ..@..d..
0080 68 01 18 6c 01 18 70 01 18 74 01 18 84 01 18 88 h..l..p. .t......
0090 01 18 8c 01 18 90 01 18 95 01 1e 99 01 1e 9d 01 ....
IEEE 802.11 wireless LAN (wlan), 312 bytes
Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.79
Profile: Default
Beacon *(continued)*
Advertising SSID (“Nilla 5GHz”)
Advertising many different TX/RX
capabilities at vary data rates

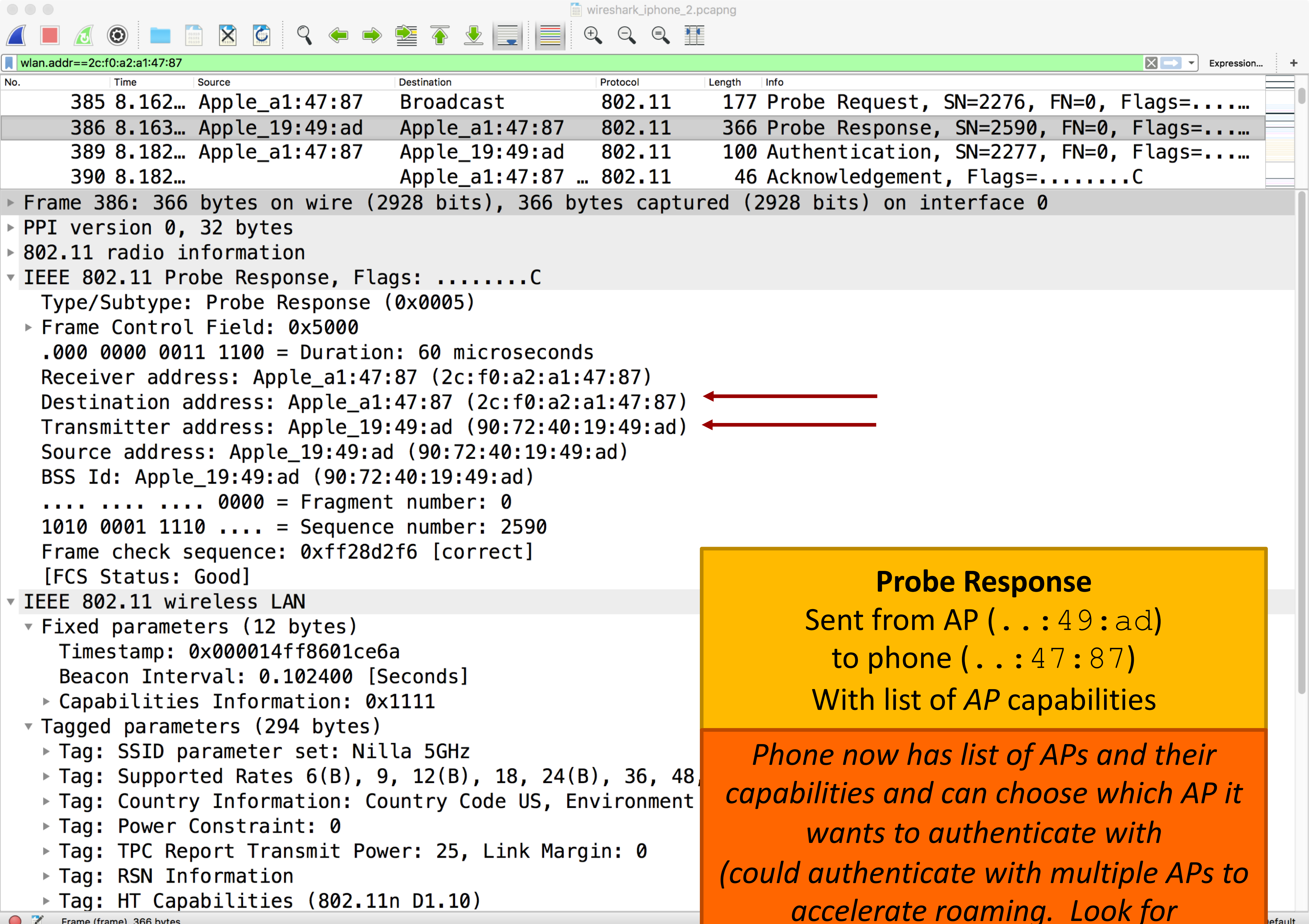
Association



https://documentation.meraki.com/MR/WiFi_Basics_and_Best_Practices/802.11_Association_process_explained



Probe Request
Sent from Phone (`..:47:87`)
to everyone (`..:FF:FF`)
with list of *phone* capabilities.
(This probe was looking for a specific SSID, but probes could be for all Aps)



Probe Response
Sent from AP (..:49:ad)
to phone (..:47:87)
With list of *AP* capabilities

Phone now has list of APs and their capabilities and can choose which AP it wants to authenticate with (could authenticate with multiple APs to accelerate roaming. Look for strongest?)

wireshark_iphone_2.pcapng

wlan.addr==2c:f0:a2:a1:47:87

Expression... +

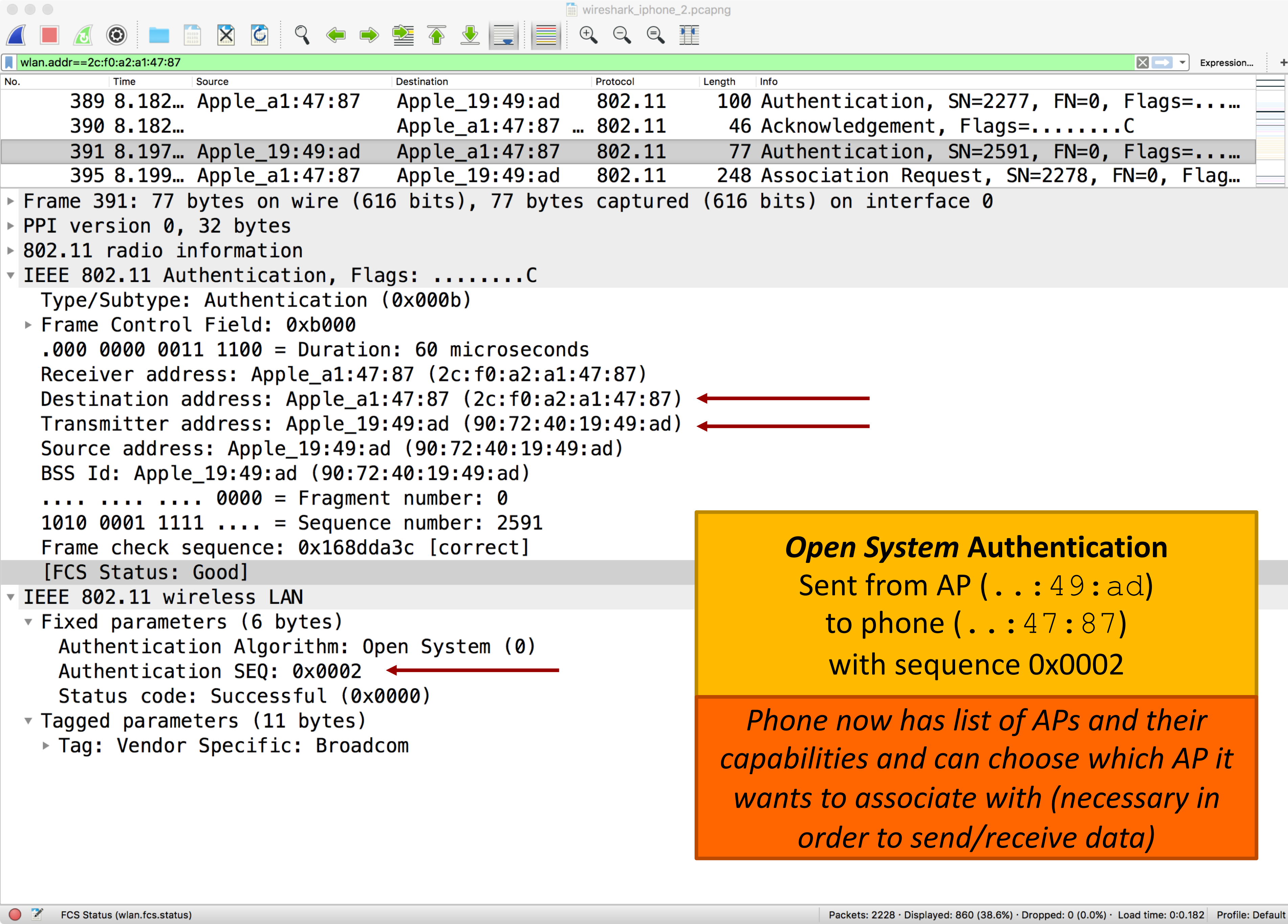
No.	Time	Source	Destination	Protocol	Length	Info
385	8.162…	Apple_a1:47:87	Broadcast	802.11	177	Probe Request, SN=2276, FN=0, Flags=.....…
386	8.163…	Apple_19:49:ad	Apple_a1:47:87	802.11	366	Probe Response, SN=2590, FN=0, Flags=....…
389	8.182…	Apple_a1:47:87	Apple_19:49:ad	802.11	100	Authentication, SN=2277, FN=0, Flags=....…
390	8.182…		Apple_a1:47:87 …	802.11	46	Acknowledgement, Flags=........C

```
▸ Frame 389: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
▸ PPI version 0, 32 bytes
▸ 802.11 radio information
▾ IEEE 802.11 Authentication, Flags: ........C
    Type/Subtype: Authentication (0x000b)
  ▸ Frame Control Field: 0xb000
    .000 0000 0011 1100 = Duration: 60 microseconds
    Receiver address: Apple_19:49:ad (90:72:40:19:49:ad)
    Destination address: Apple_19:49:ad (90:72:40:19:49:ad)
    Transmitter address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
    Source address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
    BSS Id: Apple_19:49:ad (90:72:40:19:49:ad)
    .... .... .... 0000 = Fragment number: 0
    1000 1110 0101 .... = Sequence number: 2277
    Frame check sequence: 0x4709c80e [correct]
    [FCS Status: Good]
▾ IEEE 802.11 wireless LAN
  ▾ Fixed parameters (6 bytes)
      Authentication Algorithm: Open System (0)
      Authentication SEQ: 0x0001
      Status code: Successful (0x0000)
  ▾ Tagged parameters (34 bytes)
    ▸ Tag: Extended Capabilities (8 octets)
    ▸ Tag: Vendor Specific: Apple
    ▸ Tag: Vendor Specific: Broadcom
```

