

No IP
Broadcast
WLC B
Null MAC
AP wired C
AP wired I2
AP wired B
AP wired A
AP wired I3

stateless_ha_nn1.pcapng

DHCP Discover

Transaction ID	0x0000020c
IP	0.0.0.0
Client MAC address	84:b2:61:21:b7:2c
Client MAC address	84:b2:61:21:b7:2c
Hostname	AP84b2.6121.b72c

DHCP Offer

Transaction ID	0x0000020c
IP	10.0.40.104
Client MAC address	84:b2:61:21:b7:2c

DHCP Request

Transaction ID	0x0000020c
IP	0.0.0.0
Client MAC address	84:b2:61:21:b7:2c
Client MAC address	84:b2:61:21:b7:2c
Hostname	AP84b2.6121.b72c

DHCP ACK

Transaction ID	0x0000020c
IP	10.0.40.104
Client MAC address	84:b2:61:21:b7:2c

Data (0x0020)

Dst	ff:ff:ff:ff:ff:ff
Src	08:cc:a7:86:55:20
BSS	00:00:00:00:00:00
Seq	0

DHCP Discover

Transaction ID	0x000013e0
IP	0.0.0.0
Client MAC address	84:b2:61:0e:0c:18
Client MAC address	84:b2:61:0e:0c:18
Hostname	AP84b2.610e.0c18

DHCP Offer

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

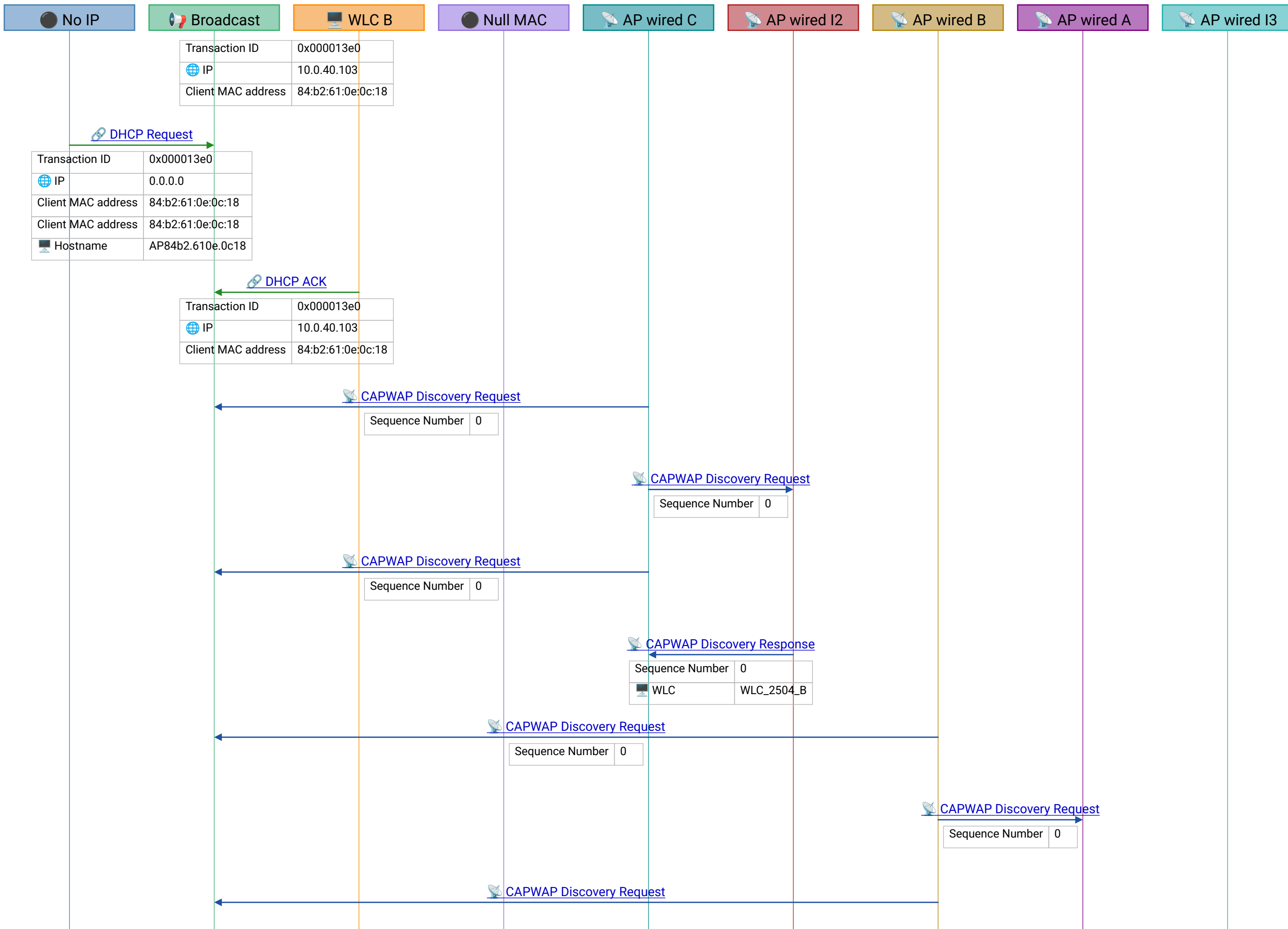
💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

