

Please form an orderly queue at the bar and you will be served

Sept 2026 © Jono Thompson Multithread Consultants Ltd



Jono Thompson



- Networking background started as a Cisco Engineer
- 24 years working in enterprise networking and data centres.
- MikroTik Consultant Since Dec 2014
- MikroTik Trainer since March 2017
- Ubiquiti Trainer since March 2019
- Certifications from other vendors.
- CWNA, CWSP, CWDP

Lin ITX
WIRELESS & NETWORKING EXPERTS

PRODUCTS | TRAINING | CONSULTANCY

UBIQUITI



ABOUT US:

- Technical sales team
- Professional support
- Extended warranties
- Trade accounts
- Credit accounts
- Live stock levels
- Trusted seller
- Training provider
- Next day delivery

WHAT WE OFFER:

- Wireless & networking products online
- MikroTik, Ubiquiti & TP-Link training
- Wireless & network consultancy

OUR TRAINERS

JONO
THOMPSON

Jono has over 20 years of experience in networking. He holds multiple MikroTik and Ubiquiti qualifications. He is also a fully Certified Ubiquiti and MikroTik training partner.

RON
TOUW

Ron has 40 years of experience in wireless and networking protocols. He holds multiple certifications in Ubiquiti, MikroTik, Ruckus, Meru, HP, Rohde & Schwarz and more.

Linitx.com | 01449 888000 | shop@linitx.com

Should you apply a queue to the WiFi?

Please form an orderly queue at the bar and you will be served



Common Misconception

- I regularly come across networks ‘designed’ to give a good Quality of Service by applying speed limits to different networks
- I have ‘fixed’ many networks by removing all the queues!

What people want….

- We want the very best WiFi for some of our users
- We want to reserve the WiFi for some of our users
- We want our network to have the best performance for some of our users

How do we do it?

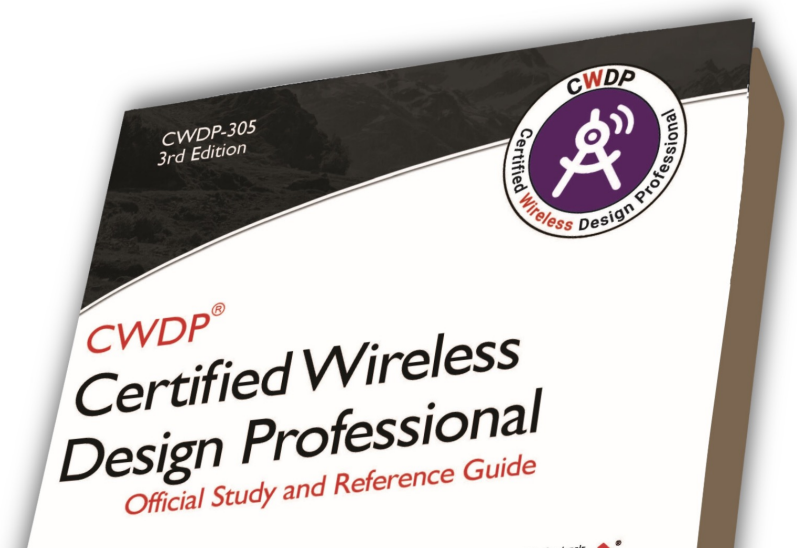
- Apply a very small limit to guest to prevent them using too much so there is more ‘WiFi’ for the higher proparity devices

- Appy a limit per user to prevent one user using all the WiFi so it will be fairer for everyone else

What I expected

- I expected a client who was limited to 2Mbps to use up more airtime than an unlimited client because they were sending their data really slowly
- So out of curiosity I thought I want to know why this is the case so I captured some packets and tried to work out what I could see
- I also wanted to see if I could put some numbers on the extra airtime