***Open System* Authentication**

Sent from Phone (`..:47:87`)
to selected AP (`..:49:ad`)
to open authentication with
sequence 0x0001

Frame (frame), 100 bytes | Packets: 2228 · Displayed: 860 (38.6%) · Dropped: 0 (0.0%) · Load time: 0:0.182 | Profile: Default



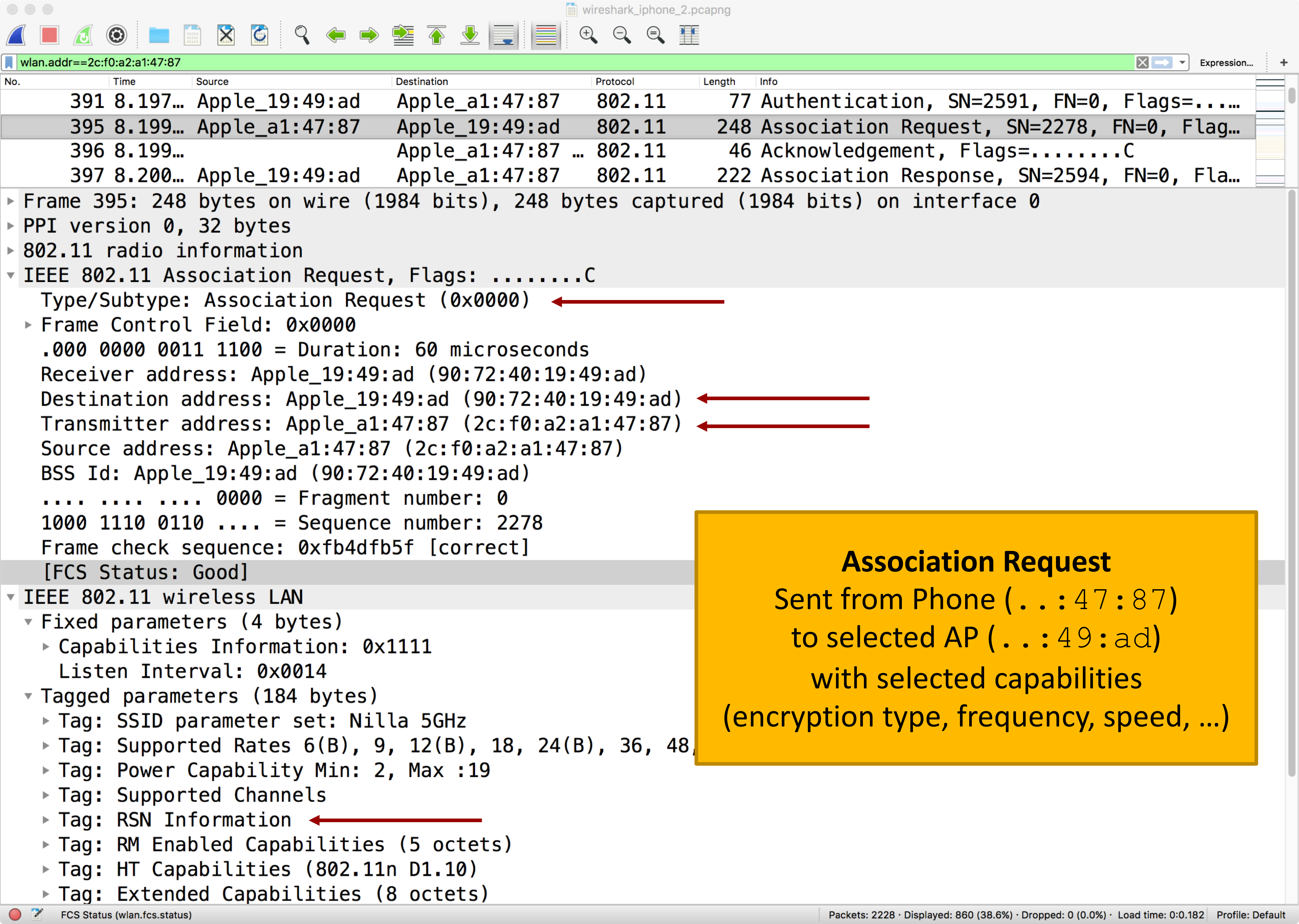
***Open System* Authentication**

Sent from AP (..:49:ad)

to phone (..:47:87)

with sequence 0x0002

Phone now has list of APs and their capabilities and can choose which AP it wants to associate with (necessary in order to send/receive data)



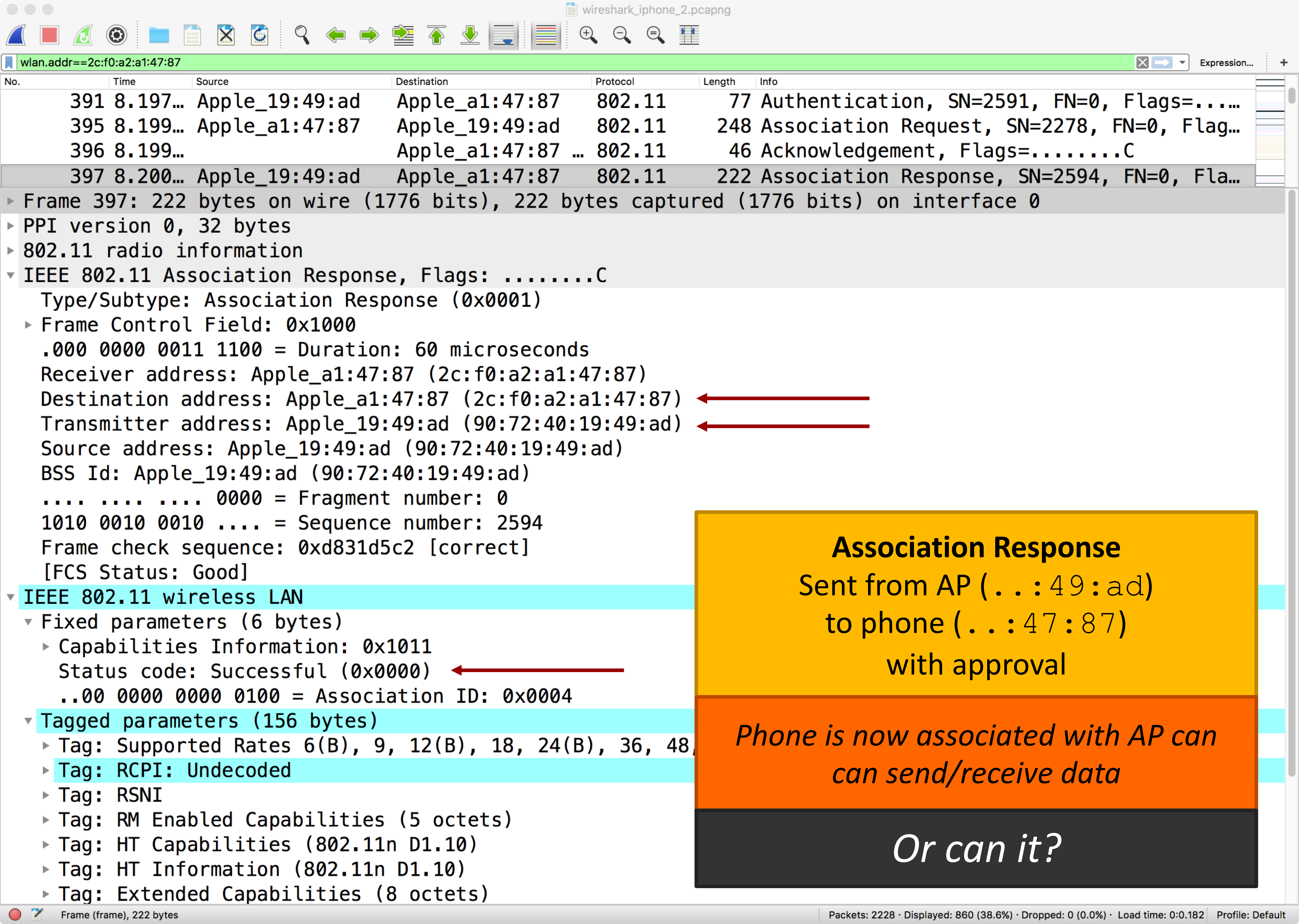
Association Request

Sent from Phone (...:47:87)

to selected AP (...:49:ad)

with selected capabilities

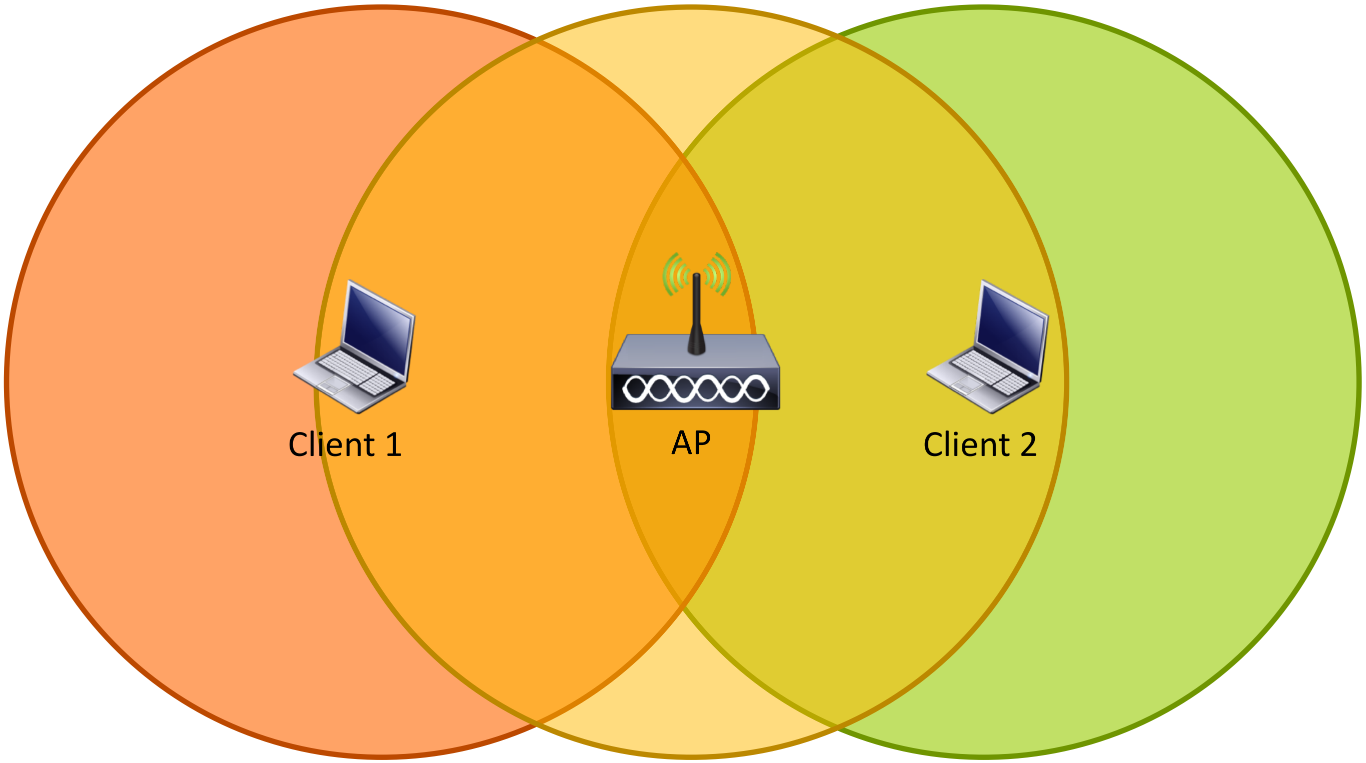
(encryption type, frequency, speed, ...)


