SHARKFEST 2015

WIRESHARK DEVELOPER AND USER CONFERENCE

Wireless Troubleshooting Tips using AirPcaps: DFS & Module Debugging

COMPUTER HISTORY MUSEUM

Megumi Takeshita

ikeriri network service co.,ltd

Megumi Takeshita, ikeriri network service a.k.a. packet otaku since first Sharkfest

- Founder, ikeriri network service co.,ltd I am network troubleshooter and debugger using packet analysis.
- Wrote 10+ books of packet capturing
- Reseller of Riverbed Technology and Metageek, Dualcomm etc. in Japan
- Attending all Sharkfest and first translator of QT Wireshark into Japanese！日本語ワイヤーシャーク

Wireless troubleshooting TIPS using AirPcaps: DFS & Module Debugging

- Now I talk about 20 TIPS and troubleshooting in wireless environment
- AirPcap(s) is necessary for debugging in Windows environment.
- Please ask me if you have some question.

#1 Collect host / AP info (Windows)

- chcp 437 (English codeset)
- “netsh wlan sh all | more “
  Driver description
  Driver version (important)
  INF file name
  MAC address
  SSID / BSSID
  authentication/encryption
  Channel / speed /signal
- Demonstration

#1 Collect host / AP info (iOS)

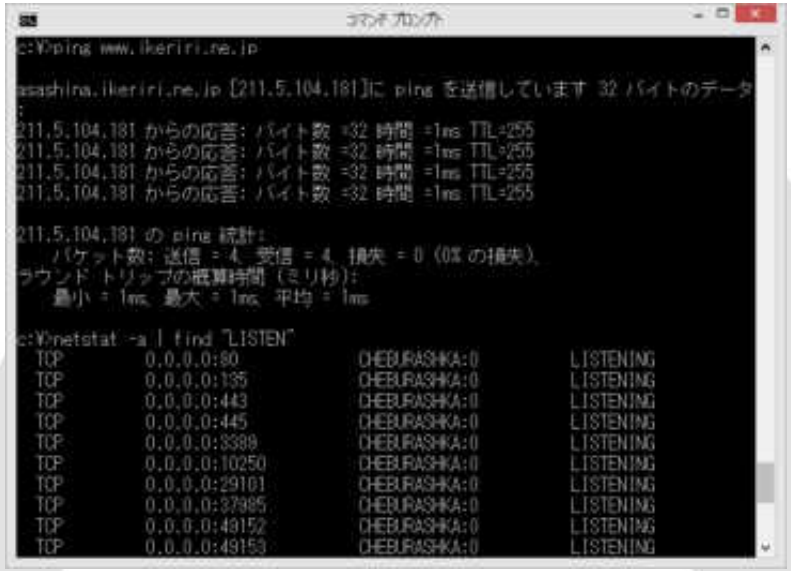
- Setting>General>Info
  “MAC address”
- Setting>Privacy>Location
  if “disabled” and no carrier setting may causes randomize
  MAC address (iOS8)
- Setting>Wi-Fi
  SSID / IP address / mask / gateway / DNS…

#1 Collect host / AP info (AP Side)

- SSID / BSSID / Channel / Channel bandwidth connection speed/mode encryption type / SSID etc.
- Also check the controller settings (if user use),
- Short Guard Interval 20 and Greenfield mode (High Throughput) are not supported by AirPcap series.

#2 Collect Baseline of network

- Latency and lost of Ping command
- tracert and pathping
- netstat –a | find “LISTEN”
- Iperf (throughput test)



- Demonstration

#3 Choose Physical header type

Type	Radiotap	PPI
Packet	⊟ Radiotap Header v0, Length 26 Header revision: 0 Header pad: 0 Header length: 26 ⊞ Present flags MAC timestamp: 297237576237288344 ⊞ Flags: 0x00 Data Rate: 1.0 Mb/s Channel frequency: 2427 [BG 4] ⊞ Channel type: 802.11b (0x00a0) SSI Signal: -41 dBm SSI Noise: -83 dBm Antenna: 0 SSI Signal: 42 dB	⊟ PPI version 0, 32 bytes Version: 0 ⊞ Flags: 0x00 Header length: 32 DLT: 105 ⊟ 802.11-Common Field type: 802.11-Common (2) Field length: 20 TSFT: 27056577967 ⊞ Flags: 0x0001 Rate: 1.0 Mbps Channel frequency: 2467 [BG 12] ⊞ Channel type: 802.11b (0x00a0) FHSS hopset: 0x00 FHSS pattern: 0x00 dBm antenna signal: -61 dBm antenna noise: -94

We can capture wireless frames as 2 kinds of frame format in Physical layer using AirPcap and Wireshark

#3 Choose Physical header type

Type	Radiotap	PPI
GOOD	• Easy to read, simple • Fixed format • Easy filter radiotap.dbm_antsignal	• Extensible format future info 11ac, etc • Includes multiple antenna information
BAD	• Cannot collect multiple anntena information	• Hard to read, complex • Long filter ppi.80211n-mac-phy.dbmant0.signal

- RECOMMEND Radiotap in 11a/b/g/n(20MHz)
- Demonstration Wireless toolbar> setting

#4 Use AirPcap(s)

- Using multiple AirPcaps tell us a different discovery of target devices (multiple channel info)
- We can use different PC with an AirPcap capturing specific channel (then merge pcap files)
- Trying 3 times or more sometimes AirPcap could not capture the packet.

#4 Use AirPcap(s)

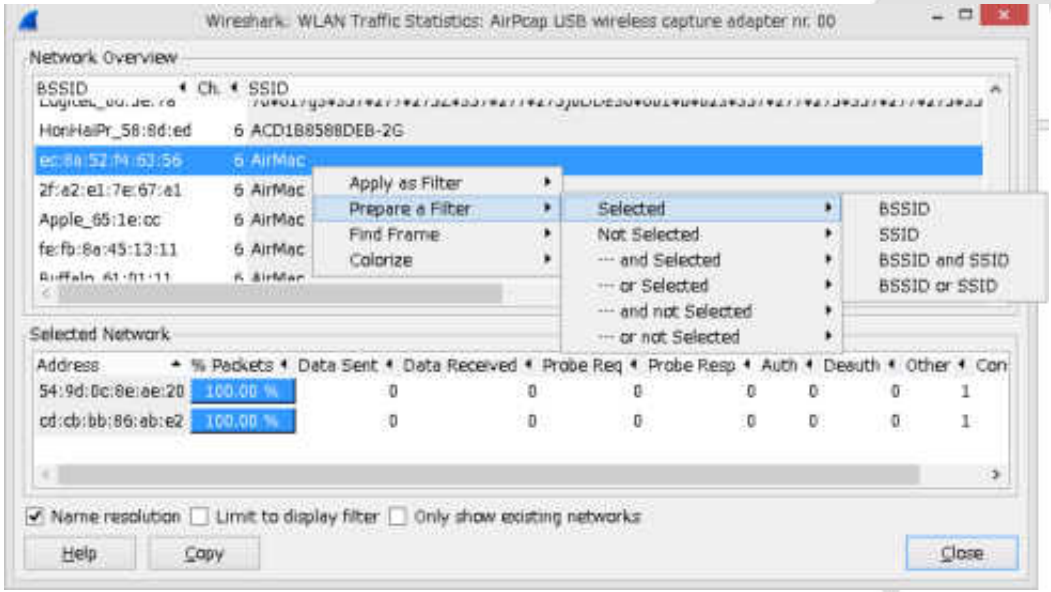
Setting	Offset -1	Offset 0	Offset +1
Channel	Main Channel 5 + Sub 1 1+5(40MHz)	Channel 5 (20MHz)	Main Channel 5 + Sub 9 5+9(40MHz)
	802.11 Channel: 2432 [BG 5] Channel Offset: -1 FCS No. Time Source Destination	802.11 Channel: 2432 [BG 5] Channel Offset: 0 FC No. Time Source Destination	802.11 Channel: 2432 [BG 5] Channel Offset: +1 F No. Time Source Destinatic

Setting	All Frame	Valid Frame	Invalid Frame
	FCS Filter: All Frames	FCS Filter: Valid Frames	FCS Filter: Invalid Frames

- Demonstration

#5 Filter packet in rough

- Wireless trace file is big,
  Connected wireless trace files are huge.
- Using Statistics>WLAN Traffic
  is the best way to filter packet in rough



#5 Filter packet in rough

- Once filter, or Mark packets or something, then File>Export specified packets.
- Iteration of exporting trace file, we can go back, look up the IO Graph, filtered packets at the moment.
- Small trace file is also good to open and read

testtt.pcap	2011/06/10 14:12	Wireshark capt...	116,212 KB
testwlan20011.pcap	2011/06/09 14:26	Wireshark capt...	42,244 KB

- Demonstration

#6 Customize summary pane

- Summary pane is the first chance to find the important packet
- Choosing field, right click to Apply as Column