CWDP



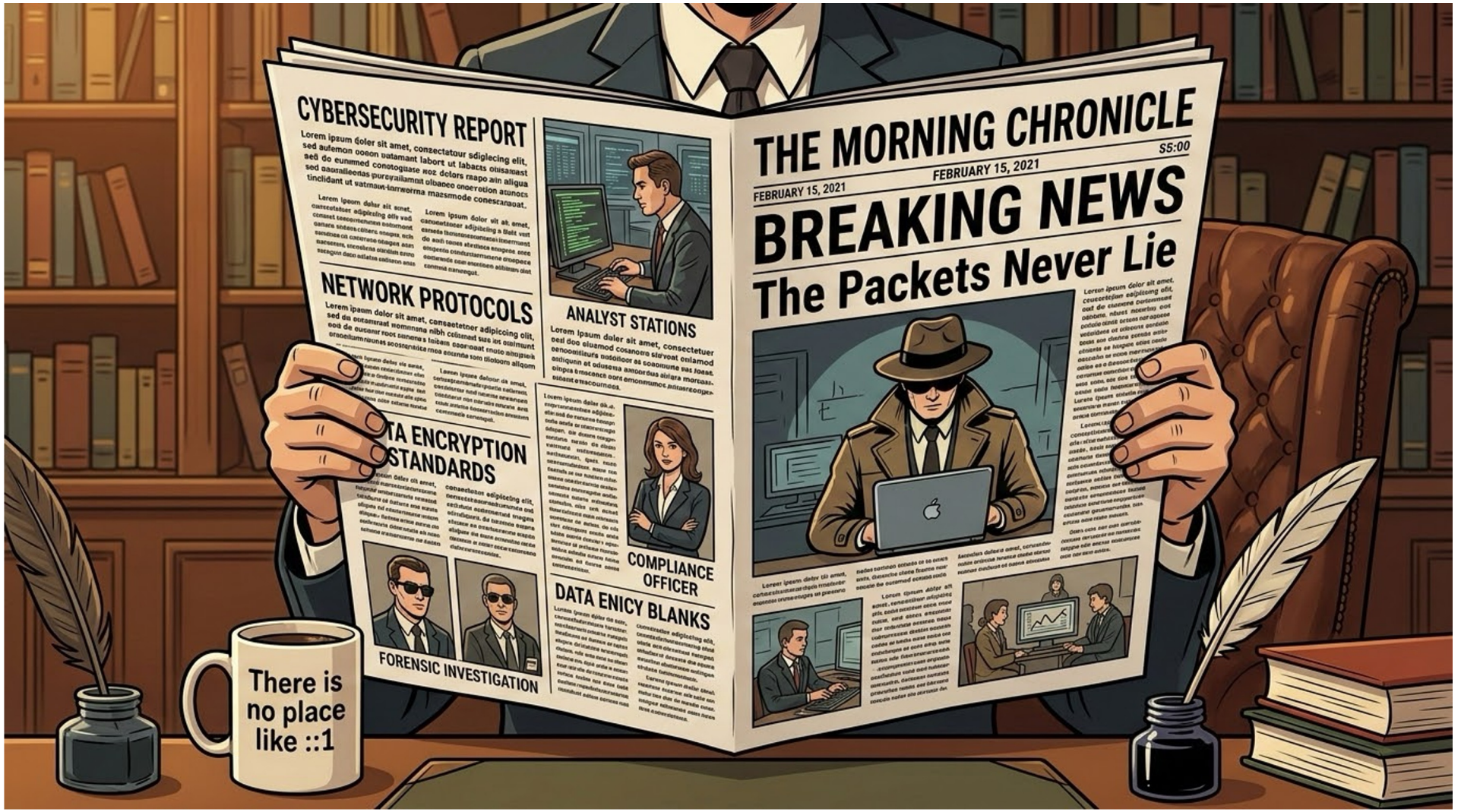
“Limit bandwidth per user: Most AP platforms support limiting the upstream and downstream bandwidth per client device. While this may have little practical effect on managing the airtime……”

CDWP Official Study and Reference Guide Pg 362 6.5: High-Density Applications

The Netflix Effect

- The Netflix Effect on Guest Wi-Fi
  Jim Palmer
  WLPC Phoenix 2019
- Looks at the effect of Queuing streaming services on the wired network
- https://www.youtube.com/watch?v=aChweomaqpM

Hopefully Old News!



Disclaimer

- These are taken form a very sterile lab environment which has no reflection on the real world!
- I am still learning packets so I could have made a massive mistake somewhere and not spotted it.

Presentation Objectives

- To see if can prove why its not always a good idea to Queue Wireless users
- To inspire you to go and look at the frames, anyone can do it you don’t need to be a magician to work it out!
- To just have a go and try to work it out… (and maybe make a fool when you get it wrong!)

Upload & Download Queues

- There are many different ways to apply a Speed Queue and many different queue types. This presentation only uses one type of queue.
- UniFi Wi-Fi Speed Limit which applies a speed limit per user at the Access Point

The Lab

- UniFi U7 Pro Triband AP.
- Mixture of laptops with 802.11ax and 802.11ac wireless cards
- Nano Pi running a webserver and downloaded a test 50Mb File
- Open SSID so if I wanted to decode data traffic I could
- 20MHz 5 GHz Channel 48 – no DFS, Low TX power
- Min Rates left as default
- Packets captured with Sidos Wave

Test 1

- Single AX laptop (Intel AX210) downloading 50Mb File
- WiFi Speed limits set at:
  - 2mb
  - 10mb
  - Unlimited

Time Taken

```
user@DESKTOP-FHQDOFK:~$ wget http://10.2.14.79/50MB.zip
--2026-07-16 15:37:16--  http://10.2.14.79/50MB.zip
Connecting to 10.2.14.79:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 52428800 (50M) [application/zip]
Saving to: '50MB.zip.39'

50MB.zip.39                 100%[=================================================>]  50.00M  23.0MB/s    in 2.2s

2026-07-16 15:37:18 (23.0 MB/s) - '50MB.zip.39' saved [52428800/52428800]
```