wireshark_iphone_2.pcapng
wlan.addr==2c:f0:a2:a1:47:87
Expression…
No. Time Source Destination Protocol Length Info
391 8.197… Apple_19:49:ad Apple_a1:47:87 802.11 77 Authentication, SN=2591, FN=0, Flags=.…
395 8.199… Apple_a1:47:87 Apple_19:49:ad 802.11 248 Association Request, SN=2278, FN=0, Flag…
396 8.199… Apple_a1:47:87 … 802.11 46 Acknowledgement, Flags=........C
397 8.200… Apple_19:49:ad Apple_a1:47:87 802.11 222 Association Response, SN=2594, FN=0, Fla…
Frame 397: 222 bytes on wire (1776 bits), 222 bytes captured (1776 bits) on interface 0
PPI version 0, 32 bytes
802.11 radio information
IEEE 802.11 Association Response, Flags: C
Type/Subtype: Association Response (0x0001)
Frame Control Field: 0x1000
.000 0000 0011 1100 = Duration: 60 microseconds
Receiver address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
Destination address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
Transmitter address: Apple_19:49:ad (90:72:40:19:49:ad)
Source address: Apple_19:49:ad (90:72:40:19:49:ad)
BSS Id: Apple_19:49:ad (90:72:40:19:49:ad)
.... 0000 = Fragment number: 0
1010 0010 0010 .... = Sequence number: 2594
Frame check sequence: 0xd831d5c2 [correct]
[FCS Status: Good]
IEEE 802.11 wireless LAN
Fixed parameters (6 bytes)
Capabilities Information: 0x1011
Status code: Successful (0x0000)
..00 0000 0000 0100 = Association ID: 0x0004
Tagged parameters (156 bytes)
Tag: Supported Rates 6(B), 9, 12(B), 18, 24(B), 36, 48,
Tag: RCPI: Undecoded
Tag: RSNI
Tag: RM Enabled Capabilities (5 octets)
Tag: HT Capabilities (802.11n D1.10)
Tag: HT Information (802.11n D1.10)
Tag: Extended Capabilities (8 octets)
Frame (frame), 222 bytes
Packets: 2228 · Displayed: 860 (38.6%) · Dropped: 0 (0.0%) · Load time: 0:0.182
Profile: Default
Association Response
Sent from AP (..:49:ad)
to phone (..:47:87)
with approval
Phone is now associated with AP can
can send/receive data
Or can it?

Block Acknowledgements

- Phone and AP *negotiate* to enable **block acknowledgement** mode
  - Ability to send one ACK for multiple QoS data blocks
  - Introduced in 802.11e standard
  - Mandated in 802.11n and newer revisions

Hidden Node Problem



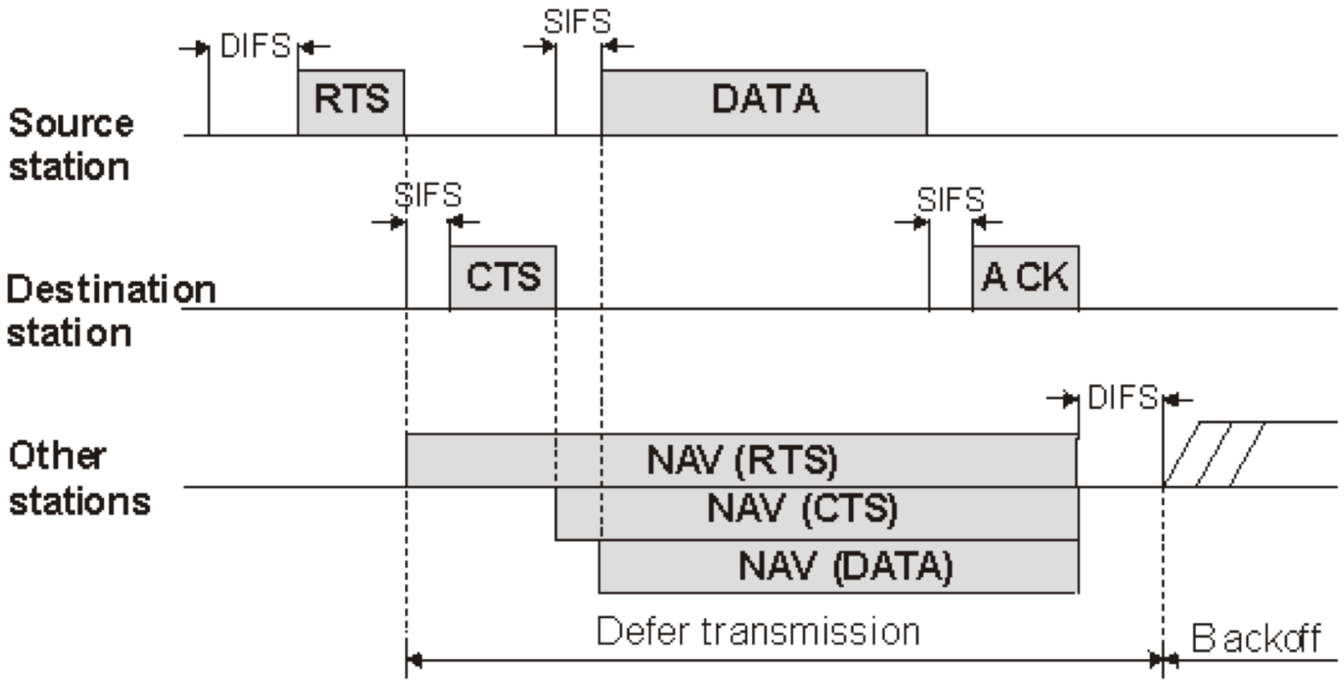
Client 1 <-> AP ✓ AP <-> Client 2 ✓

Client 1 <-> Client 2 🚫

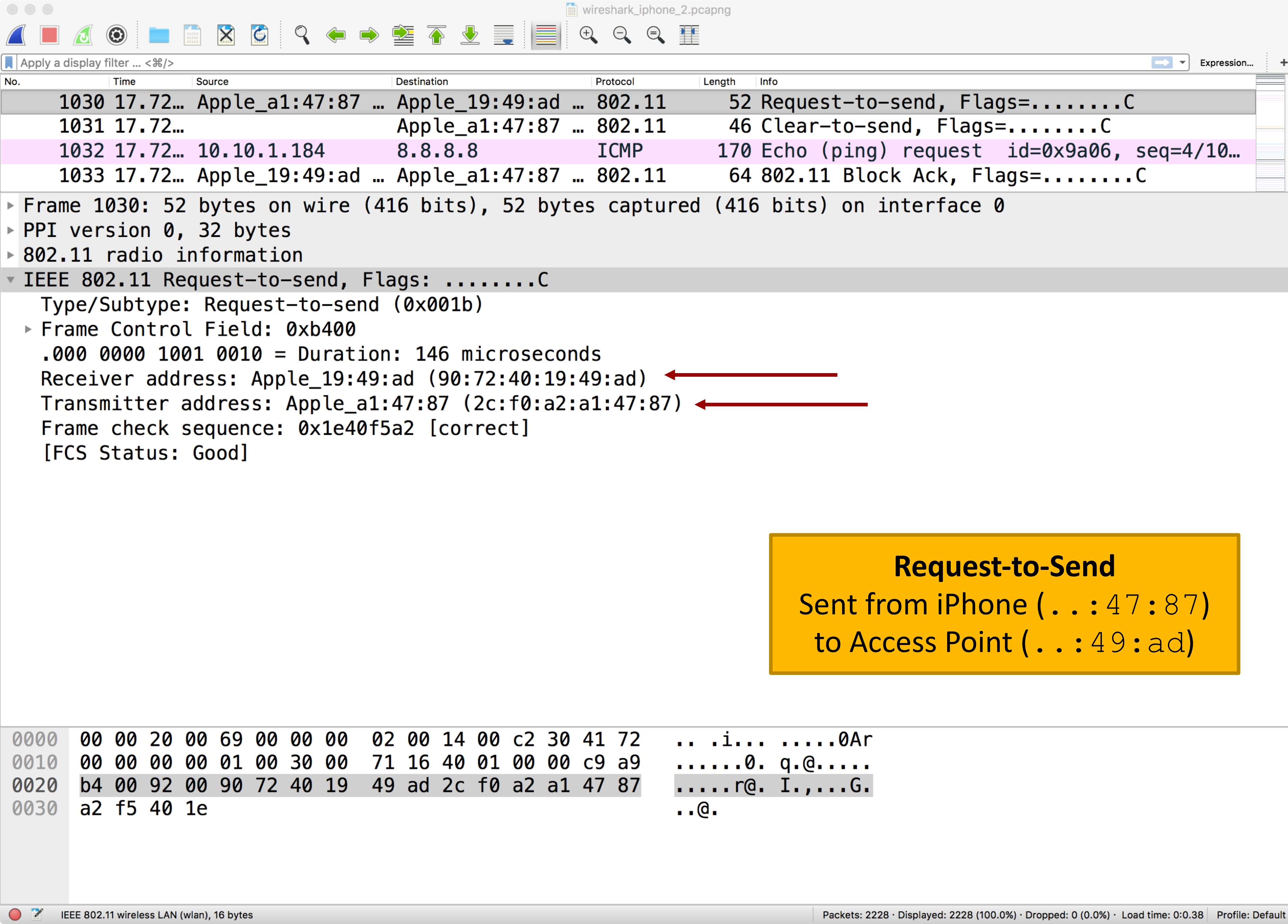
CSMA/CA, RTS/CTS

- CSMA/CA
  - Carrier Sense Multiple Access / Collision Avoidance
  - Listen for other parties transmitting
  - Channel clear? Go ahead and transmit
  - Does not solve hidden node problem
- RTS/CTS
  - Request to Send / Clear to Send