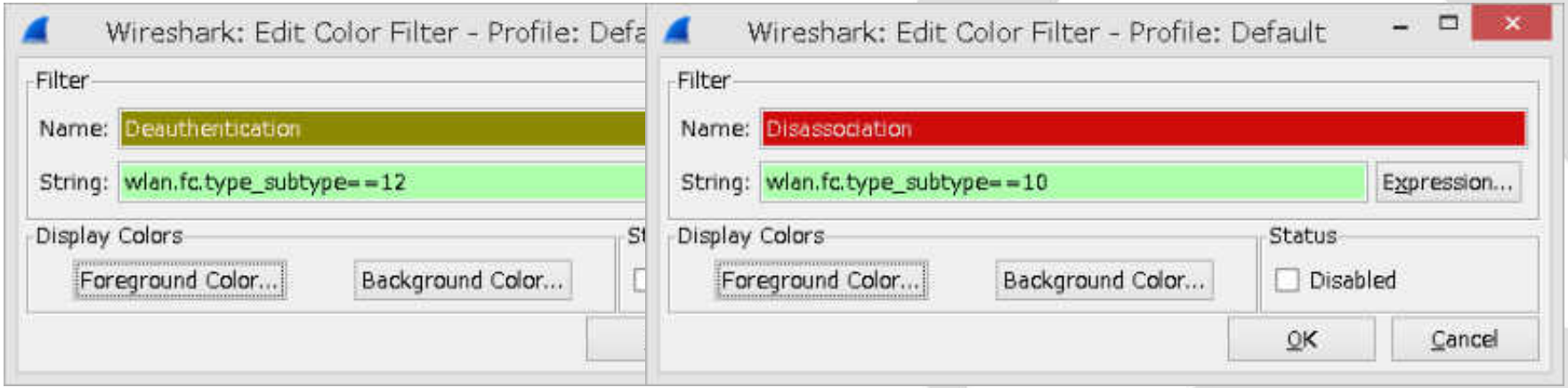
No.	Time	Channel	SigStrength	RSSI	Type/Subtype	TX Rate
1	0.000000					
2	0.000462					

Source	BSS Id	Destination	Protocol	Info
Matsushi_94:9f:1e		Broadcast	ARP	Who
Bug_31:34:f6		Matsushi_94:9f:1e	ARP	10.

- Type/Subtype … absolutely Apply as Column
  Channel / RSSI / SigStrength / TX Rate …

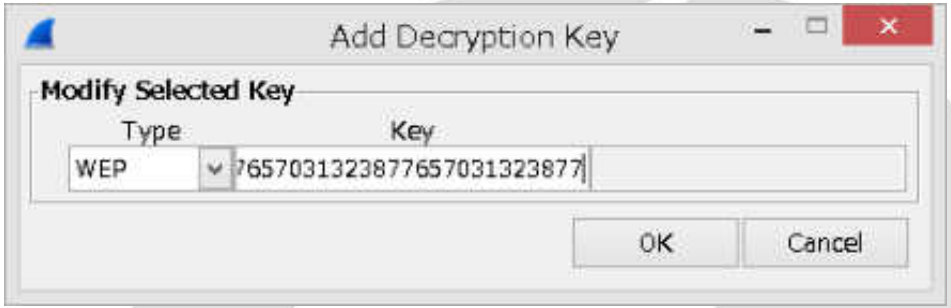
#7 Customize coloring rules

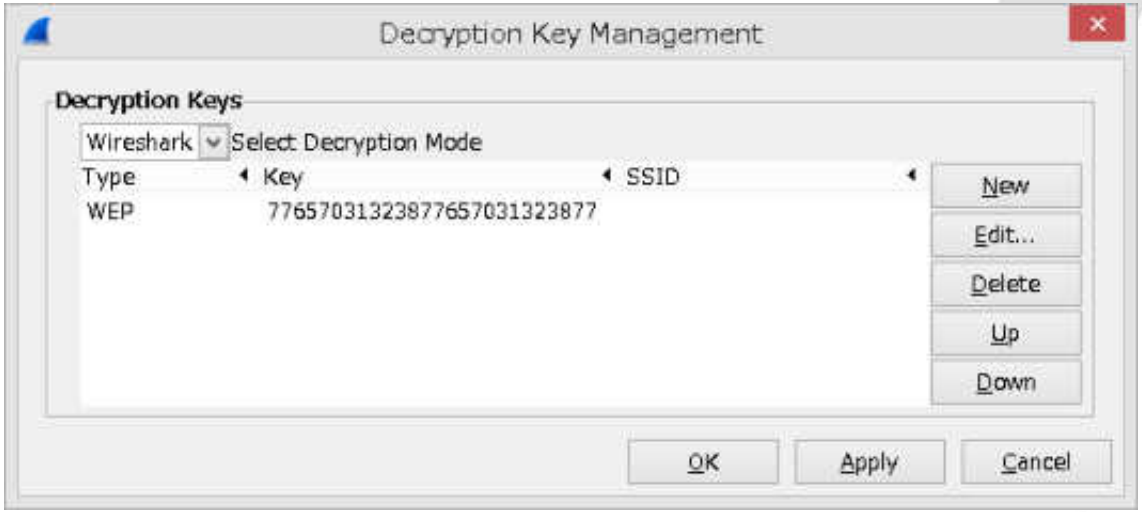
- typical troublesome packet
  Deauthentication from AP or from Client
  wlan.fc.type_subtype==12
  Disassociation from AP or from Client
  wlan.fc.type_subtype==10



#8 Set WEP Key

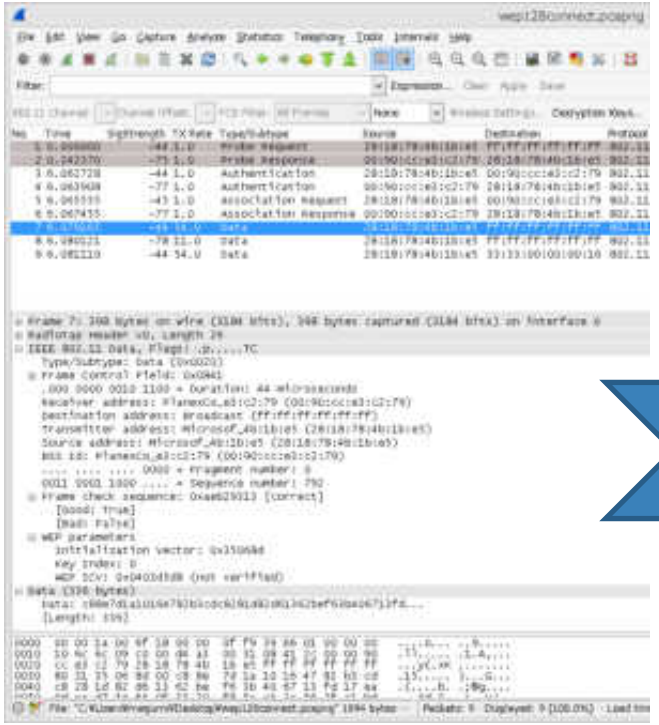
- WEP decryption in Wireshark is easy.
- Any AP, any Client any data frame can be decrypted if the key is correct

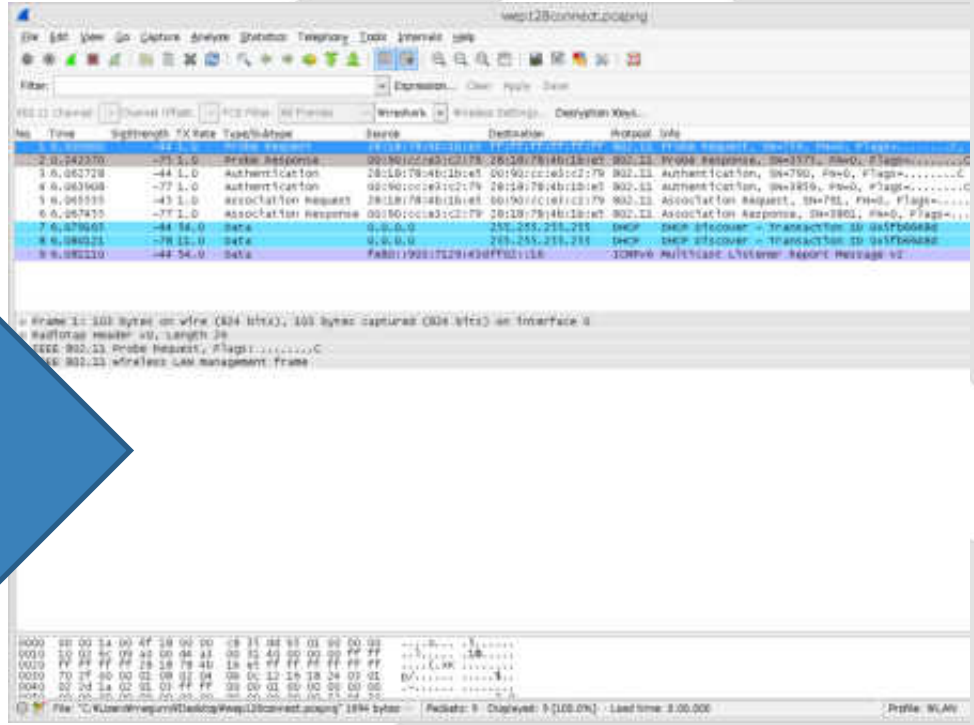




#8 Set WEP Key

- Remember to enter the key in ASCII format wep128wep128w
  77 65 70 31 32 38 77 65 70 31 32 38 77