Unlimited
2.2 sec

```
user@DESKTOP-FHQDOFK:~$ wget http://10.2.14.79/50MB.zip
--2026-07-16 15:31:15--  http://10.2.14.79/50MB.zip
Connecting to 10.2.14.79:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 52428800 (50M) [application/zip]
Saving to: '50MB.zip.35'

50MB.zip.35                 100%[=================================================>]  50.00M  1.14MB/s    in 44s

2026-07-16 15:31:59 (1.14 MB/s) - '50MB.zip.35' saved [52428800/52428800]
```

10 Mbps Queue
44 sec

```
user@DESKTOP-FHQDOFK:~$ wget http://10.2.14.79/50MB.zip
--2026-07-16 15:07:35--  http://10.2.14.79/50MB.zip
Connecting to 10.2.14.79:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 52428800 (50M) [application/zip]
Saving to: '50MB.zip.32'

50MB.zip.32                 100%[=================================================>]  50.00M   234KB/s    in 3m 39s

2026-07-16 15:11:14 (234 KB/s) - '50MB.zip.32' saved [52428800/52428800]
```

2 Mbps Queue
3 min 39 sec

Data Rates

- Data Frames
  - 2mb Limit – 286.8
  - 10mb Limit – 286.8
  - Unlimited – 286.8
- Basic Rate 24.0

- So even when limited data frames still use high PHY rates ☺

Airtime usage

- To analyse the airtime I used Wireshark field `wlan_radio.duration`.
- However, I found that this was only populated for management and control frames
- Further reading suggested that for data frames could use `wlan_radio.aggregate.duration` However this was blank and I've no idea why

Data Frame airtime usage

- The only way I could see is to calculate it based on Data rate and frame size (length)

$$\text{Airtime } (\mu s) = \left(\frac{\text{Frame Length (Bytes)} \times 8}{\text{Data Rate (Mbps)}}\right) + \text{Preamble Overhead}$$

- Preamble Overhead to allow for the radiotap header of the packet. Have used an average of 40 μs for my calculation

(May not be accurate?!)

Wlan.duration

- Each frame (data, management and control) have a value for `wlan_duration`.
- This can not just be totalled to work out total airtime
- `wlan_duration` is the airtime being reserved though the rts/cts process for the remainder of the transmission not for the current frame.

Wlan.duration

- The `wlan.duration` values are for example:
- In a RTS Frame: `wlan.duration` is the time for

  CTS frame + data frame + block ACK + 3 SIFS intervals
- In a CTS Frame: `wlan.duration` is the time for

  data frame + block ACK + 2 SIFS intervals

Data Frame airtime usage

- Wireshark Filter:

```
wlan.addr == ac:45:ef:a4:77:02 &&
wlan.fc.type_subtype == 0x0028
```

- `wlan.addr == ac:45:ef:a4:77:02` – test laptop mac address
- `wlan.fc.type_subtype == 0x0028` – QoS Data frames

Data Frame airtime usage

- Exporting raw data from wireshark I thought was tedious so I learnt tshark CLI.

```
tshark -r input.pcap -Y " wlan.addr == ac:45:ef:a4:77:02 &&
wlan.fc.type_subtype == 0x0028" -T fields -E separator=, -E header=y -e
frame.number -e frame.len -e wlan_radio.data_rate >
client_data_raw_math_data.csv
```

- Exported data into excel, done the calculation from previous slide and added it all up

Why QoS Data Frames?

- All data payload should be only in QoS Data frames
- Because 802.11ac makes WMM/QoS mandatory
- The none QoS Data frames contain none of the user data payload and are an Overhead – maybe we can look at these later!

Management and Control Frames

- Wireshark Filter:

```
wlan.addr == ac:45:ef:a4:77:02 &&
wlan.fc.type != 2
```

- `wlan.addr == ac:45:ef:a4:77:02` – test laptop mac address
- `wlan.fc.type != 2` – Anything which is not a Data frame

Management and Control Frames

- Non data frames is easier as Wireshark already has the frame duration

```
tshark -r your_file.pcap -Y "wlan.addr == ac:45:ef:a4:77:02 && wlan.fc.type != 2" -T
  fields -e frame.number -e wlan_radio.duration -E separator=, -E header=y 2>/dev/null
  > client_nondata_radio_durations.csv
```

- wlan.addr == ac:45:ef:a4:77:02 – test laptop mac address
- wlan.fc.type != 2 – Anything which is not a Data frame

- Exported data into excel and added it all up

Test 1

- Download of 50Mb file
- Total airtime is QoS Data Frames & Management & Control frames airtime

	Download time	Airtime
2mb limit	3 min 39 sec	8131.74 ms
10mb limit	44 sec	7108.05 ms
unlimited	2.2 sec	2734.08 ms

- The unlimited download uses approx. 66% LESS airtime
- Why?