RTS/CTS



- **NAV** = Network Allocation Vector (countdown timer of imposed silence based on RTS/CTS messages that a station has overheard)
- **SIFS** = Short Inter-Frame Space (gap to detect end of frame before transmitting)
- **DIFS** = DCF Inter-Frame Space (CSMA/CA – exponential backoff from collision)


wireshark_iphone_2.pcapng
Apply a display filter … <⌘/>
Expression…
No. Time Source Destination Protocol Length Info
1030 17.72… Apple_a1:47:87 … Apple_19:49:ad … 802.11 52 Request-to-send, Flags=........C
1031 17.72… Apple_a1:47:87 … 802.11 46 Clear-to-send, Flags=........C
1032 17.72… 10.10.1.184 8.8.8.8 ICMP 170 Echo (ping) request id=0x9a06, seq=4/10…
1033 17.72… Apple_19:49:ad … Apple_a1:47:87 … 802.11 64 802.11 Block Ack, Flags=........C
Frame 1030: 52 bytes on wire (416 bits), 52 bytes captured (416 bits) on interface 0
PPI version 0, 32 bytes
802.11 radio information
IEEE 802.11 Request-to-send, Flags: C
Type/Subtype: Request-to-send (0x001b)
Frame Control Field: 0xb400
.000 0000 1001 0010 = Duration: 146 microseconds
Receiver address: Apple_19:49:ad (90:72:40:19:49:ad)
Transmitter address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
Frame check sequence: 0x1e40f5a2 [correct]
[FCS Status: Good]
Request-to-Send
Sent from iPhone (..:47:87)
to Access Point (..:49:ad)
0000 00 00 20 00 69 00 00 00 02 00 14 00 c2 30 41 72 .. .i... .....0Ar
0010 00 00 00 00 01 00 30 00 71 16 40 01 00 00 c9 a9 ......0. q.@.....
0020 b4 00 92 00 90 72 40 19 49 ad 2c f0 a2 a1 47 87 .....r@. I.,...G.
0030 a2 f5 40 1e ..@.
IEEE 802.11 wireless LAN (wlan), 16 bytes
Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.38
Profile: Default

wireshark_iphone_2.pcapng

Apply a display filter … <⌘/>   Expression…

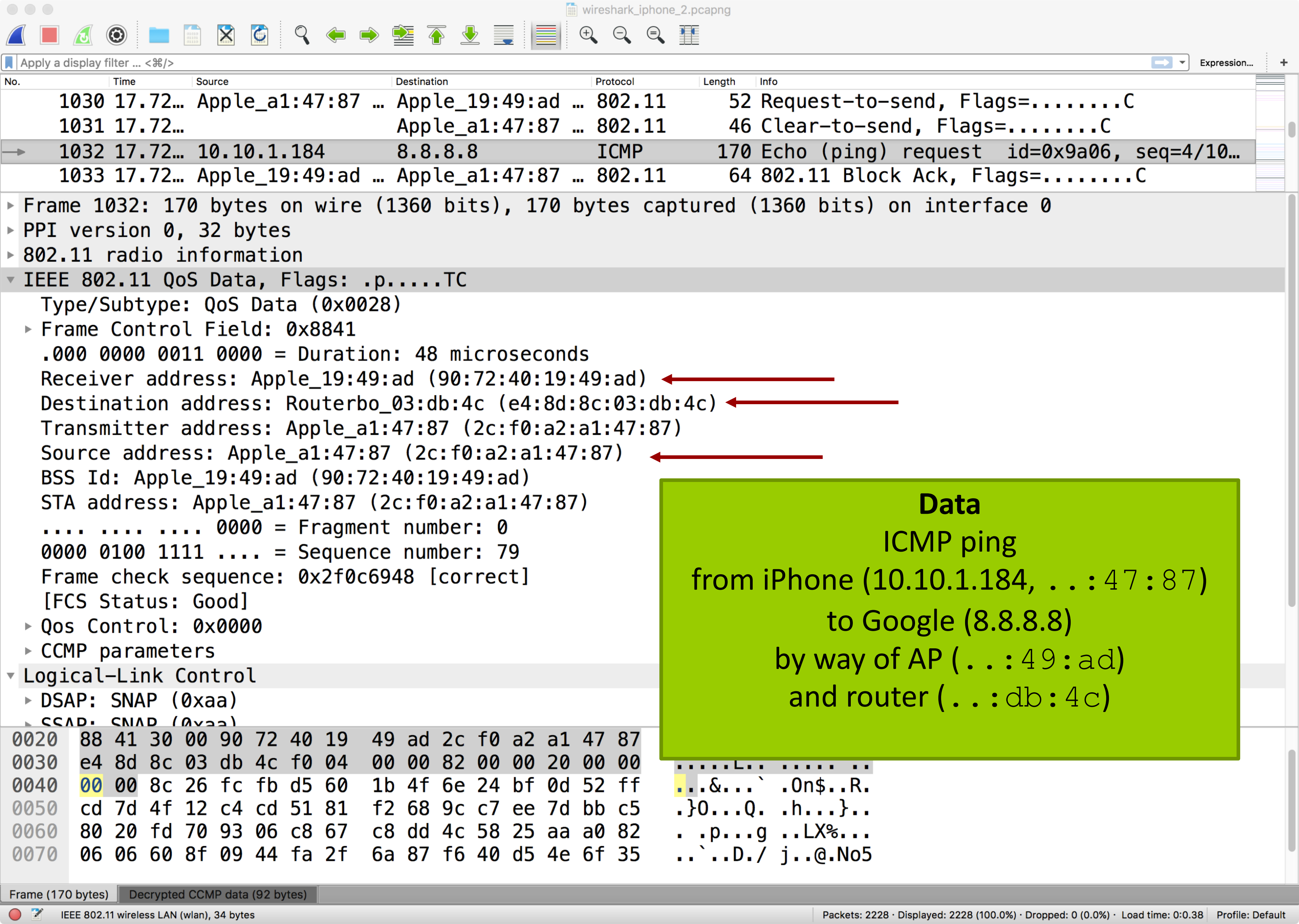
No.	Time	Source	Destination	Protocol	Length	Info
1030	17.72…	Apple_a1:47:87 …	Apple_19:49:ad …	802.11	52	Request-to-send, Flags=........C
1031	17.72…		Apple_a1:47:87 …	802.11	46	Clear-to-send, Flags=........C
1032	17.72…	10.10.1.184	8.8.8.8	ICMP	170	Echo (ping) request id=0x9a06, seq=4/10…
1033	17.72…	Apple_19:49:ad …	Apple_a1:47:87 …	802.11	64	802.11 Block Ack, Flags=........C

```
▸ Frame 1031: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface 0
▸ PPI version 0, 32 bytes
▸ 802.11 radio information
▾ IEEE 802.11 Clear-to-send, Flags: ........C
    Type/Subtype: Clear-to-send (0x001c)
  ▸ Frame Control Field: 0xc400
    .000 0000 0101 1100 = Duration: 92 microseconds
    Receiver address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
    Frame check sequence: 0x5b319cac [correct]
    [FCS Status: Good]
```

Clear-to-Send
Sent from Access Point
to iPhone (..:47:87)

```
0000  00 00 20 00 69 00 00 00  02 00 14 00 ef 30 41 72   .. .i... .....0Ar
0010  00 00 00 00 01 00 30 00  71 16 40 01 00 00 d2 a9   ......0. q.@.....
0020  c4 00 5c 00 2c f0 a2 a1  47 87 ac 9c 31 5b         ..\.,... G...1[
```

IEEE 802.11 wireless LAN (wlan), 10 bytes   Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.38   Profile: Default


wireshark_iphone_2.pcapng
Apply a display filter … <⌘/>
Expression…
No. Time Source Destination Protocol Length Info
1030 17.72… Apple_a1:47:87 … Apple_19:49:ad … 802.11 52 Request-to-send, Flags=........C
1031 17.72… Apple_a1:47:87 … 802.11 46 Clear-to-send, Flags=........C
1032 17.72… 10.10.1.184 8.8.8.8 ICMP 170 Echo (ping) request id=0x9a06, seq=4/10…
1033 17.72… Apple_19:49:ad … Apple_a1:47:87 … 802.11 64 802.11 Block Ack, Flags=........C
Frame 1032: 170 bytes on wire (1360 bits), 170 bytes captured (1360 bits) on interface 0
PPI version 0, 32 bytes
802.11 radio information
IEEE 802.11 QoS Data, Flags: .p.....TC
Type/Subtype: QoS Data (0x0028)
Frame Control Field: 0x8841
.000 0000 0011 0000 = Duration: 48 microseconds
Receiver address: Apple_19:49:ad (90:72:40:19:49:ad)
Destination address: Routerbo_03:db:4c (e4:8d:8c:03:db:4c)
Transmitter address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
Source address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
BSS Id: Apple_19:49:ad (90:72:40:19:49:ad)
STA address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
.... 0000 = Fragment number: 0
0000 0100 1111 .... = Sequence number: 79
Frame check sequence: 0x2f0c6948 [correct]
[FCS Status: Good]
Qos Control: 0x0000
CCMP parameters
Logical-Link Control
DSAP: SNAP (0xaa)
0020 88 41 30 00 90 72 40 19 49 ad 2c f0 a2 a1 47 87
0030 e4 8d 8c 03 db 4c f0 04 00 00 82 00 00 20 00 00
0040 00 00 8c 26 fc fb d5 60 1b 4f 6e 24 bf 0d 52 ff ...&...` .On$..R.
0050 cd 7d 4f 12 c4 cd 51 81 f2 68 9c c7 ee 7d bb c5 .}O...Q. .h...}..
0060 80 20 fd 70 93 06 c8 67 c8 dd 4c 58 25 aa a0 82 . .p...g ..LX%...
0070 06 06 60 8f 09 44 fa 2f 6a 87 f6 40 d5 4e 6f 35 ..`..D./ j..@.No5
Frame (170 bytes)
Decrypted CCMP data (92 bytes)
IEEE 802.11 wireless LAN (wlan), 34 bytes
Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.38
Profile: Default
Data
ICMP ping
from iPhone (10.10.1.184, ..:47:87)
to Google (8.8.8.8)
by way of AP (..:49:ad)
and router (..:db:4c)

wireshark_iphone_2.pcapng

Apply a display filter … <⌘/>

Expression…

No.	Time	Source	Destination	Protocol	Length	Info
1030	17.72…	Apple_a1:47:87 …	Apple_19:49:ad …	802.11	52	Request-to-send, Flags=........C
1031	17.72…		Apple_a1:47:87 …	802.11	46	Clear-to-send, Flags=........C
1032	17.72…	10.10.1.184	8.8.8.8	ICMP	170	Echo (ping) request id=0x9a06, seq=4/10…
1033	17.72…	Apple_19:49:ad …	Apple_a1:47:87 …	802.11	64	802.11 Block Ack, Flags=........C

```
▸ Frame 1033: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0
▸ PPI version 0, 32 bytes
▸ 802.11 radio information
▾ IEEE 802.11 802.11 Block Ack, Flags: ........C
    Type/Subtype: 802.11 Block Ack (0x0019)
  ▸ Frame Control Field: 0x9400
    .000 0000 0000 0000 = Duration: 0 microseconds
    Receiver address: Apple_a1:47:87 (2c:f0:a2:a1:47:87)
    Transmitter address: Apple_19:49:ad (90:72:40:19:49:ad)
    .... .10. = Block Ack Type: Compressed Block (0x2)
  ▸ Block Ack Request Control: 0x0005
  ▸ Block Ack Starting Sequence Control (SSC): 0x04f0
  ▸ Block Ack Bitmap: 0100000000000000
    Frame check sequence: 0x565263e4 [correct]
    [FCS Status: Good]
```