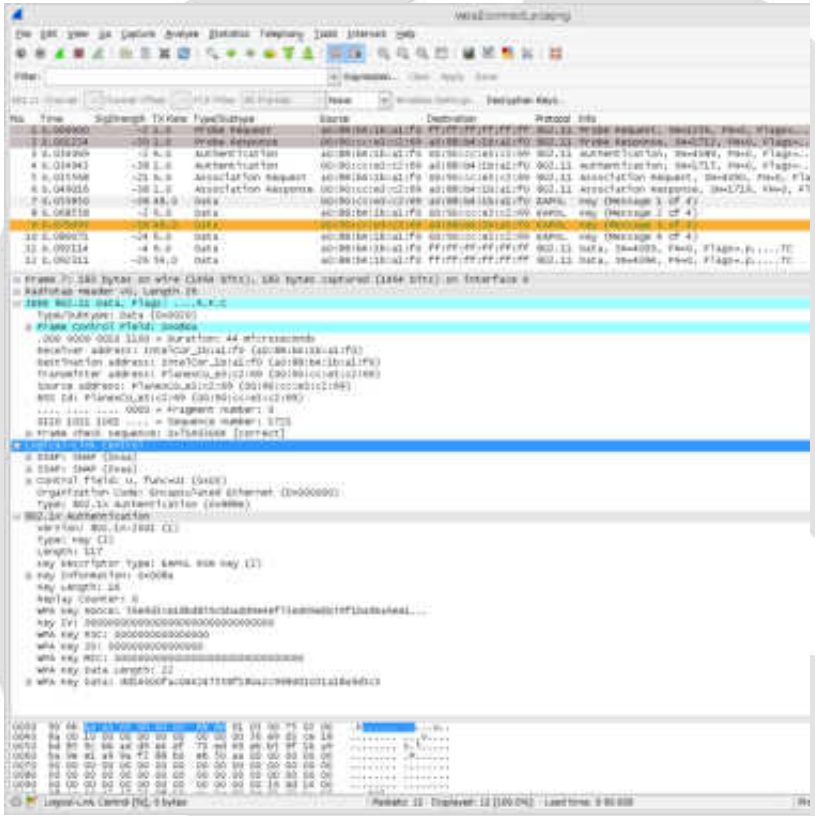


#9 Set WPA/WPA2 Key

- WPA 1/2 needs both Passphrase and SSID key input in alphabet format. (or PMK 256bit Hex)
- The difficulties lies in EAPOL 4-way handshake. The complete 4 packet of a series of handshake is necessary for decryption.
- Note some Windows and IOS use **the cache information** of the past connection to the AP, in this case, decryption fails.

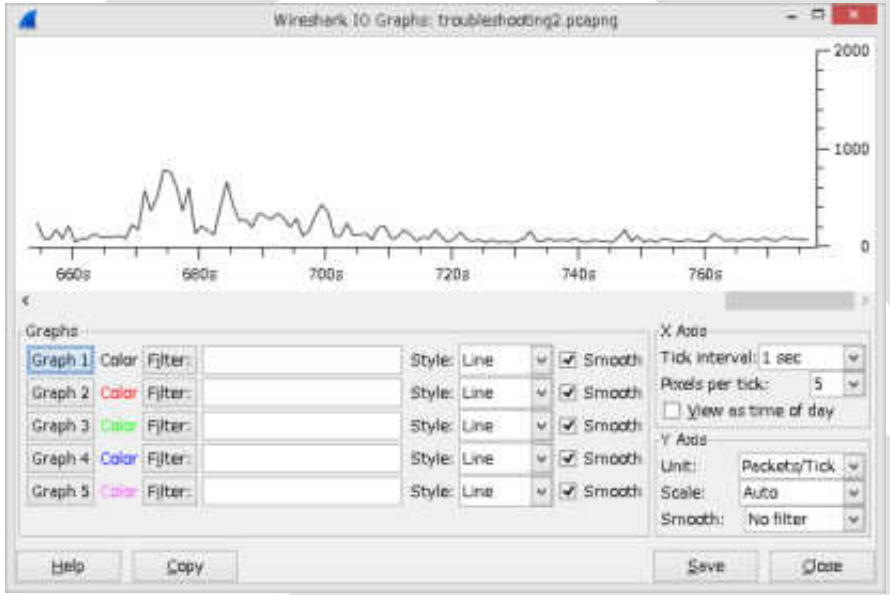
#9 Set WPA/WPA2 Key

- Please note the complete 4 way handshake
- Key/SSID wpa2aespsk



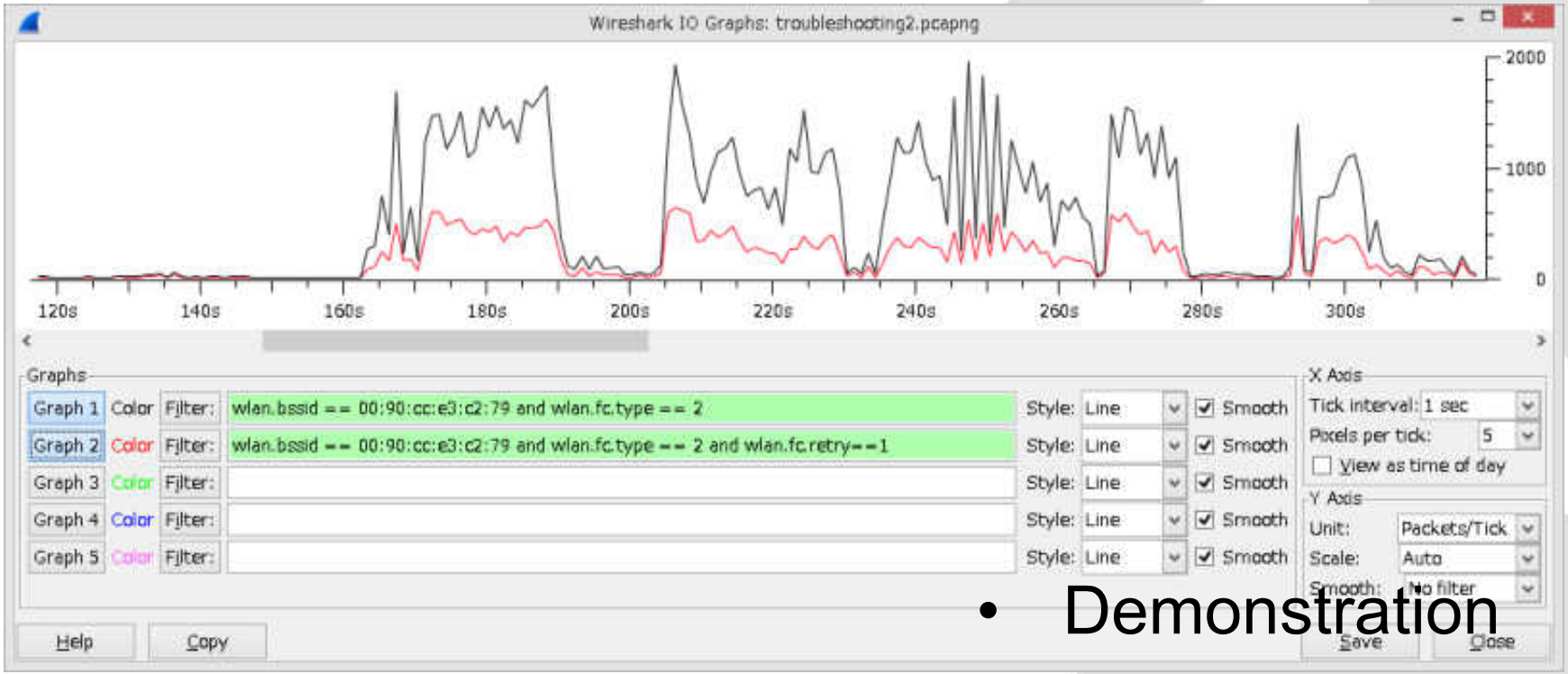
#10 Visualize (1) Retry

- Easy way to check the CSMA/CA status.
- We can check the retry packet rate, as well as the throughput of data frame.
- Filter packet within the specified AP or Client
- Statistics>IO Graph
  Retry rate graph
  Y/X axis -> packet/sec
  Throughput graph
  Y/X axis -> bit/sec



#10 Visualize (1) Retry

- Graph1: specified BSSID and data frame
- Graph2: the same with Graph1 and **“wlan.fc.retry==1”**