Retries

- When applying queues on the wired clients it's common to see some retries while the TCP protocol adapts and slows down.
- Retries are going to use up valuable airtime
- The Wireless medium can also cause retries
- Is the reason for the reduction in airtime usage because of less retries?

TCP Retries

- TCP is a connection orientated protocol so knows when packets don’t get through so will resend them
- Packet retries are wasting valuable airtime repeating
- Queuing traffic often creates TCP retries while the application adjusts to the speed limit

```
tshark -r your_file.pcap -Y "wlan.addr == ac:45:ef:a4:77:02 &&
  (tcp.analysis.retransmission || tcp.analysis.duplicate_ack)" | wc -l
```

MAC Layer Retries

- The wireless medium can cause retries
- Again these are using up airtime!

```
tshark -r your_file.pcap -Y "wlan.addr == ac:45:ef:a4:77:02 && wlan.fc.retry
  == 1" | wc -l
```

TCP and MAC Retries

	MAC Retries			TCP Retries		
	Number Packets	Airtime	% of total airtime	Number Packets	Airtime	% of total airtime
2mb Limit	6588	376.60 ms	3.84%	11450	655.56 ms	9.27 %
10 Mb Limit	567	47.20 ms	0.38%	3999	205.01 ms	2.90 %
Unlimited	848	73.53 ms	2.52%	149	12.61 ms	0.18 %

- Yes some airtime saved but not whole picture!

Total Airtime Split

- The total airtime was a total of
  - QoS Data Frames
  - All Management Frames
  - All Control Frames

QoS Data Frames

- 1x 802.11ax Client downloading 50mb file

Speed Limit	Number QoS Data frame	Airtime used by QoS data frames	% of Total Airtime
2mb	59404	4466.34 ms	54.92 %
10mb	55577	3988.12 ms	56.11 %
unlimited	31329	2647.78 ms	96.84 %

- Limited Downloads only 50% of the airtime is used by Data
- Unlimited download most of airtime is used by QoS Data

Is that the whole picture?

- We only looked at QoS Data frames
- There are other frames!

- What about the management and control frames
- What about data frames which are not QoS Data frames?

Data Frames that are not QoS Data frames

- Yes, there some frames which are data frame but not QoS Data frame. There were minimal so going to skip over these as its less than ¼ %

Speed Limit	Number None QoS Data frame	Airtime of none QoS data frames	% of Total Airtime
2mb	78	11.8ms	0.17 %
10mb	28	8.4ms	0.12 %
unlimited	16	2.21ms	0.03 %

Management and Control Frames

- 1x 802.11ax Client downloading 50mb file

Speed Limit	Non Data frame	Airtime used by Non data frames	% of total Airtime used
2Mbps	112216	3653.60 ms	51.69 %
10Mbps	95274	3111.53 ms	44.02 %
unlimited	2337	84.09 ms	1.19 %

- Unlimited download uses 3569.51 ms less airtime than a Queue of 2Mbps. So its using 97% less airtime on ‘overhead’
- this sent at basic rate testing was at 24Mbps, what if that was 6Mbps!

Management and Control Frames

- Can we learn anything from the Management and Control Frames?

- Looking at the types of Management and Control Frames, most are either ACK/Block ACK or RTS and CTS Frames lets break it down

Management and Control frames

- wlan.fc.type_subtype == 0x1b || wlan.fc.type_subtype == 0x1c
- wlan.fc.type_subtype == 0x001d || wlan.fc.type_subtype == 0x0019"

Speed Limit	RTS/CTS			ACK/Block ACK		
	Number of Frames	Airtime used	% of total airtime	Number of Frames	Airtime used	% of total airtime
2Mbps	54666	1605.23 ms	22.71 %	57377	2039.48 ms	28.85 %
10Mbps	39808	1127.66 ms	15.95 %	55427	1981.76 ms	28.04 %
Unlimited	1019	34.76 ms	0.49 %	1311	48.94 ms	0.69 %

None Data Frames

- Why is there such a reduction in RTS/CTS frames and ACK/Block ACK frames when the WiFi is unlimited…
- Management and Control frames are sent at basic rate (ie very slowly!)
- This took me back to what I learnt on my CWNA

A-MPDU

- 802.11n and newer supports frame aggregation.
- 802.11ac all data frames are sent as A-MPDUs

- What I needed to do is find out how many MPDU were in each A-MPDU

A-MPDU

- Each MPDU which is part of a A-MPDU has the same A-MPDU ID which can be viewed in Wireshark.