Acknowledgement

Sent from Access Point (..:49:ad)

to iPhone (..:47:87)

```
0000  00 00 20 00 69 00 00 00  02 00 14 00 58 31 41 72   .. .i... ....X1Ar
0010  00 00 00 00 01 00 30 00  71 16 40 01 00 00 d2 a9   ......0. q.@.....
0020  94 00 00 00 2c f0 a2 a1  47 87 90 72 40 19 49 ad   ....,... G..r@.I.
0030  05 00 f0 04 01 00 00 00  00 00 00 00 e4 63 52 56   ........ .....cRV
```

IEEE 802.11 wireless LAN (wlan), 16 bytes

Packets: 2228 · Displayed: 2228 (100.0%) · Dropped: 0 (0.0%) · Load time: 0:0.38 Profile: Default

802.11 and Security

Authentication and Association

- 802.11 network attachment process
  - **Authentication** – Station establishes its identity (MAC address) with AP
    - No password was verified!
    - No encryption!
    - This is "Open system authentication"
  - **Association** – Station chooses a specific AP to send/receive data with
- **So where does *security* come into play?**

WiFi Security

- IEEE 802.11 – WEP
  - **WEP** – Wired Equivalent Privacy
  - Used RC4 stream cipher
  - Insecure, don’t use! ☹

WiFi Security

- IEEE 802.11i – WPA and WPA2
  - **WPA** – WiFi Protected Access
    - *Transitional* spec
    - Requires support for TKIP and RC4 stream cipher
      - Insecure, don't use! ☹
    - Optional support for AES-CCMP
  - **WPA2**
    - Final spec
    - Also known as RSN (Robust Security Network)
    - **Required support for AES-CCMP – Use this!**

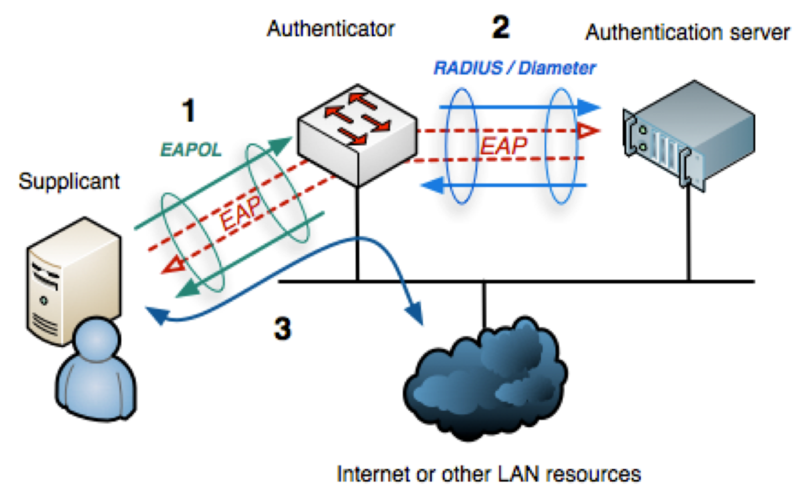
IEEE 802.11i, WPA2

- **AES-CCMP?**
- Advanced Encryption Standard (AES)
  - Block cipher, symmetric key encryption
- **C**ounter Mode **C**ipher Block Chaining **M**essage Authentication Code **P**rotocol (CCMP)
  - Operating mode for AES cipher
    - Counter Mode, Cipher Block Chaining
    - Message Authentication Code
  - Confidentiality + Integrity

IEEE 802.11i, WPA2

- Extensible Authentication Protocol (**EAP**)
- Extensible Authentication Protocol over LAN **(EAPOL)**
  - Defined in IEEE 802.1X standard
  - Used in wired Ethernet (802.3) and WiFi (802.11)

- Terminology
  - *Supplicant* – Client device (e.g. your laptop)
  - *Authenticator* – The AP
  - *Authentication server* – RADIUS server (or AP)



IEEE 802.11i, WPA2

- 4-Way Handshake
- Proven secure (in part)
  - Key secrecy and session authentication
  - Key ordering and key secrecy for group key handshake

He, Changhua & Sundararajan, Mukund & Datta, Anupam & Derek, Ante & C. Mitchell, John. (2005). ***A modular correctness proof of IEEE 802.11i and TLS***. Proceedings of the ACM Conference on Computer and Communications Security.

IEEE 802.11i, WPA2

- Pairwise Master Key (**PMK**)
  - Shared Secret between client and AP
  - Derived from pre-shared password in personal network
    - PBKDF2 key derivation function
    - SSID is salt
    - 4096 iterations of HMAC-SHA1
  - Derived from 802.1x authentication in enterprise network
    - RADIUS authentication server

https://papers.mathyvanhoef.com/ccs2017.pdf

IEEE 802.11i, WPA2

- Pairwise Transient Key (**PTK**) – 64 bytes
  - Used as a session key (is changed periodically)
  - Derived from:
    - PMK
    - Authenticator Nonce (ANonce)
    - Supplicant Nonce (SNonce)
    - MAC address of authenticator
    - MAC address of supplicant

https://papers.mathyvanhoef.com/ccs2017.pdf

IEEE 802.11i, WPA2

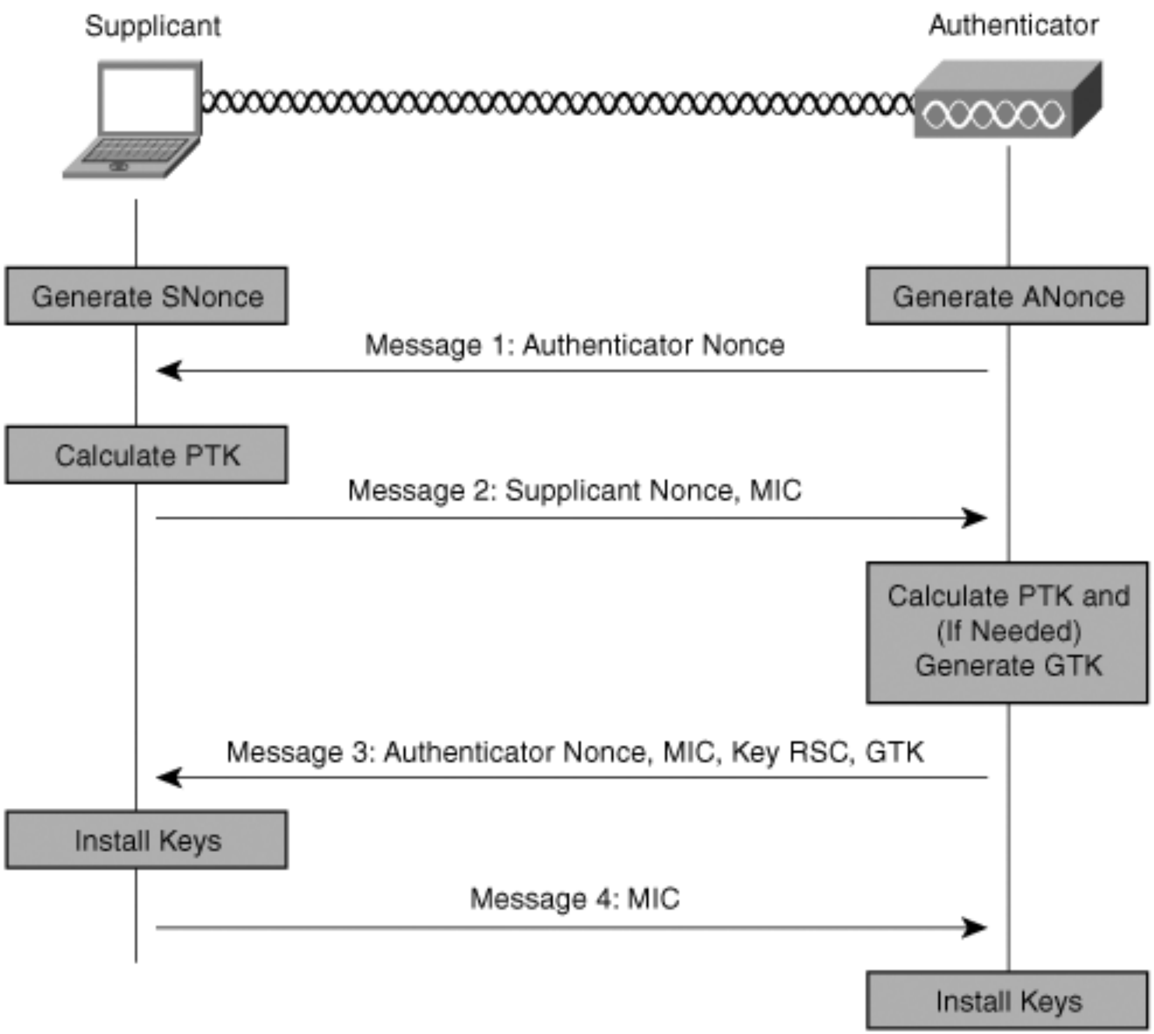
- Client never tells AP its PMK/PTK (or vice-versa)
  - Decryption only works if both parties independently calculate the same PMK/PTK
  - Protects against rogue AP impersonating real AP