Frame 5 | 2017-01-21T13:57:48.757648Z

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 DHCP – client obtains IP address after 802.11 association



and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 DHCP – client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

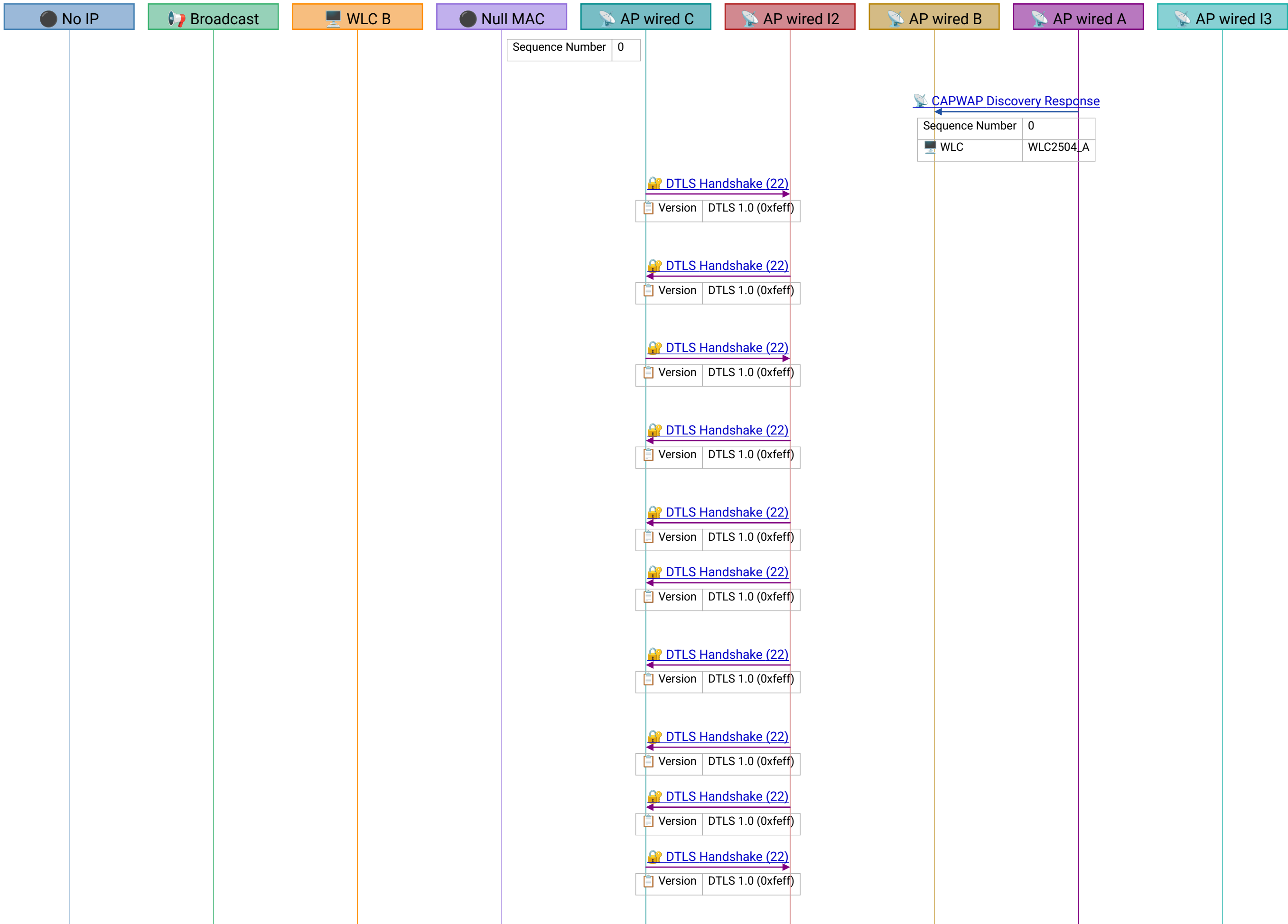
💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)



WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

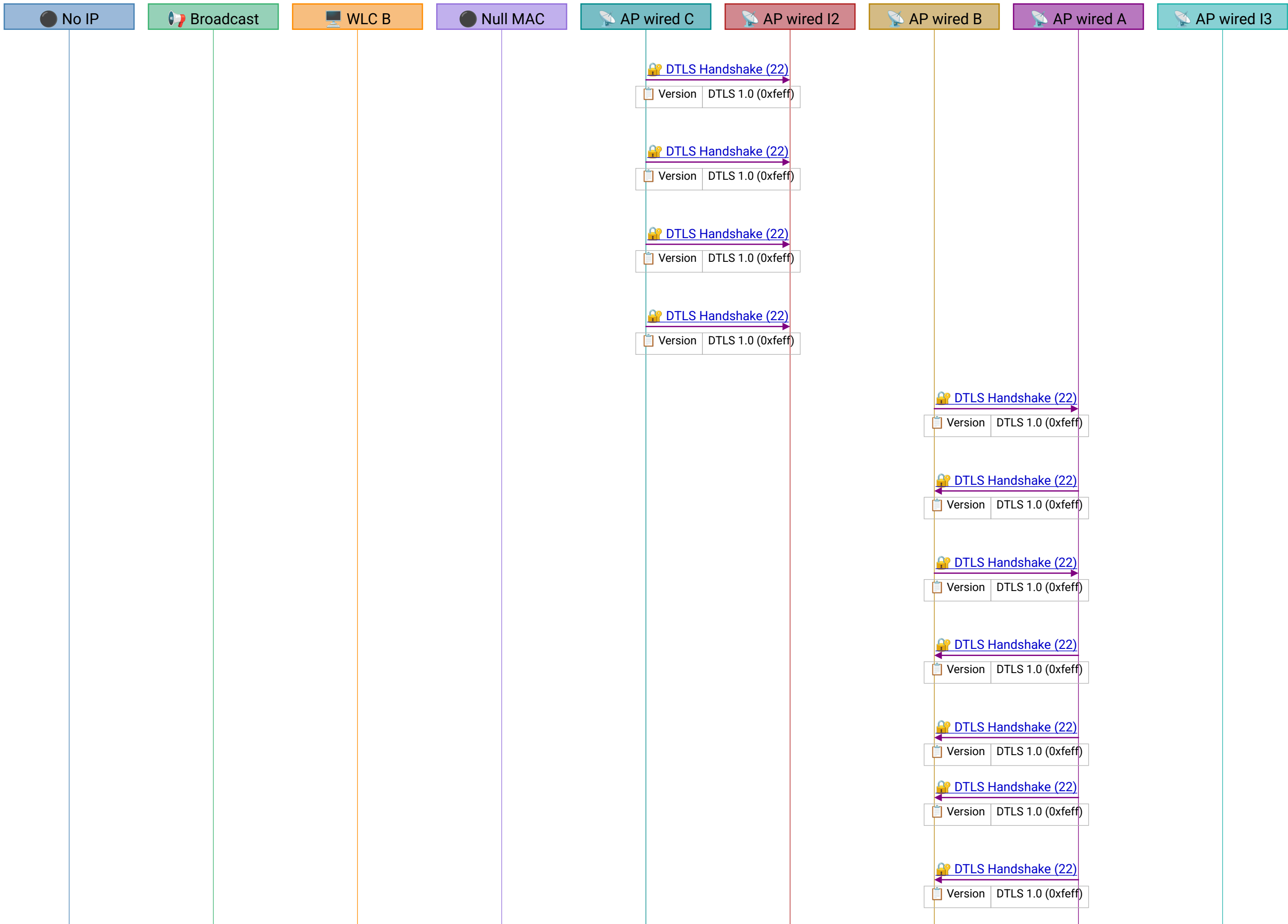
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