- Exporting all the A-MPDU IDs and counting how MPDUs have the same AMPDU ID gives the number of MPDU aggerated

TCP ACK

- TCP ACK frames are still QoS data frames but do not have an A-MPDU ID (actually all had ID of 0 which was a problem!)
- These can not be aggerated as this would cause issues for the TCP protocol believing there is a delay and slowing it down
- These have been excluded from AMPDU average calculations and categorise separately

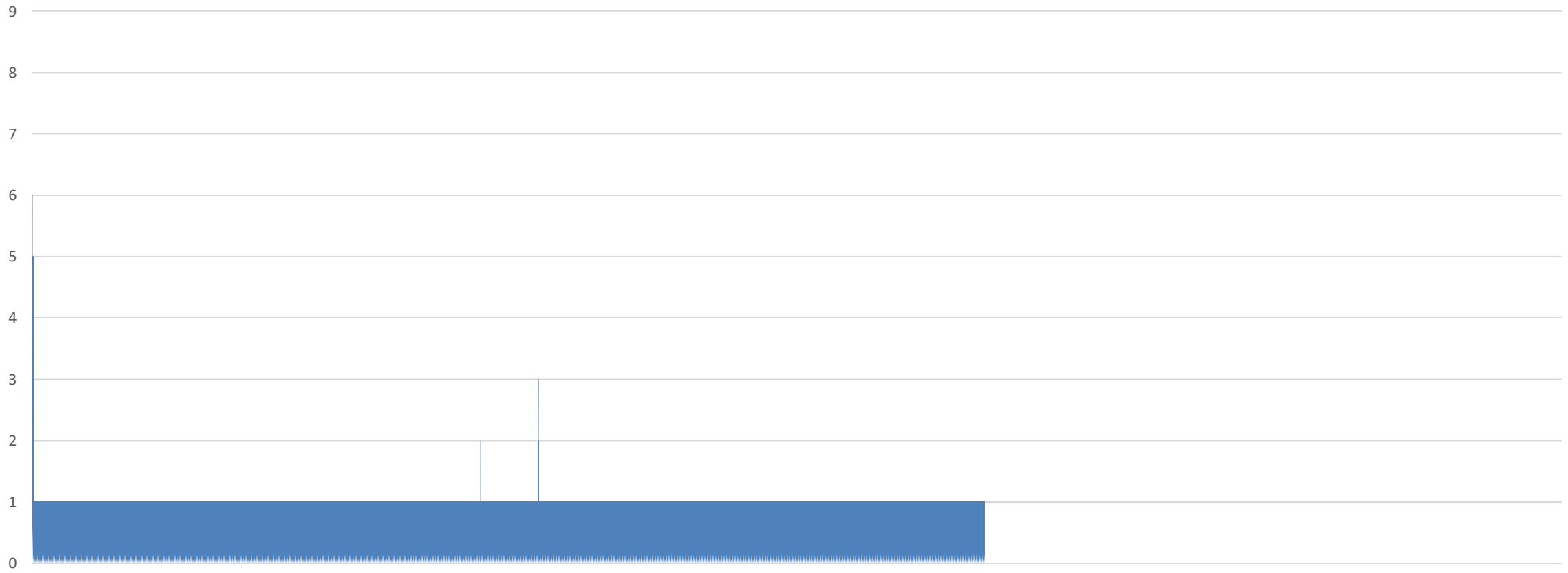
A-MPDU excluding TCP ACK

- Wireshark filter

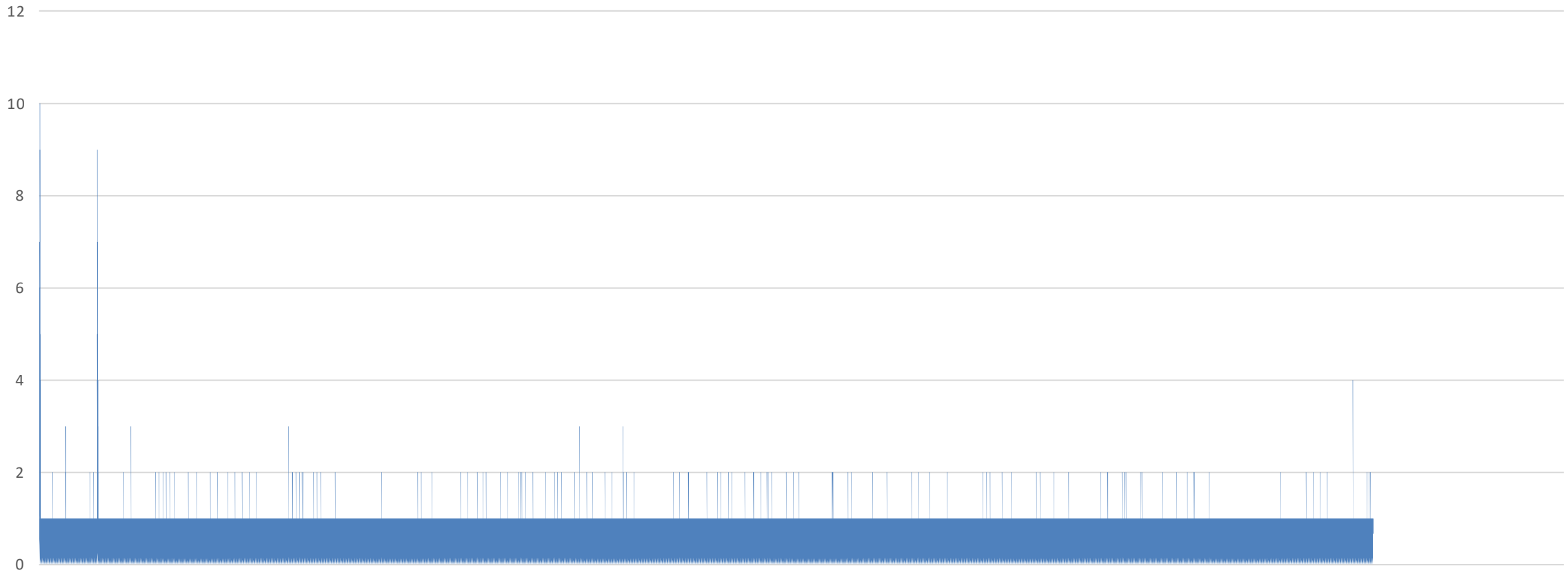
```
wlan.addr == ac:45:ef:a4:77:02 &&
wlan.fc.type_subtype == 0x0028 && tcp.len >
0
```