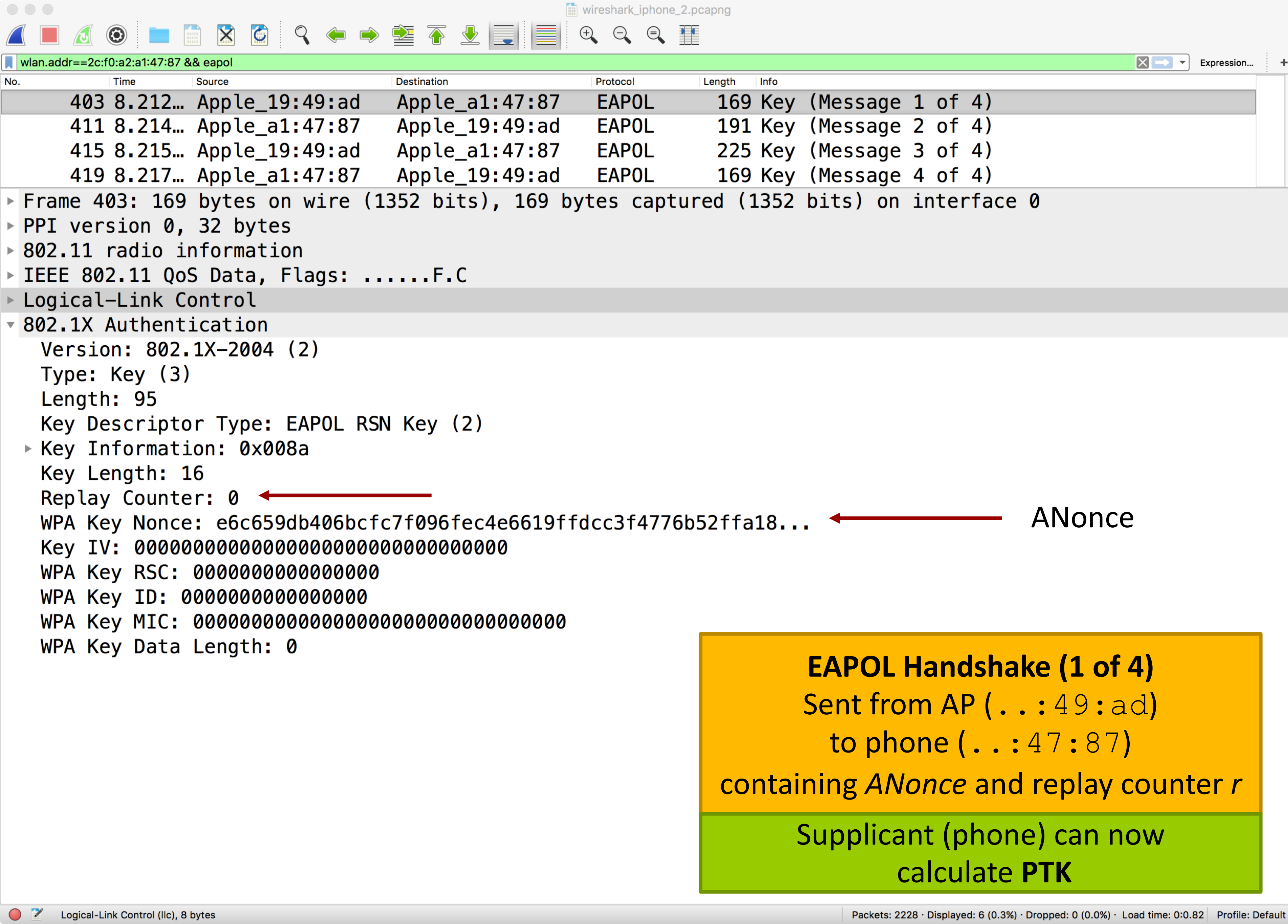
https://papers.mathyvanhoef.com/ccs2017.pdf

IEEE 802.11i, WPA2

- Keys obtained from slices of PTK
  - Key Confirmation Key (**KCK**) – 16 bytes of PTK
  - Key Encryption Key (**KEK**) – 16 bytes of PTK
    - Protects handshake messages
  - Temporal Key (**TK**) – 16 bytes of PTK
    - Protects normal data frames
  - Message Integrity Check (**MIC**) for TX – 8 bytes of PTK
  - Message Integrity Check (**MIC**) for RX – 8 bytes of PTK
- Group Temporal Key (**GTK**)
  - Used for broadcast and multicast frames
  - Randomly generated by AP
  - Rotated periodically

Authentication with EAPOL





EAPOL Handshake (1 of 4)

Sent from AP (..:49:ad)

to phone (..:47:87)

containing *ANonce* and replay counter *r*

Supplicant (phone) can now calculate **PTK**

wireshark_iphone_2.pcapng

wlan.addr==2c:f0:a2:a1:47:87 && eapol

Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
403	8.212…	Apple_19:49:ad	Apple_a1:47:87	EAPOL	169	Key (Message 1 of 4)
411	8.214…	Apple_a1:47:87	Apple_19:49:ad	EAPOL	191	Key (Message 2 of 4)
415	8.215…	Apple_19:49:ad	Apple_a1:47:87	EAPOL	225	Key (Message 3 of 4)
419	8.217…	Apple_a1:47:87	Apple_19:49:ad	EAPOL	169	Key (Message 4 of 4)

```
▸ Frame 411: 191 bytes on wire (1528 bits), 191 bytes captured (1528 bits) on interface 0
▸ PPI version 0, 32 bytes
▸ 802.11 radio information
▸ IEEE 802.11 QoS Data, Flags: .......TC
▸ Logical-Link Control
▾ 802.1X Authentication
    Version: 802.1X-2004 (2)
    Type: Key (3)
    Length: 117
    Key Descriptor Type: EAPOL RSN Key (2)
  ▸ Key Information: 0x010a
    Key Length: 16
    Replay Counter: 0
    WPA Key Nonce: efbc821bf790bc830df290697cb50d2996b05ac3d61d4f15...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000000000
    WPA Key ID: 0000000000000000
    WPA Key MIC: 932ee74d84bb7a9158ea9ff2277b268a
    WPA Key Data Length: 22
  ▸ WPA Key Data: 30140100000fac040100000fac040100000fac020c(
```

SNonce

EAPOL Handshake (2 of 4)

Sent from phone (`..:47:87`)

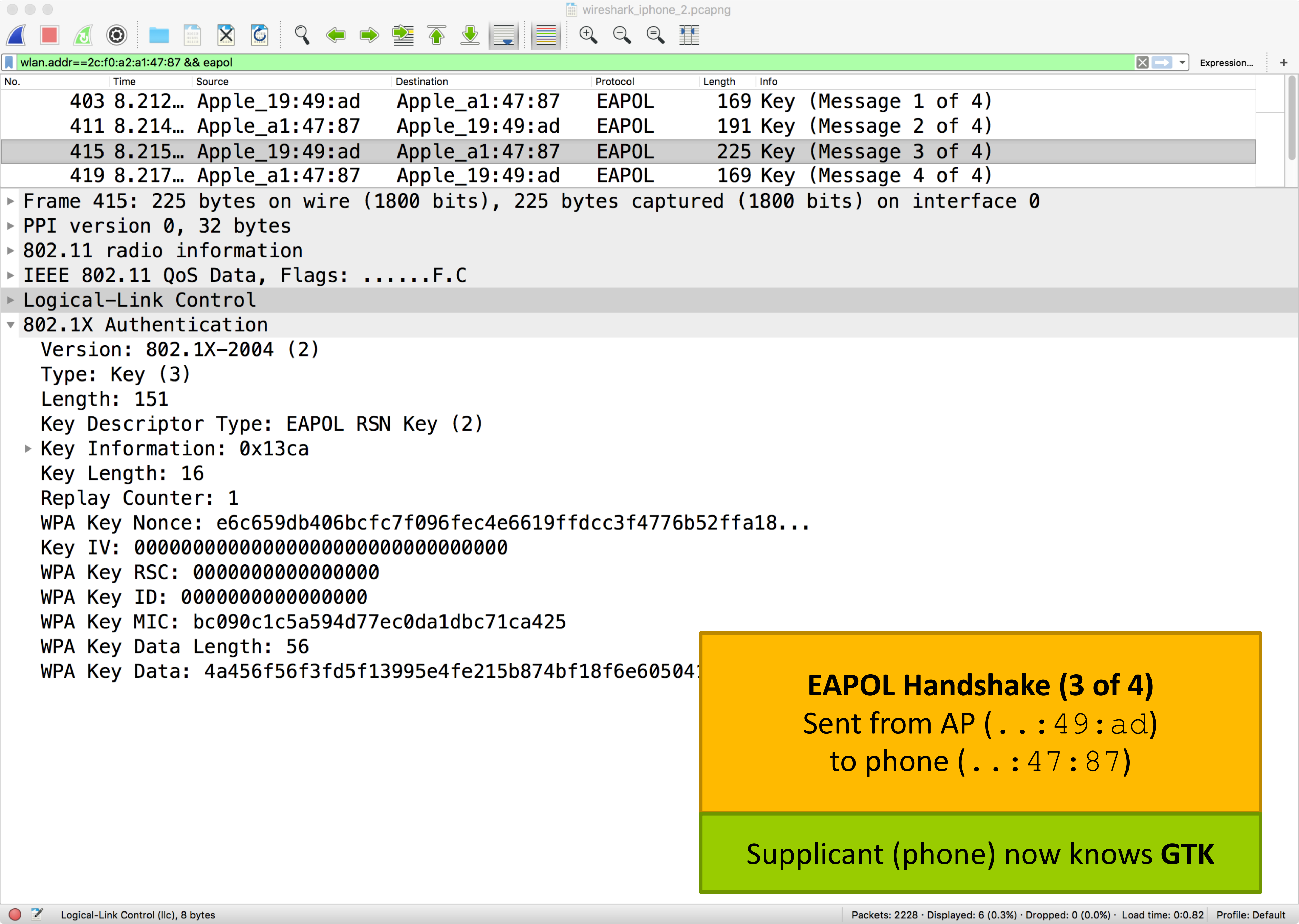
to AP (`..:49:ad`)

containing *SNonce* and replay counter *r*

Authenticator (AP) can now calculate **PTK**

Logical-Link Control (llc), 8 bytes

Packets: 2228 · Displayed: 6 (0.3%) · Dropped: 0 (0.0%) · Load time: 0:0.82 Profile: Default



EAPOL Handshake (3 of 4)
Sent from AP (..:49:ad)
to phone (..:47:87)