- Demonstration

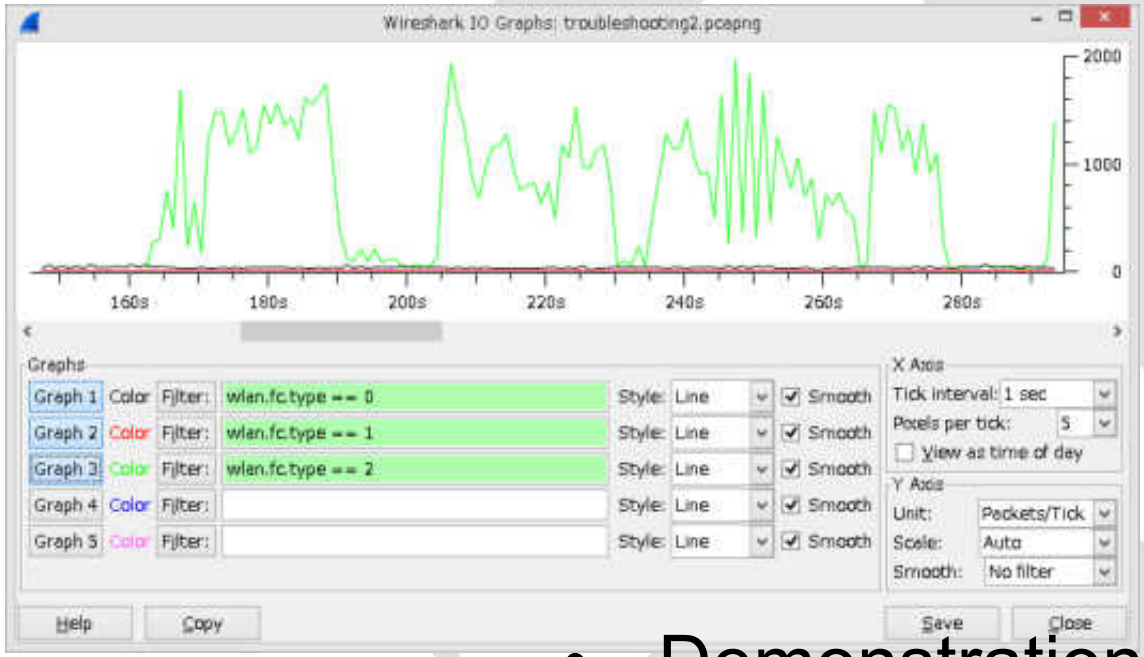
#11 Visualize (2) Frame type

- What type of IEEE802.11 frames in RF is important in analysis the compose of frame tells us the status of RF

Status	Management	Control	Data
IDLE	Many	Few	Few
BUSY (GOOD)	Few	Many same as Data	Many same as Control
BUSY (BAD)	Few	Many less than Data	Many more than Control
RTS/CTS (protect mode)	Few	Many more than Data	Many less than Control

#11 Visualize (2) Frame type

- Management frame wlan.fc.type==0
  Control Frame wlan.fc.type==1
  Data Frame (includes NULL) wlan.fc.type==2
- Statistics>
  IO Graph
  Y/X Axis ->
  packets / sec
- This time is
  BAD RF
  (many retry)



- Demonstration

#12 Visualize (3) management frames

- Management frame contains many good information for debugging and troubleshooting.
- Some AP sends important information in management frame.
- IEEE802.11e has QBSS (QoS Based Service Set) CCA (Clear Channel Assignment) information that contains the number of the connected station and utilization of the channel.

#12 Visualize (3) management frames

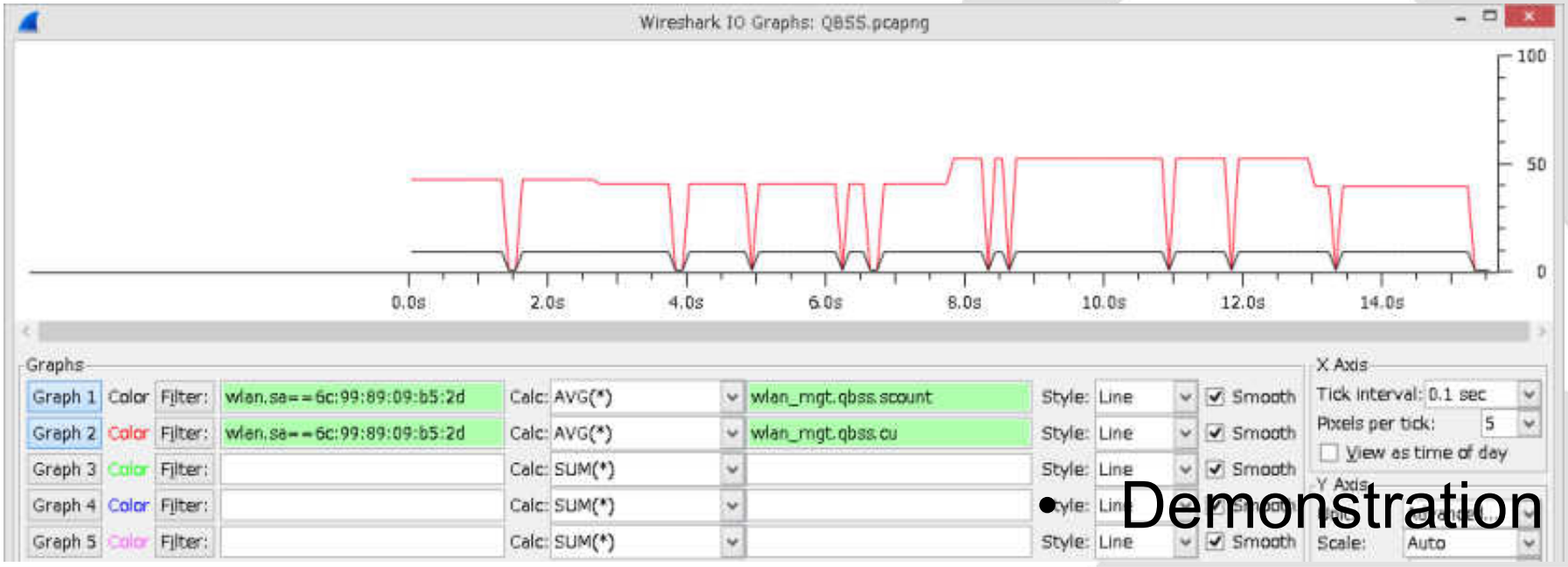
IEEE802.11e Beacon frame contains QBSS Tag
QBS Load Element CCA has the number of the Station and Channel Utilization

Station Count
wlan_mgt.qbss.scount
Channel Utilization
wlan_mgt.qbss.cu

```
▷ Radiotap Header v0, Length 26
▷ IEEE 802.11 Beacon frame, Flags: ........C
◢ IEEE 802.11 wireless LAN management frame
  ▷ Fixed parameters (12 bytes)
  ◢ Tagged parameters (244 bytes)
    ▷ Tag: SSID parameter set: Broadcast
    ▷ Tag: Supported Rates 12(B), 18, 24, 36, 48, 54
    ▷ Tag: Traffic Indication Map (TIM): DTIM 1 of 0
    ▷ Tag: Country Information: Country Code JP, Env
    ◢ Tag: QBSS Load Element 802.11e CCA Version
        Tag Number: QBSS Load Element (11)
        Tag length: 5
        QBSS Version: 2
        Station Count: 9
        Channel Utilization: 42 (16%)
        Available Admission Capabilities: 23437 (749
```

#12 Visualize (3) management frames Visualizing Station and Utilization

- Statistics>IO Graph and set Y Axis to advanced filtering specified AP and use AVG(*) and counting Station(Black) / Utilization (Red)



- Demonstration

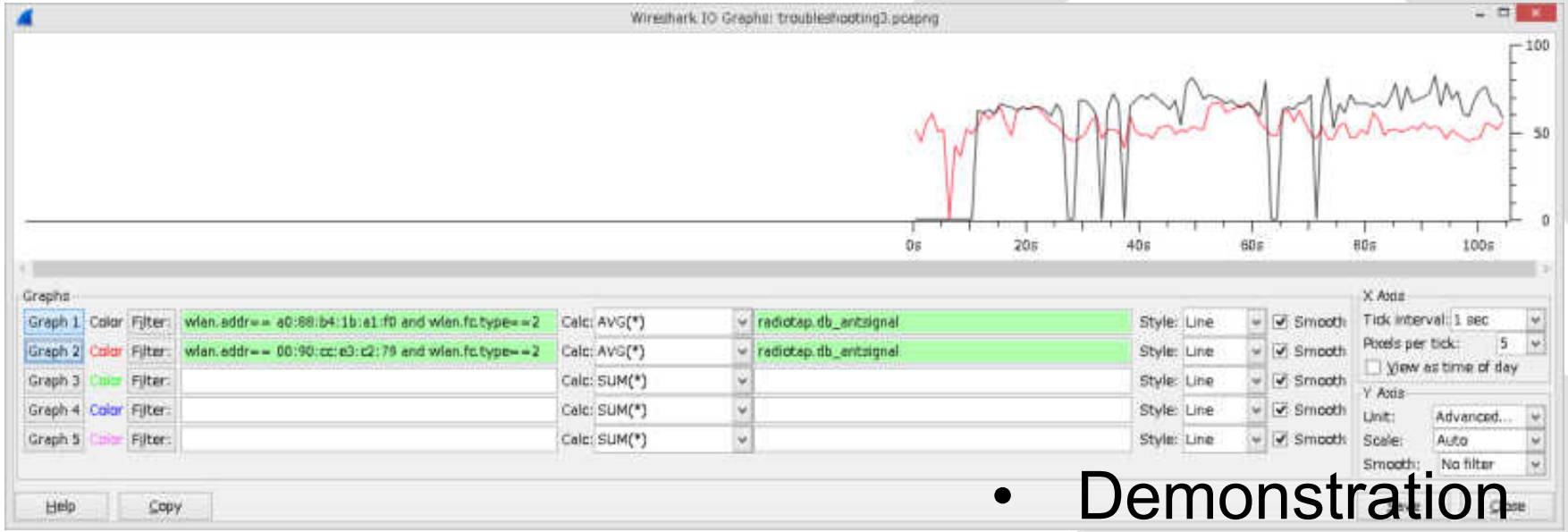
#13 Visualize (4) signal

- Signal / Noise ratio is useful, and good ratio is 20 (signal is 10 times louder than noise)
  20x log 10/1 = 20dB
- AirPcap collect signal info and display filter is radiotap.db_antsignal

dB	multiple
1	1.122018
2	1.258925
3	1.412538
4	1.584893
5	1.778279
6	1.995262
7	2.238721
8	2.511886
9	2.818383
10	3.162278
11	3.548134
12	3.981072
13	4.466836
14	5.011872
15	5.623413
16	6.309573
17	7.079458
18	7.943282
19	8.912509
20	10
21	11.22018
22	12.58925
23	14.12538
24	15.84893
25	17.78279
26	19.95262
27	22.38721
28	25.11886
29	28.18383
30	31.62278
31	35.48134
32	39.81072
33	44.66836
34	50.11872
35	56.23413
36	63.09573
37	70.79458
38	79.43282
39	89.12509
40	100
41	112.2018
42	125.8925
43	141.2538
44	158.4893
45	177.8279
46	199.5262
47	223.8721
48	251.1886
49	281.8383
50	316.2278