- `Tcp.len > 0` filters out tcp packets with zero payload which are pure tcp ack's

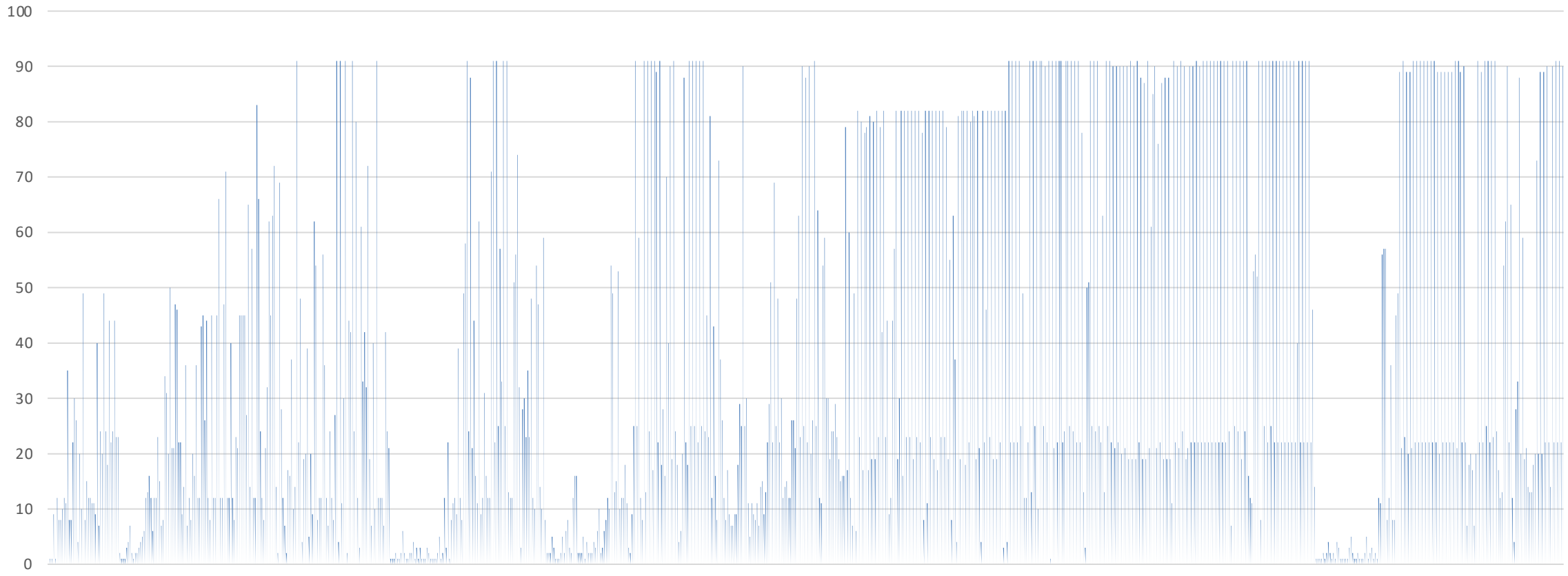
MPDU’s per A-MDPU – 2Mb Queue



MPDU’s per A-MDPU – 2Mb Queue



MPDU’s per A-MDPU – Unlimited



A-MPDU Summary

	Total A-MPDU	Avg MPDU	Max MPDU	TCP ACK
2mb Limit	37541	1.0	8	21631
10mb Limit	35794	1.0	10	19489
Unlimited	872	34.9	91	1019

A-MPDU

- When these very slow limits are applied the access point is unable to put more than one or two frames into a A-MPDU because it can’t wait any longer before it needs to send the data

A-MPDU

- Each A-MPDU needs the medium clearing to transmit so needs a RTS/CTS exchange
- Each A-MPDU needs Acknowledging so needs a block ACK

- This is all ‘wasted’ airtime as its not used for data transmissions

A-MPDU

- When the traffic is sent unlimited and we let the protocol work to full extent

- There are larger A-MPDU, which means Less total A-MPDU which means less ‘wasted’ airtime clearing the medium and confirming the traffic has arrived

A-MPDU

- Transmitting the data as a ‘super’ block frees up huge amount of air time
- Each ’super’ block only need one RTS/CTS exchange
- Each ‘super’ block only needs one Block ACK

- The RTS/CTS and the Block ACK are all sent as basic rate

What Next?