Supplicant (phone) now knows **GTK**

wireshark_iphone_2.pcapng

wlan.addr==2c:f0:a2:a1:47:87 && eapol

Expression... +

No.	Time	Source	Destination	Protocol	Length	Info
403	8.212…	Apple_19:49:ad	Apple_a1:47:87	EAPOL	169	Key (Message 1 of 4)
411	8.214…	Apple_a1:47:87	Apple_19:49:ad	EAPOL	191	Key (Message 2 of 4)
415	8.215…	Apple_19:49:ad	Apple_a1:47:87	EAPOL	225	Key (Message 3 of 4)
419	8.217…	Apple_a1:47:87	Apple_19:49:ad	EAPOL	169	Key (Message 4 of 4)

```
▸ Frame 419: 169 bytes on wire (1352 bits), 169 bytes captured (1352 bits) on interface 0
▸ PPI version 0, 32 bytes
▸ 802.11 radio information
▸ IEEE 802.11 QoS Data, Flags: .......TC
▸ Logical-Link Control
▾ 802.1X Authentication
    Version: 802.1X-2004 (2)
    Type: Key (3)
    Length: 95
    Key Descriptor Type: EAPOL RSN Key (2)
  ▸ Key Information: 0x030a
    Key Length: 16
    Replay Counter: 1
    WPA Key Nonce: 000000000000000000000000000000000000000000000000...
    Key IV: 00000000000000000000000000000000
    WPA Key RSC: 0000000000000000
    WPA Key ID: 0000000000000000
    WPA Key MIC: b89f6b76a2d88decc09180c881de969a
    WPA Key Data Length: 0
```

EAPOL Handshake (4 of 4)

Sent from phone (..:47:87)

to AP (..:49:ad)

Supplicant and authenticator install PTK

Ready to exchange encrypted data

Logical-Link Control (llc), 8 bytes

Packets: 2228 · Displayed: 6 (0.3%) · Dropped: 0 (0.0%) · Load time: 0:0.82 Profile: Default

WPA2 Decryption

- WPA2 Personal
  - Every user shares the same password
  - Every user shares the same Pairwise Master Key (PMK) which is derived from the password
  - Each user has a *different* Pairwise Transient Key (PTK) …
    - … but can be calculated by a party that knows the password and can observe the ANonce and Snonce handshake
  - Result = Traffic can be decrypted

WPA2 Decryption

- WPA2 Enterprise (802.1x)
  - Each user has a *different* password
  - Every user has a *different* Pairwise Master Key (PMK) which is derived from the password
  - Each user has a *different* Pairwise Transient Key (PTK)
  - Result = Traffic cannot be decrypted
    - Unless malicious actor can steal PMK from client or Radius server

Key **R**einstallation **A**tta**ck**s
Breaking WPA2 by forcing nonce reuse

Discovered by Mathy Vanhoef of
imec-DistriNet, KU Leuven

https://www.krackattacks.com/

WPA2 KRACK 🙀

WPA2 KRACK

When a client joins a network, it executes the 4-way handshake to negotiate a fresh encryption key. It will install this key after receiving message 3 of the 4-way handshake. Once the key is installed, it will be used to encrypt normal data frames using an encryption protocol. However, because messages may be lost or dropped, the Access Point (AP) will retransmit message 3 if it did not receive an appropriate response as acknowledgment. As a result, the client may receive message 3 multiple times. Each time it receives this message, it will reinstall the same encryption key, and thereby reset the incremental transmit packet number (nonce) and receive replay counter used by the encryption protocol. We show that ***an attacker can force these nonce resets by collecting and replaying retransmissions of message 3 of the 4-way handshake****. By forcing nonce reuse in this manner, the encryption protocol can be attacked, e.g., packets can be replayed, decrypted, and/or forged. The same technique can also be used to attack the group key, PeerKey, TDLS, and fast BSS transition handshake.*

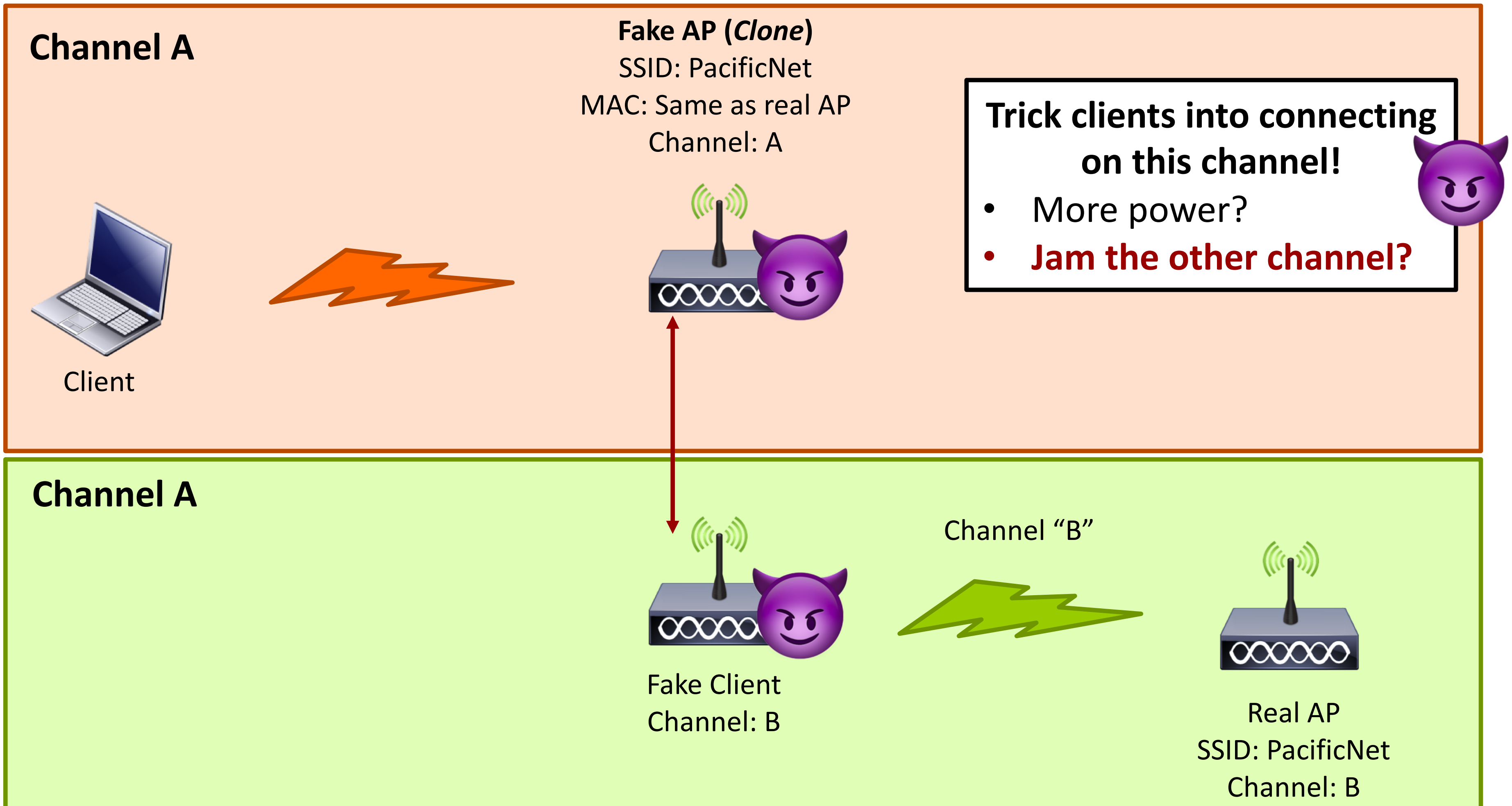
https://www.krackattacks.com/

WPA2 KRACK

- Design flaw (oversight) in specification for *client* state machine
  - Does not affect APs
- Attacker obtains MiTM position between supplicant and authenticator
- Attacker uses MiTM position to trigger retransmissions of Msg3 by preventing Msg4 from arriving at the authenticator