#13 Visualize (4) signal

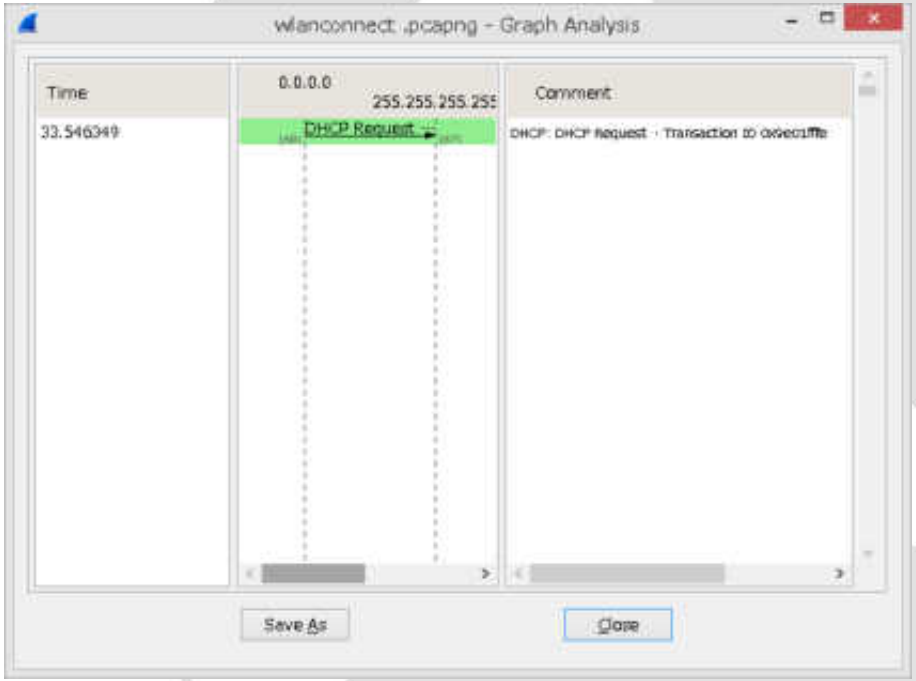
- Statistics > IO Graph and filter AP (Graph1) and filter Client (Graph2) and set Y axis to advanced, then counting AVG(*) of radiotap.db_antsignal



- Demonstration

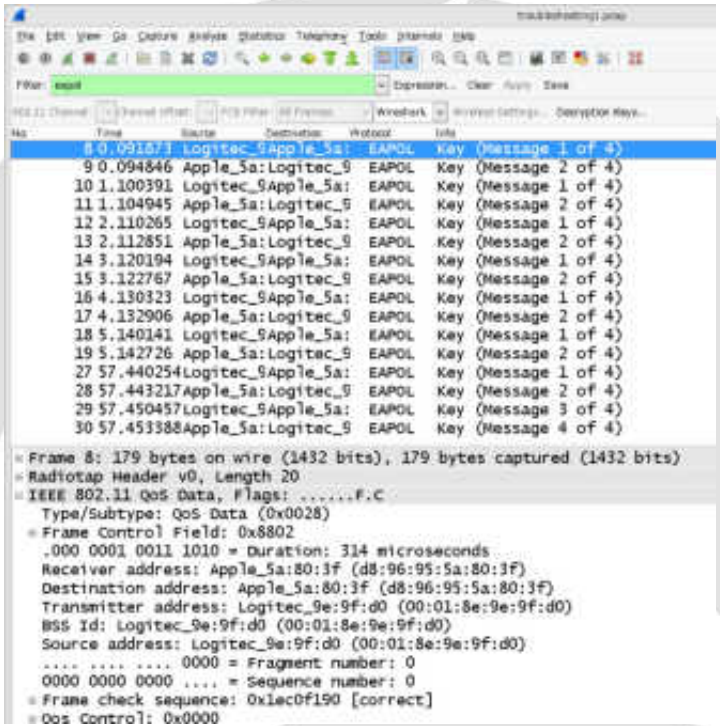
#14 Use flow graph

- If you need to draw Flow Graph under layer2 old version of Wireshark is good.
- Use Wireshark1.6 or older
- Statistics> Flow Graph



#15 Repetition of packets (iOS)

- Repetition of packets gives us the hint for debugging, troubleshooting.
- This packet contains the repetition that EAPOL(mes1/4) EAPOL(mes2/4) counts 6 times !
- The troubles lies in here.



#15 Repetition of packets (iOS)

- Wrong passphrase causes network error of EAPOL 4-way handshake.
- iOS tried 6 times.

No.	Time	Source	Destination	Protocol	Info
8	0.091873	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
9	0.094846	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)
10	1.100391	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
11	1.104945	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)
12	2.110265	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
13	2.112851	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)
14	3.120194	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
15	3.122767	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)
16	4.130323	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
17	4.132906	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)
18	5.140141	Logitec_9	Apple_5a:	EAPOL	Key (Message 1 of 4)
19	5.142726	Apple_5a:	Logitec_9	EAPOL	Key (Message 2 of 4)

- Demonstration

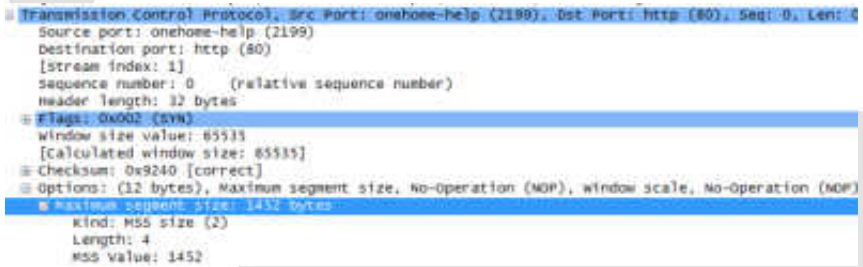
#16 Wireless Router's MTU/MSS

- Some user says they cannot see specific website. (ex. Google OK Yahoo NG)
- When MTU 1454 (default), we cannot see
  But MTU 1414, and we CAN SEE

EthernetII (14)	IP(20) DF=1 MF＝0 Offset＝	TCP (20)	MSS 1460

#16 Wireless Router’s MTU/MSS

- PPPoE(FTTH) is popular in Japan.
- NTT west’s MTU is 1454
  (Ethernet(1518)-EthernetHeader+FCS(14+4)-IP(20)-UDP(20)-L2TP(16)-PPPheader(2))
- NTT east optical fiber network’s MTU is 1438
  （MSS 1398）
- MSS value is determined in TCP negotiation, SYN/SYN-ACK packet in 3 way handshake



#16 Wireless Router's MTU/MSS

- MSS values are not the same in the debug.