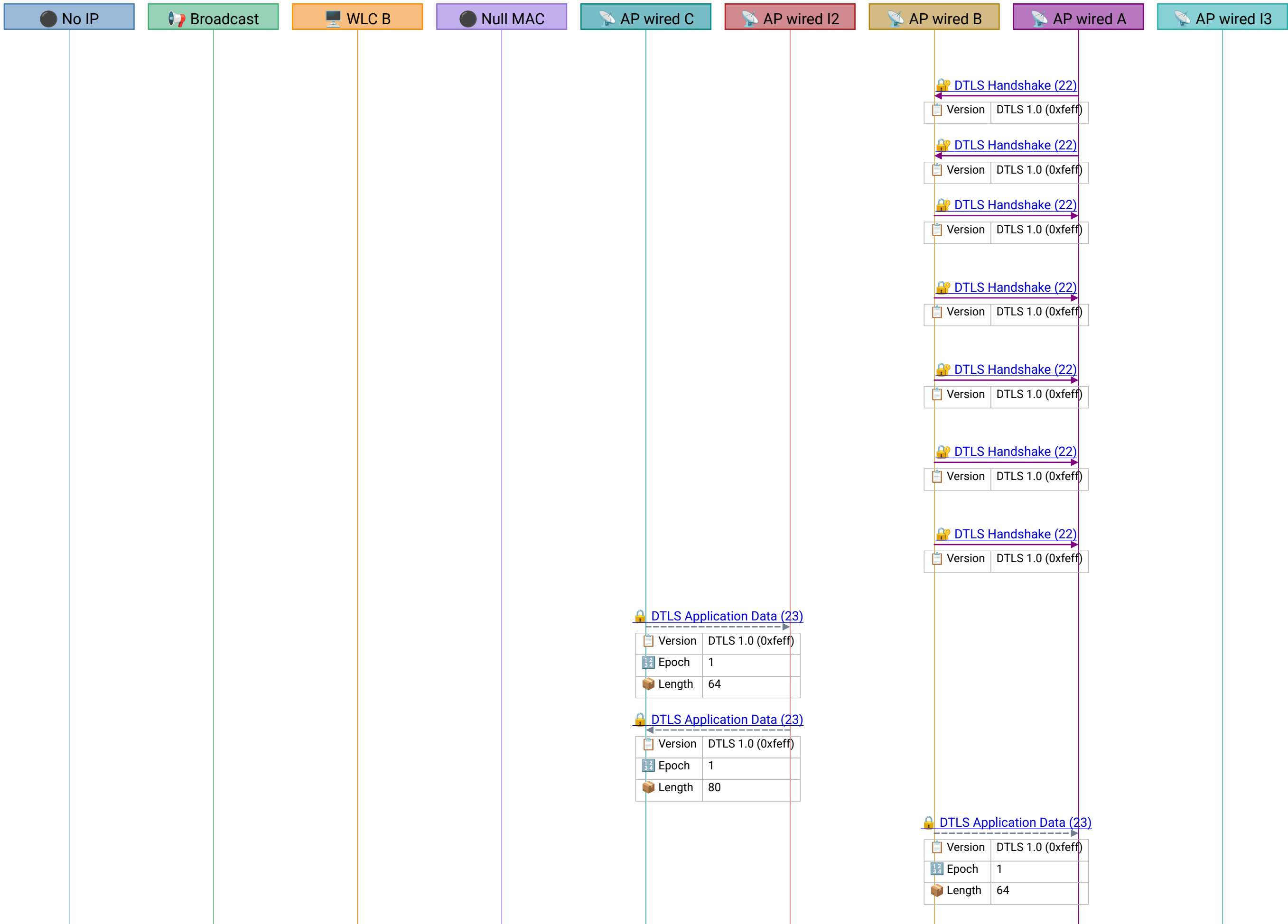
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works