https://papers.mathyvanhoef.com/ccs2017.pdf

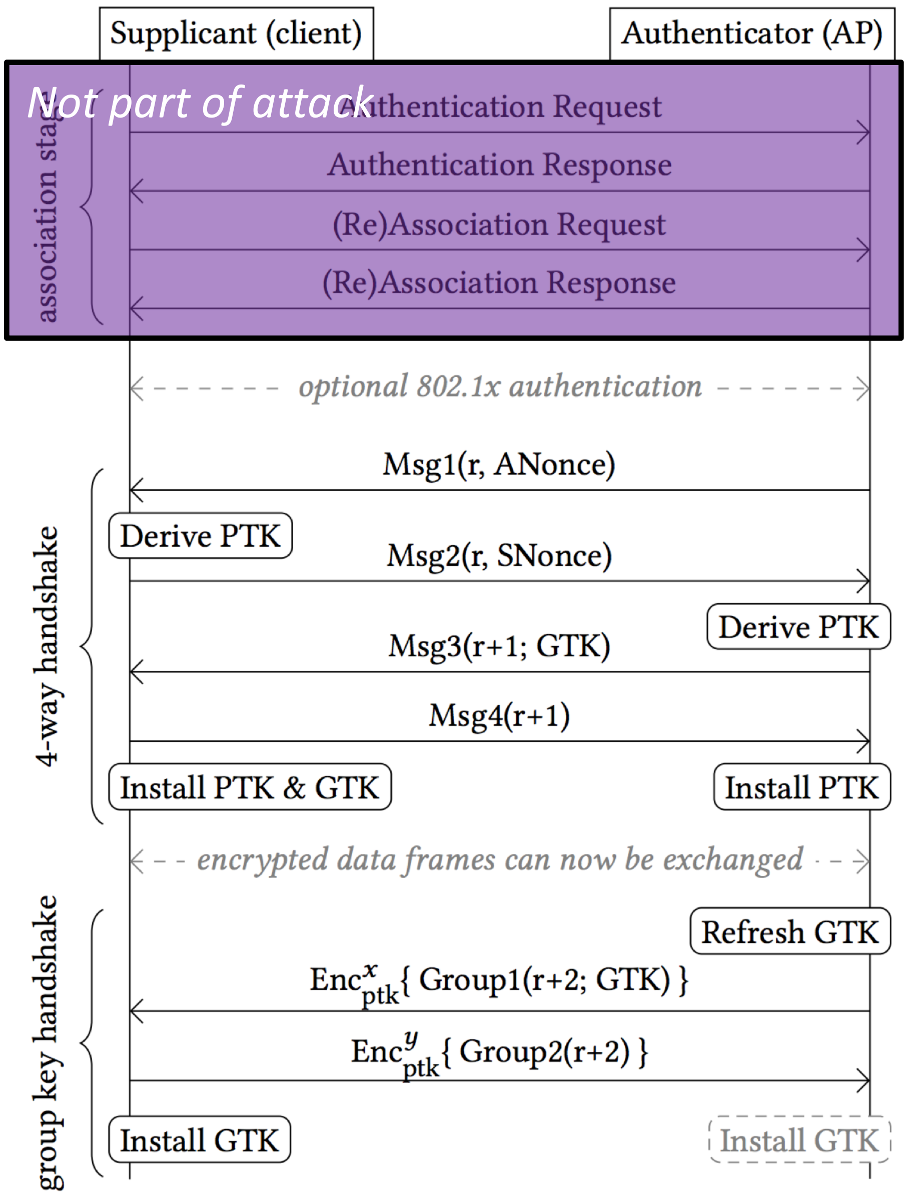
MiTM on WiFi: *Channel-Based Attack*

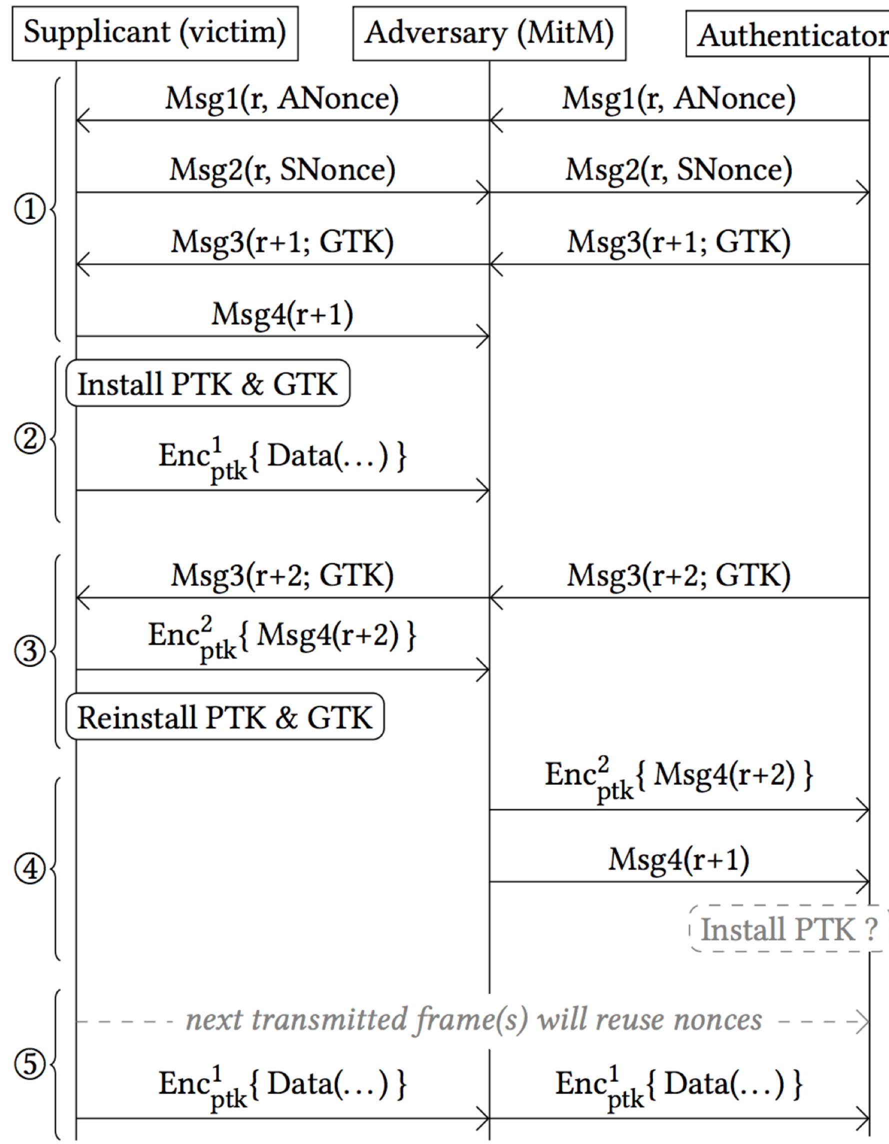


http://www.mathyvanhoef.com/2015/10/advanced-wifi-attacks-using-commodity.html

Normal

Key Reinstallation Attack





WPA2 KRACK

- Same attack can be used elsewhere in 802.11
  - Group key, PeerKey, TDLS, fast BSS transition handshake
- Impact: If the attackers knows the *plaintext* (e.g. HTTP cookie) and the nonce is re-used, attacker can decrypt other frames that were encrypted with same nonce
  - Presumes a patient attacker with some knowledge of likely victim behavior
  - Result: TCP Hijacking attack
- Impact: Attacker does *not* recover password

https://papers.mathyvanhoef.com/ccs2017.pdf

Defense?

- Step 1: Patch your clients
- Step 2: Monitor for MiTM WiFi attacks
  - `EvilAP_Defender` can look for Evil Aps
    - Evil AP with a different BSSID address
    - Evil AP with the same BSSID as the legitimate AP but a different attribute (including: channel, cipher, privacy protocol, and authentication)
    - Evil AP with the same BSSID and attributes as the legitimate AP but different tagged parameter - mainly different OUI (tagged parameters are additional values sent along with the beacon frame)
  - Notify Admin (email) + DoS the malicious AP(!)
  - https://github.com/moha99sa/EvilAP_Defender/wiki

Packet Capture

Packet Capture

How do I capture packets from a WiFi network?

Plan "A"

Plan "A" Challenges

- Have to walk around the office carrying extra hardware
- USB WiFi adaptors are cheap
  - Low cost *and* low quality
- Hit or miss finding USB WiFi adaptors that support *monitor mode*
  - Manufacturers often use different chipsets in "same" product!
- Specifications
  - Enterprise WiFi that you want to capture:
    - 802.11ac
    - MU-MIMO 3x3 or more
  - USB WiFi Adaptors used to monitor network:
    - 802.11a/b/g/n (ac *maybe)*
    - MIMO 2x2 at most (how many antennas do you see?)

Plan "A" Challenges

- Common problem:
  - Why don't I see my test pings in my capture? (I see plenty of beacon messages from my APs)
- *Pings were sent over extended channel, or via MIMO path, or at a higher data rate than your USB WiFi adaptor supports*

Plan “B”

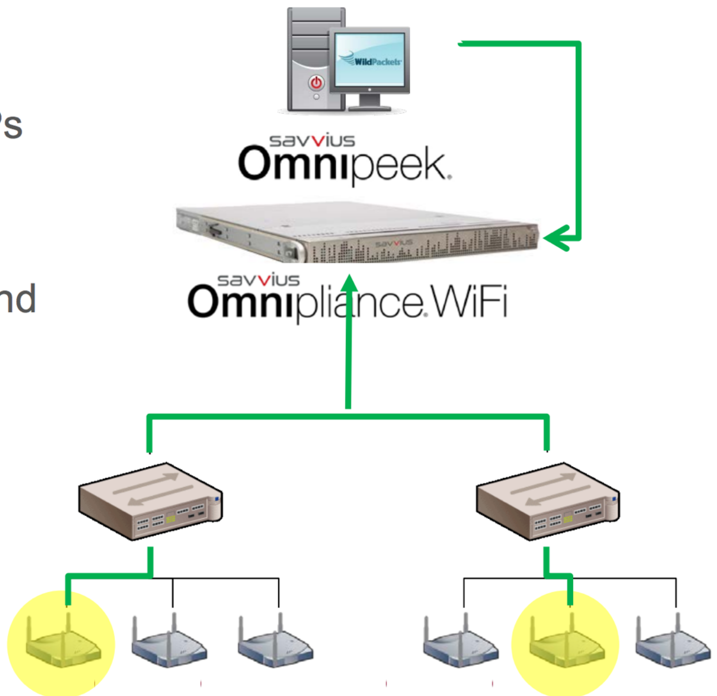
Rackmount Server
+ Storage (Lots!)
+ Fast NICs

Same Enterprise APs used by production network

- Example: Savvius *Omnipliance* product
  - No need to leave your desk!

How Omnipliance WiFi Works

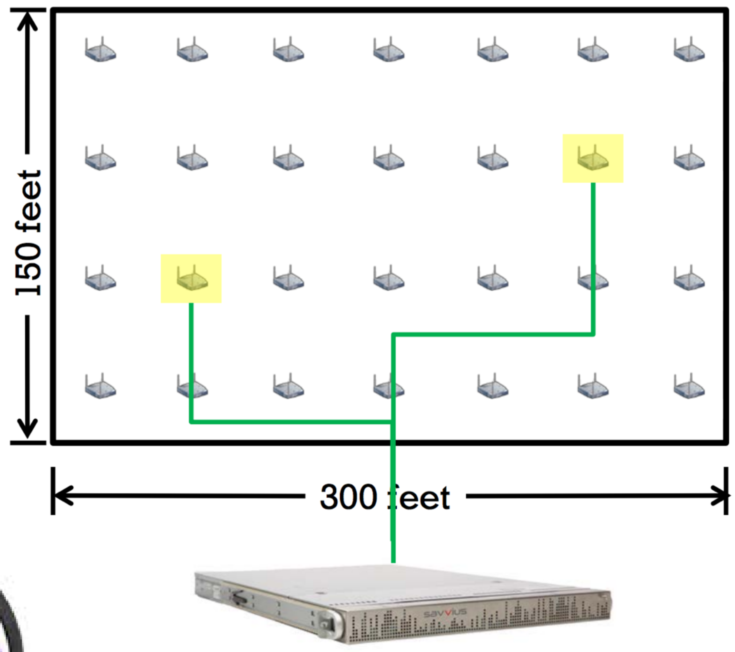
1. Using the WLAN controller UI, put the desired APs in “sniffer” mode, and direct the packets to Omnipliance WiFi – packets start flowing
2. Using Omnipeek, connect to Omnipliance WiFi and configure your Remote Adapter capture
3. Start the capture – analysis (and storage) of all packets from the APs begin immediately



http://www.youtube.com/embed/BcWWeufQn7Q

https://www.cwnp.com/nola-ppt-pdfs/WedPresentations/Jay%20Savvius%20-%20Packet%20Capture%20and%20Analysis%20in%20the%20MU-MIMO%2011ac%20World.pdf

Highly Distributed, Multi-Campus Deployment



- Dense deployment ~ 28 APs per building floor
- 100's of building floors
- Reactive capture and analysis

https://www.cwnp.com/nola-ppt-pdfs/WedPresentations/Jay%20Savvius%20-%20Packet%20Capture%20and%20Analysis%20in%20the%20MU-MIMO%2011ac%20World.pdf