FAIL

180.144.106.167	124.83.171.240	onehome-help > http [SYN] Seq=0 Win=65535 Len=0 MSS=1452 WS=1 SACK_
124.83.171.240	180.144.106.167	http > onehome-help [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460

TCP SYN
MSS=1452

TCP SYN/ACK with
MSS=1460

SUCCESS

180.146.55.43	203.216.243.211	winshadow-hd > http [SYN] Seq=0 Win=65535 Len=0 MSS=1415 WS=1 SACK_PER
203.216.243.211	180.146.55.43	http > winshadow-hd [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS

TCP SYN with
MSS=1415

TCP SYN/ACK with
MSS=1460

#17 WPS debugging

- Push button connection of WPS between wireless router and client fails in 40MHz mode, but it works in 20MHz mode.
- IEEE defines WPS but not in detail implements

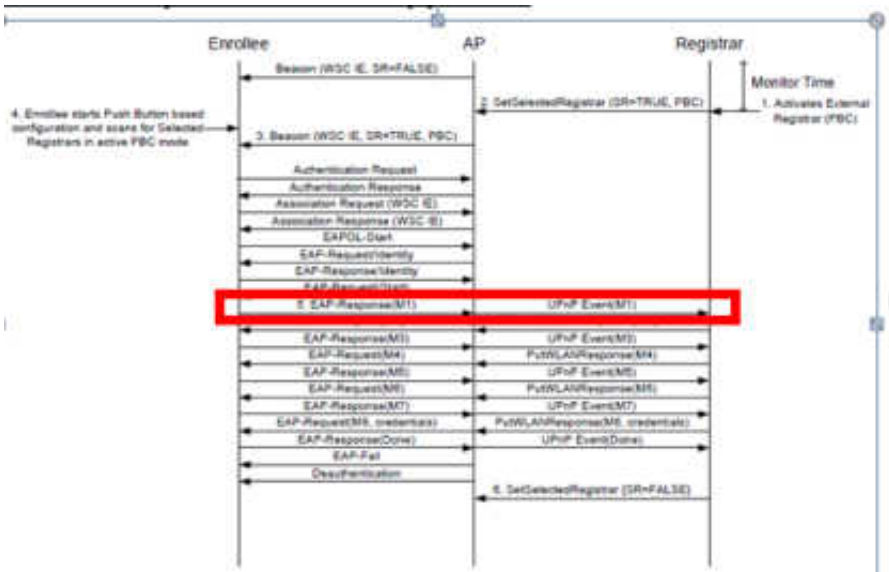
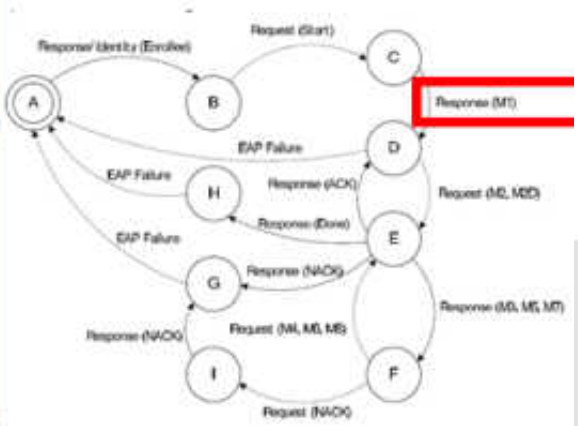
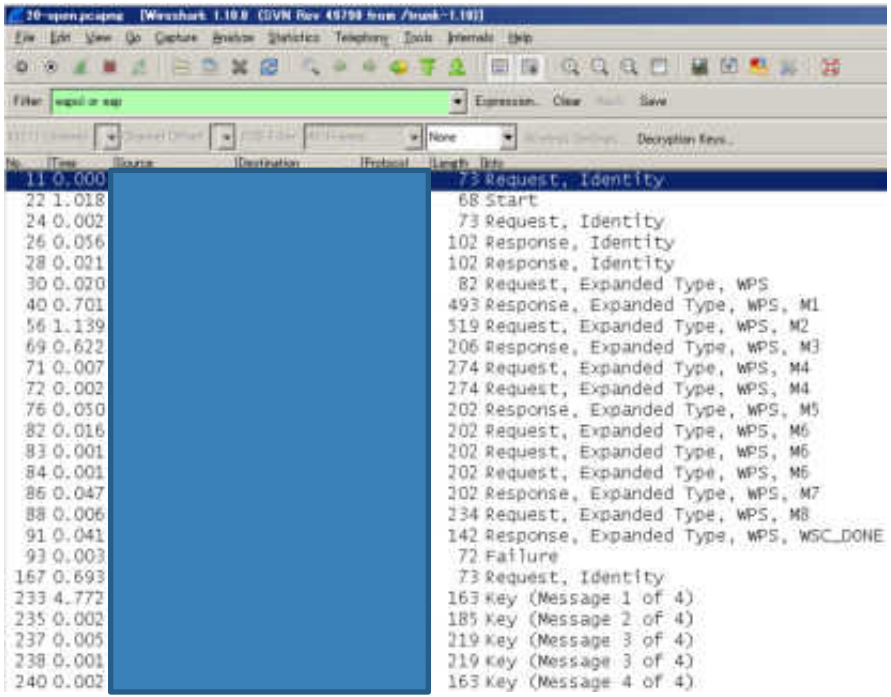


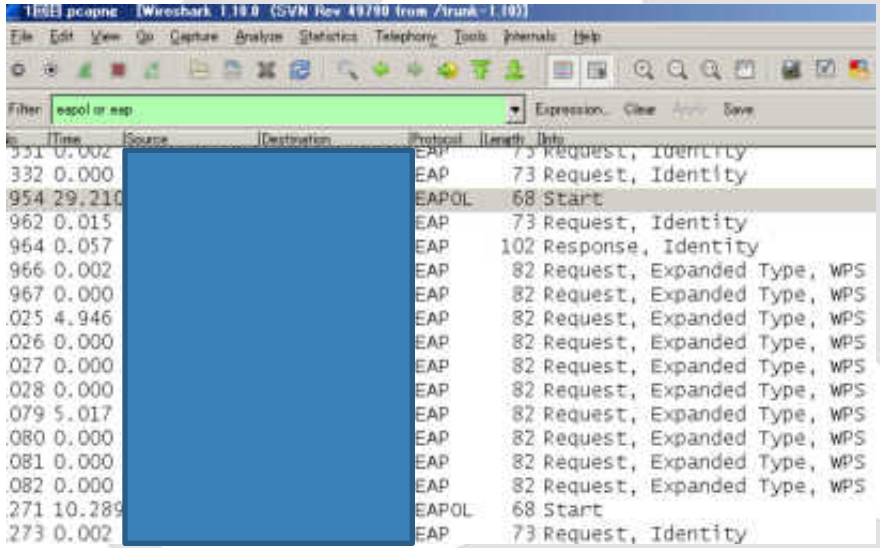
Figure 6 – PBC based setup – External Registrar trigger first



#17 WPS debugging

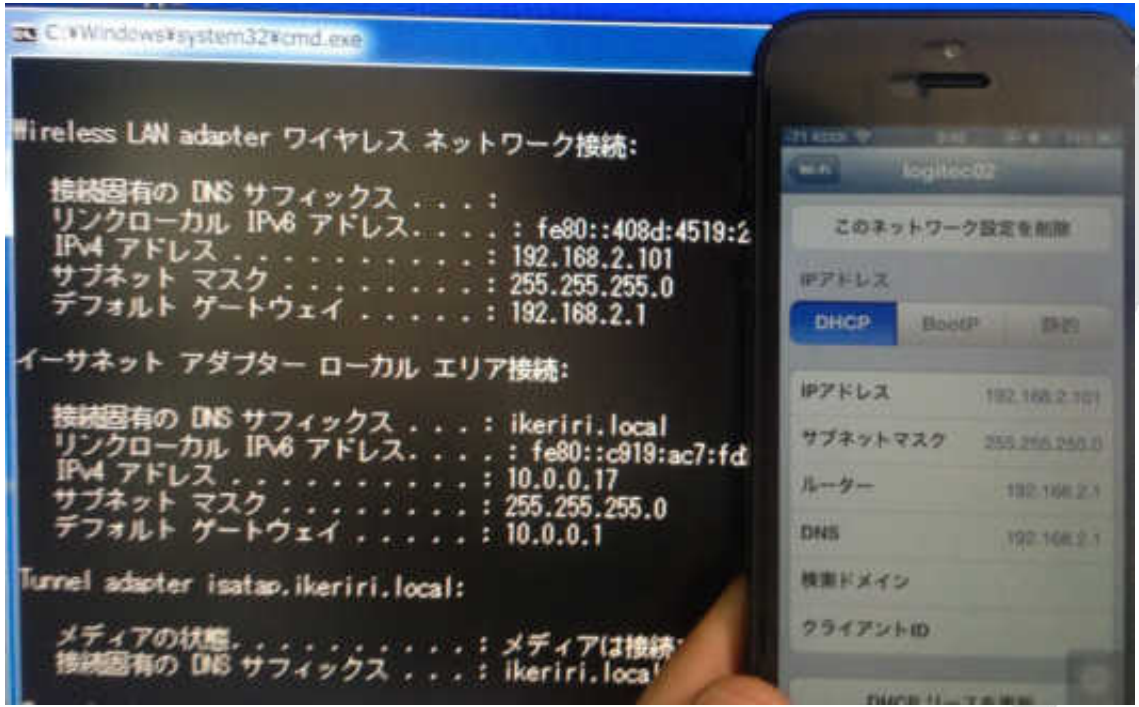
- AP sends Request Expand Type, but Client never response and stacked after ten times tries, so need to fix the one.





#18 wireless router’s DHCP issue

- The wireless router provides same IP address to another PC and smartphone in same SSID.