- 1 wireless client is not a very good sample
- Next test
- 1 x ac client – do older clients have the same effect
- 4 x mixed clients (2 x 802.11ax and 2 x 802.11ac)

Test 2

- Download of 50Mb file with a single 802.11ac client. Compare to the 802.11ax client

Speed Limit	Client	Download time	Airtime	% airtime QoS Data
2mb limit	802.11ax	3 min 39 sec	8131.74 ms	54.9 %
	802.11ac	3 min 39 sec	7068.77 ms	74.8 %
10mb limit	802.11ax	44 sec	7108.05 ms	56.1 %
	802.11ac	44 sec	7032.54 ms	74.9 %
unlimited	802.11ax	2.2 sec	2734.08 ms	96.8 %
	802.11ac	4.3 sec	5317.21 ms	98.06 %

- Not as much of a difference but the ac client is more efficient at lower speeds…

Test 2 A-MPDU

- Download of 50Mb file with a single 802.11ac client. Compare to the 802.11ax client

Speed Limit	Client	Total A-MPDU	Avg MPDU	Max MPDU
2mb limit	802.11ax	37541	1.0	8
	802.11ac	36066	1.0	11
10mb limit	802.11ax	35794	1.0	10
	802.11ac	35481	1.0	13
unlimited	802.11ax	872	34.9	91
	802.11ac	2356	15.6	28

- Not as much of a difference

Test 2 Management and Control frames

- Download of 50Mb file with a single 802.11ac client. Compare to the 802.11ax client

Speed Limit	Client	RTS/CTS			ACK/Block ACK		
		Number of Frames	Airtime used	% of total airtime	Number of Frames	Airtime used	% of total airtime
2Mbps	802.11ax	54666	1605.23 ms	22.71 %	57377	2039.48 ms	28.85 %
	802.11ac	473	24.6 ms	0.35 %	57163	1743.36 ms	24.7 %
10Mbps	802.11ax	39808	1127.66 ms	15.95 %	55427	1981.76 ms	28.04 %
	802.11ac	1716	85.73 ms	1.21 %	54852	1676.62 ms	23.7 %
Unlimited	802.11ax	1019	34.76 ms	0.49 %	1311	48.94 ms	0.69 %
	802.11ac	83	4.32 ms	0.06 %	3100	98.16 ms	1.38 %

802.11ac RTS/CTS

- Beacon Frame from the U7 Pro

```
Tag: HT Operation
   Tag Number: HT Operation (61)
   Tag length: 22
   Primary Channel: 48
   HT Operation Information Subset (1 of 3): 0x00
   HT Operation Information Subset (2 of 3): 0x0004
      .... .... .... ..00 = HT Protection: No protection mode (0x0)
      .... .... .... .1.. = Non-greenfield STAs present: One or more associated STAs are not greenfield capable
      .... .... .... 0... = Reserved: 0x0
      .... .... ...0 .... = OBSS non-HT STAs present: Use of protection for non-HT STAs by overlapping BSSs is not needed
      ...0 0000 000. .... = Channel Center Frequency Segment 2: 0
      000. .... .... .... = Reserved: 0x0
```

- As there's no legacy 802a/g devices the AP has set HT Protection: 0x0 (No Protection Required)
- RTS/CTS not needed for legacy clients

802.11ac RTS/CTS

- RTS/CTS is left optional to the AP
- As RTS/CTS though its benefits does add an overhead from this test around 25%
- RTS/CTS is does help solve problems caused by hidden nodes
- Stations will/might use some metrics to decide when to use RTS/CTS eg high retries
- Stations could choose to use CTS to self which only needs 1 frame not 2

802.11ax RTS/CTS

- 802.11ax as it's a different phy mode doesn't use VHT Protection: to decide to use RTS/CTS
- 802.11ac clients can't decode the HE preamble or MAC layer headers inside 802.11ax clients so can't read the Duration field to set their NAV timer
- 802.11ax sending RTS/CTS before A-MPDU frames means legacy clients can decode the duration field and set their NAV Timers.