with CAPWAP's UDP transport

- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport
- 💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport



uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

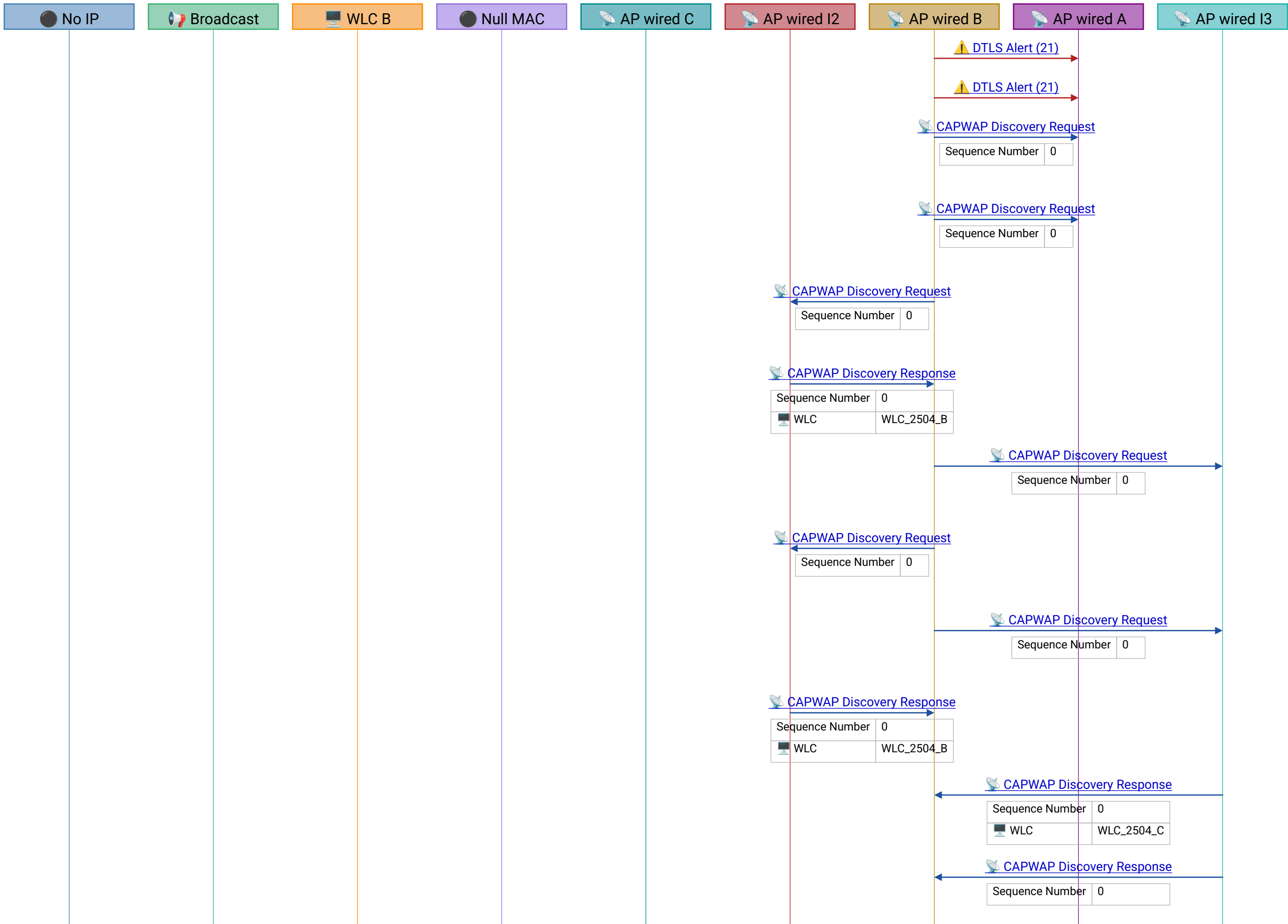
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