#18 wireless router's DHCP issue

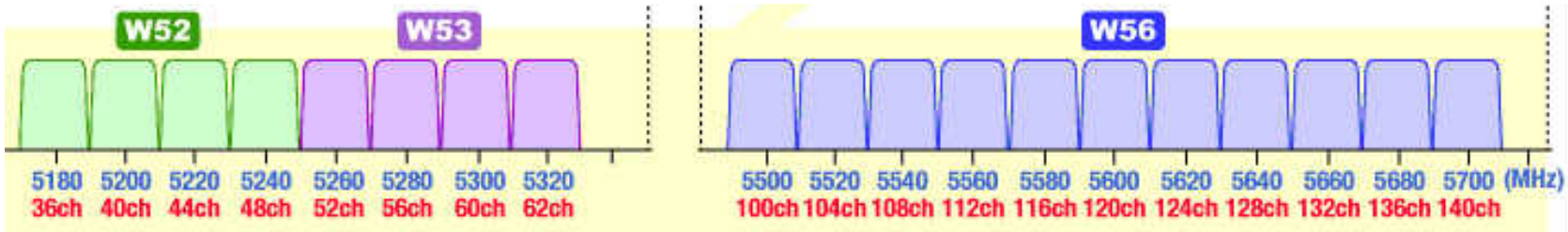
- The wireless router sends DHCP-ACK with 31536000 seconds (3650 days) of lease time

```
PPI version 0, 32 bytes
IEEE 802.11 QoS Data, Flags: .......F.C
Logical-Link Control
Internet Protocol Version 4, Src: 192.168.2.1 (192.1
User Datagram Protocol, Src Port: bootps (67), Dst P
Bootstrap Protocol
  Message type: Boot Reply (2)
  Hardware type: Ethernet
  Hardware address length: 6
  Hops: 0
  Transaction ID: 0x3eef299b
  Seconds elapsed: 0
⊞ Bootp flags: 0x0000 (Unicast)
  Client IP address: 0.0.0.0 (0.0.0.0)
  Your (client) IP address: 192.168.2.101 (192.168.2
  Next server IP address: 0.0.0.0 (0.0.0.0)
  Relay agent IP address: 0.0.0.0 (0.0.0.0)
  Client MAC address: [redacted]
  Client hardware address padding: 00000000000000000
  Server host name not given
  Boot file name not given
  Magic cookie: DHCP
⊞ Option: (53) DHCP Message Type
⊞ Option: (54) DHCP Server Identifier
⊟ Option: (51) IP Address Lease Time
    Length: 4
    IP Address Lease Time: (315360000s) 3650 days
⊞ Option: (1) Subnet Mask
⊞ Option: (3) Router
⊞ Option: (6) Domain Name Server
⊞ Option: (255) End
  Padding
```

- Both Windows and smartphone accepted, but smartphone changes lease time value into 90 days (selfishly)
- So IP duplicated.

#19 DFS debugging

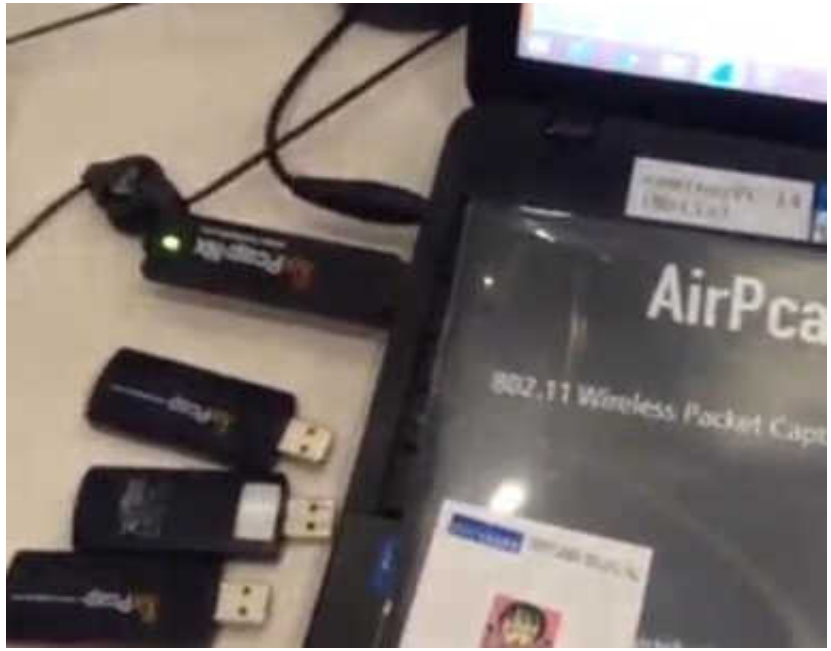
- There are tons of RF signals in Tokyo central. 2.4GHz bands are worthless, so companies tends to use 5GHz (W53, W54, W56 channel)
- W58 bandwidth is prohibited in Japanese law



- In case of indoor office, DFS comes and stack the communication 30 minutes, no fallback.
- Failed in automatically channel changing, so the customer have to re-connect manually.

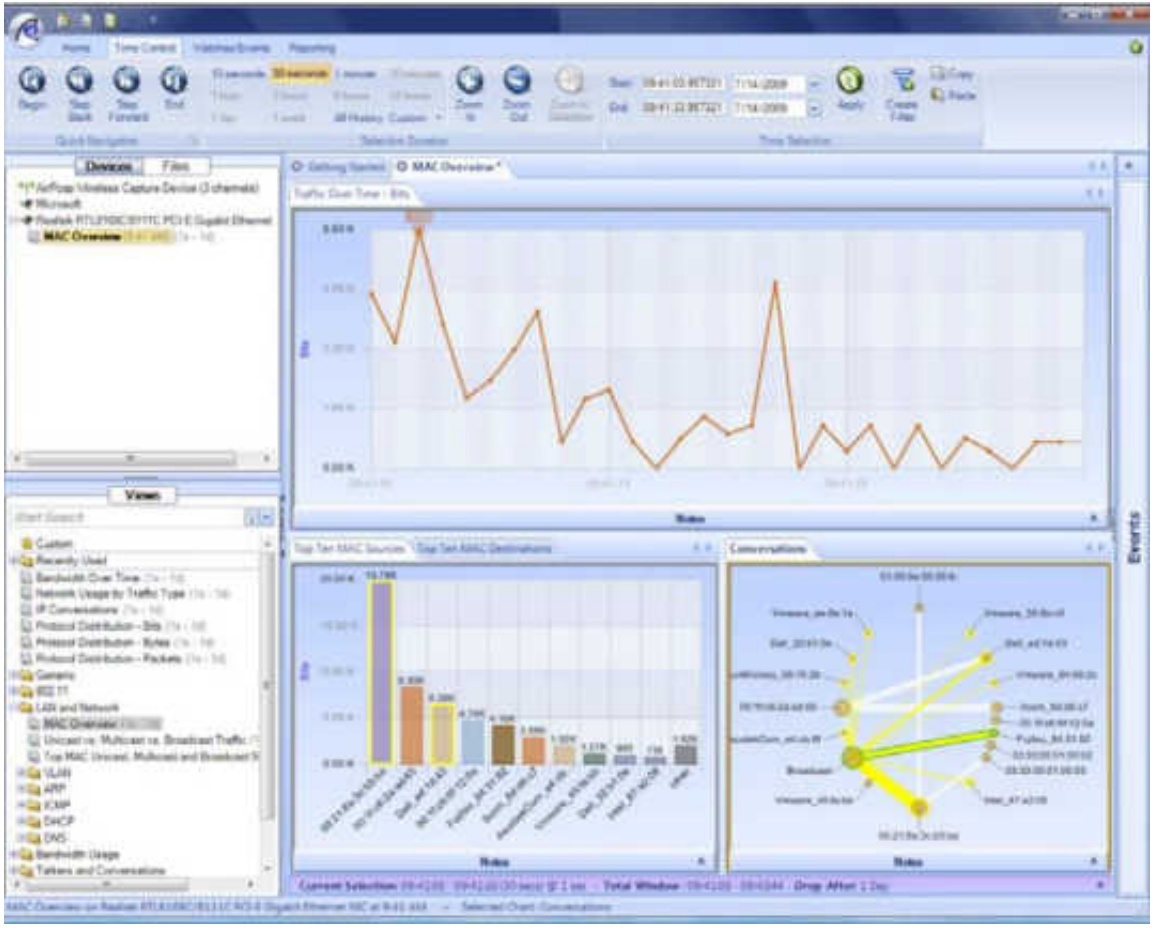
#19 DFS Debugging

- Using “tshark –i interface –b filesize:XXX –w filename.pcapng” and capture for long time.
- We uses 8 PCs with 8 AirPcapNX with 8 different CHs W53 (52 / 56 /60 / 64) and W56 (100 / 104 / 108 / 112) channel.
- Capture and wait like fishing, lurk in silence, until DFS comes (3 days …)