802.11ax RTS/CTS

- 802.11ax uses a longer symbol duration and a much larger single A-MPDU so there is higher airtime cost needed for a retransmission.
- As we have seen this much larger A-MPDU the use of RTS/CTS takes up very little additional time so its ‘cheep airtime insurance’
- The higher basic rate of 24Mbps uses less airtime

Test 3

- Download of 50Mb file with 4 simulatious clients (2x ac, 2x ax)

Speed Limit	Client	Download time	Airtime	% airtime QoS Data
2mb limit	Client 1 (ax)	3 min 39 sec	8131.74 ms	55.8 %
	Client 2 (ax)	3 min 39 sec	8801.02 ms	56.9 %
	Client 3 (ac)	3 min 39 sec	7613.36 ms	75.2 %
	Client 4 (ac)	3 min 39 sec	7083.61 ms	74.7 %
10mb limit	Client 1 (ax)	44 sec	2294.25 ms	41.0 %
	Client 2 (ax)	44 sec	2345.36 ms	44.0 %
	Client 3 (ac)	44 sec	6160.56 ms	85.5 %
	Client 4 (ac)	44 sec	5996.10 ms	85.2 %
unlimited	Client 1 (ax)	10 sec	3913.75 ms	97.3 %
	Client 2 (ax)	9.9 sec	3366.10 ms	97.2 %
	Client 3 (ac)	12 sec	5122.04 ms	97.6 %
	Client 4 (ac)	12 sec	4878.94 ms	98.1 %

Test 3 A-MPDU

- Download of 50Mb file with 4 simulatious clients (2x ac, 2x ax)

Speed Limit	Client	Total A-MPDU	Avg MPDU	Max MPDU
2mb limit	Client 1 (ax)	36996	1.0	8
	Client 2 (ax)	37458	1.0	8
	Client 3 (ac)	37636	1.0	22
	Client 4 (ac)	36343	1.0	9
10mb limit	Client 1 (ax)	2405	1.4	15
	Client 2 (ax)	2354	1.5	18
	Client 3 (ac)	51270	2.6	17
	Client 4 (ac)	50871	2.6	16
unlimited	Client 1 (ax)	958	40.3	91
	Client 2 (ax)	876	42.3	91
	Client 3 (ac)	2110	18.5	27
	Client 4 (ac)	1924	20.3	28

Test 3 Management and Control frames

Speed Limit	Client	RTS/CTS			ACK/Block ACK		
		Number of Frames	Airtime used	% of total airtime	Number of Frames	Airtime used	% of total airtime
2Mbps	Client 1 (ax)	50080	1514.95 ms	21.4 %	57277	2034.16 ms	28.8 %
	Client 2 (ax)	53530	1732.71 ms	24.5 %	56848	2021.49 ms	28.6 %
	Client 3 (ac)	2357	117.98 ms	1.7 %	57270	1747.34 ms	24.7 %
	Client 4 (ac)	776	39.58 ms	0.6 %	57396	1751.01 ms	24.8 %
10Mbps	Client 1 (ax)	28752	826.75 ms	11.7 %	17222	524.74 ms	7.4 %
	Client 2 (ax)	28395	821.34 ms	11.6 %	15895	481.21 ms	6.8 %
	Client 3 (ac)	2123	82.24 ms	1.6 %	26350	797.58 ms	11.3 %
	Client 4 (ac)	2013	74.12 ms	1.1 %	26533	802.5 ms	11.4 %
Unlimited	Client 1 (ax)	1494	53.23 ms	0.8 %	1361	49.92 ms	0.7 %
	Client 2 (ax)	1318	46.02 ms	0.7 %	1318	47.84 ms	0.7 %
	Client 3 (ac)	192	9.12 ms	0.13 %	2970	94.04 ms	1.3 %
	Client 4 (ac)	179	9.02 ms	0.13 %	2728	86.05 ms	1.2 %

iOS 27

- Could small queues cause Connectivity Assist in iOS 27 cause devices to not even use the WiFi?

Conclusions

It is not quite what I expected to find but…

- Though unlimited clients do use less airtime for data frames
- Applying a queue prevents the efficiencies which are built into the newer protocols from working in this case frame aggregation and that's what gives the biggest airtime saving
- This is worse (well on UniFi) with newer wireless protocols of 802.11ax

Conclusions

- Leave the WiFi alone and trust the channel contention process to make it fair

- If you need to break the rules, understand impact its going to have

Conclusions

- Even though in training its very clear the benefits of RTS/CTS to the hidden node effect
- However it seems that it might not get used all the time due to the extra overhead the RTS/CTS frame have on airtime!

Should you apply a queue to the wifi?

NO

What's next?

- Having done this I then thought what about looking at the airtime reserved by RTS/CTS
  - Need to find which RTS gets a CTS and only those
- Look at how this affects Netflix or other streaming
  - How to lab up
- More client or simulate more clients somehow
  - 4 laptops pressing enter at the same time was my limit
- Other queuing methords

Hope you got served and thank you for Listening

Where to find me….

LinITX Blog

Linkedin