Frame 41 | 2017-01-21T14:01:05.26282Z

Frame 42 | 2017-01-21T14:01:05.262974Z

Frame 43 | 2017-01-21T14:01:05.340585Z



Frame 59 |
2017-01-21T14:01:11.340745Z

Frame 60 |
2017-01-21T14:01:11.340746Z

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

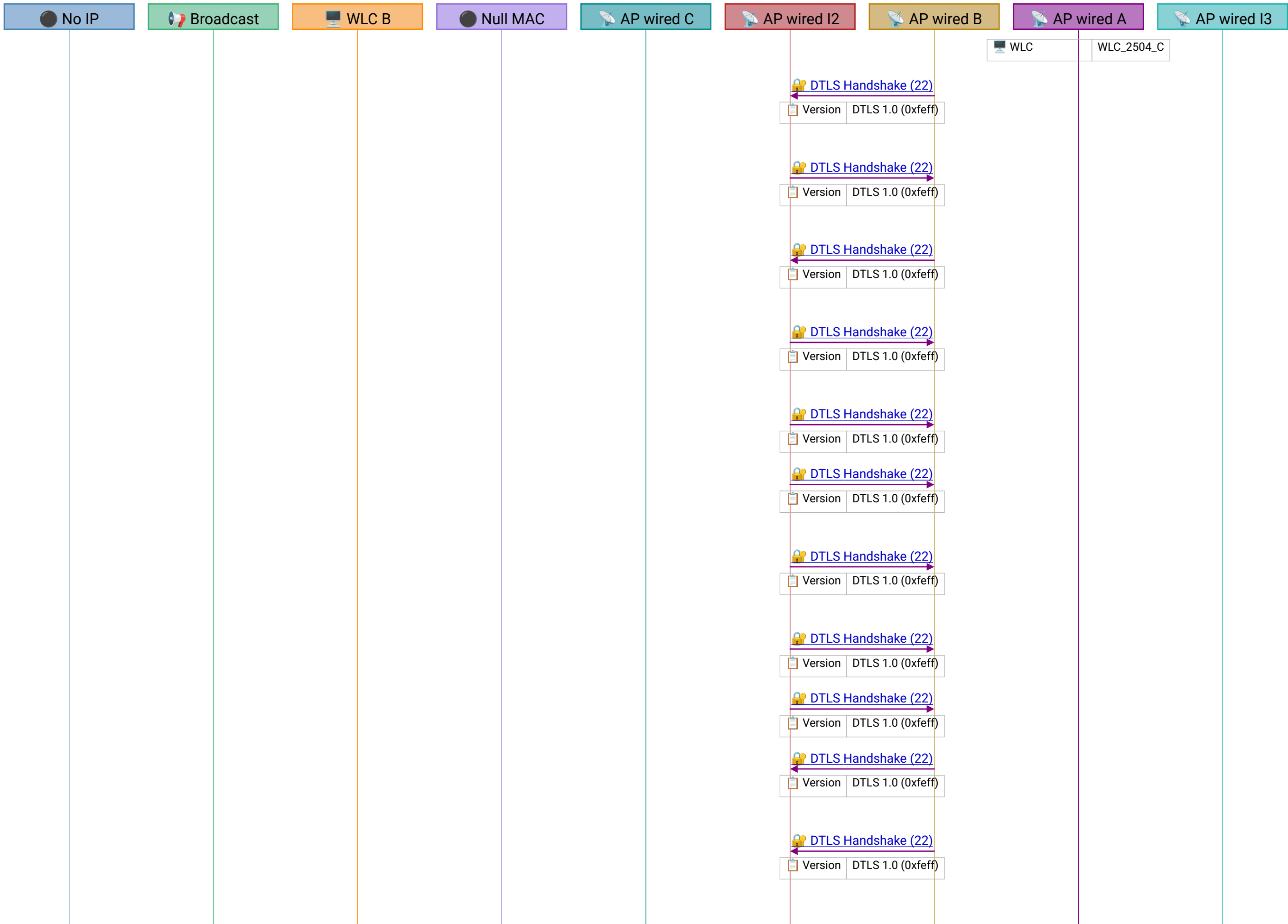
💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel,