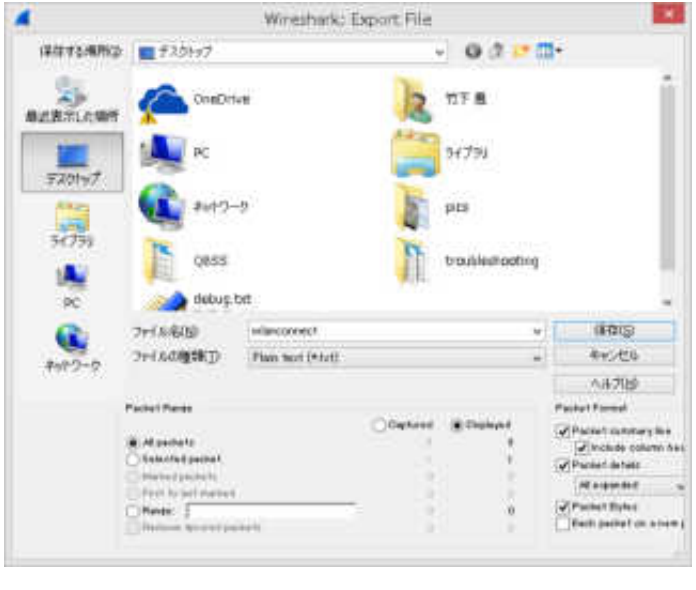
#19 DFS debugging

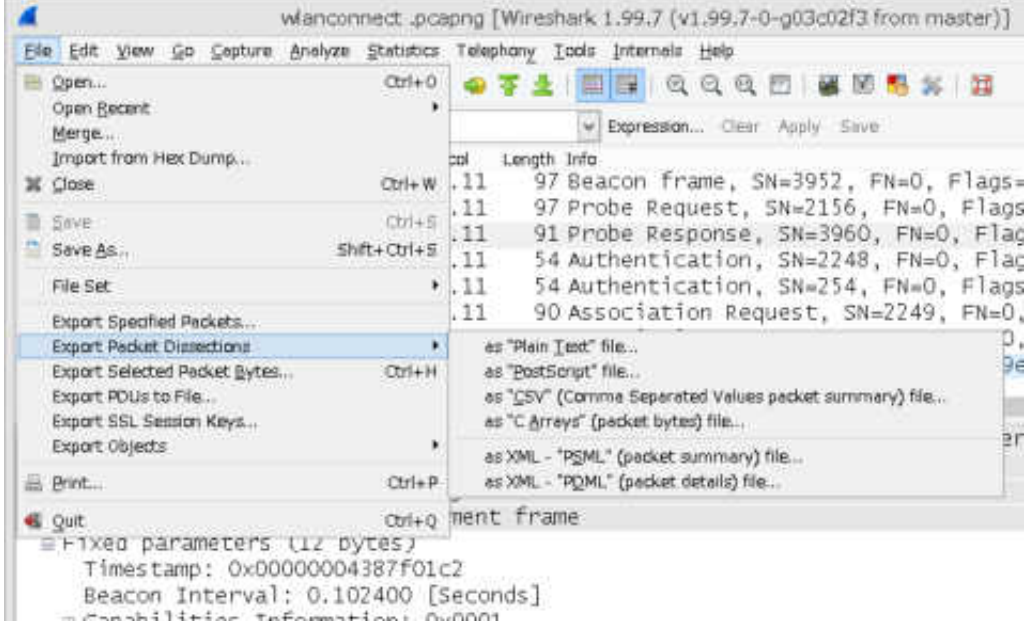
- If you have SteelCentral Packet Analyzer, you are lucky !
- If trace file size is 10GB, it is easy to create many graph, charts under 1 minutes



#19 DFS debugging

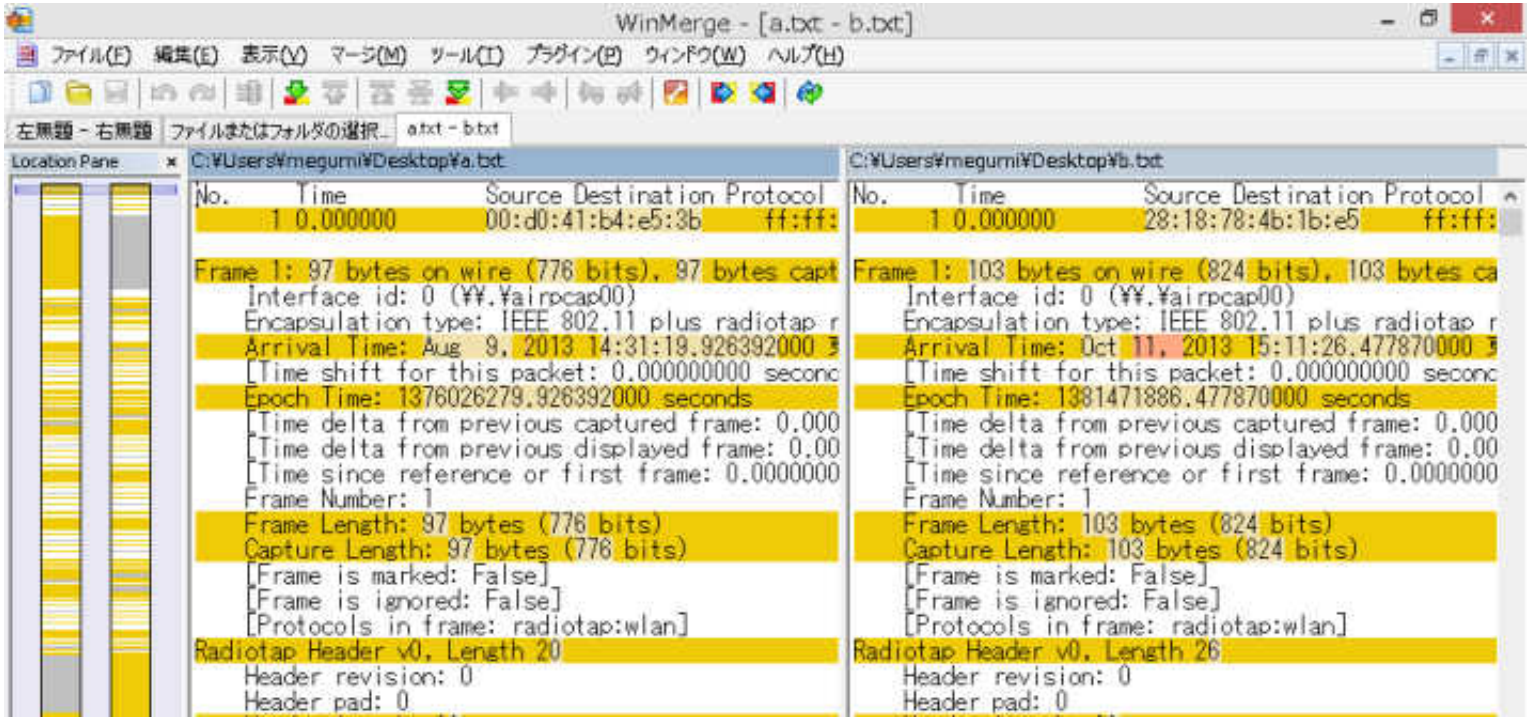
- In deep and complex debugging,
  we have to collect a lot of data,
  and have to combine a lot of data in text.
- File>Export Packet Dissections>as “Plain Text”





#19 DFS debugging

- Text based debug is the last resort.
- check a pair of the text translated trace file. Use the WinMerge

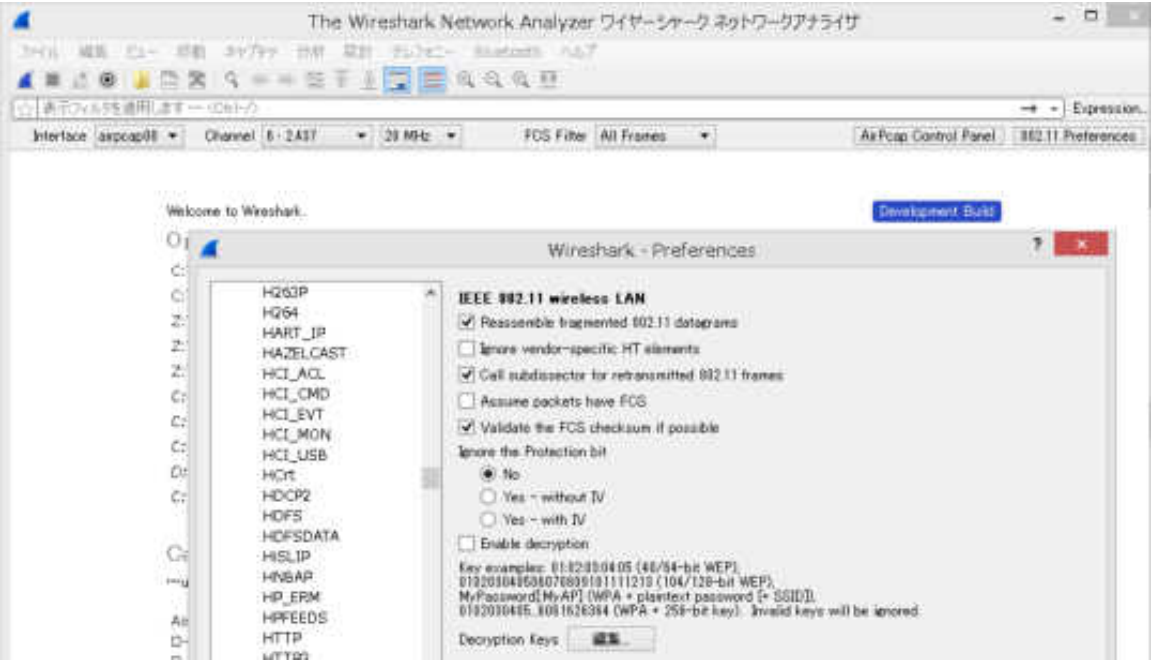


#19 DFS debugging

- We found strange management packet at result.
- Sometimes vender may not admit,
  After many months, the fixed patch was released.
- And the wrong detection bug causes the trouble of the stack and non-recovery problem.

#20 Use Wireshark !

- Wireshark help us finding many bugs and troubles in debugging and troubleshooting
- Use Wireshark !



Thank you！
どうもありがとうございます！