Config provisions AP (SSID, channel, power)

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

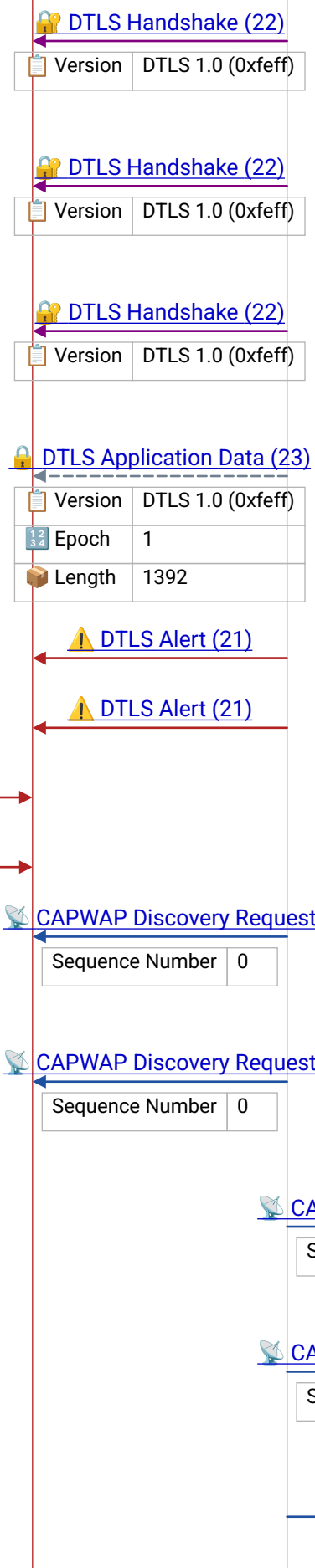
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport



💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

Frame 89 |
2017-01-21T14:02:08.208629Z

Frame 103 |
2017-01-21T14:02:14.208326Z

Frame 104 |
2017-01-21T14:02:14.208359Z

Frame 106 |
2017-01-21T14:02:14.294228Z

Frame 107 |
2017-01-21T14:02:14.294276Z

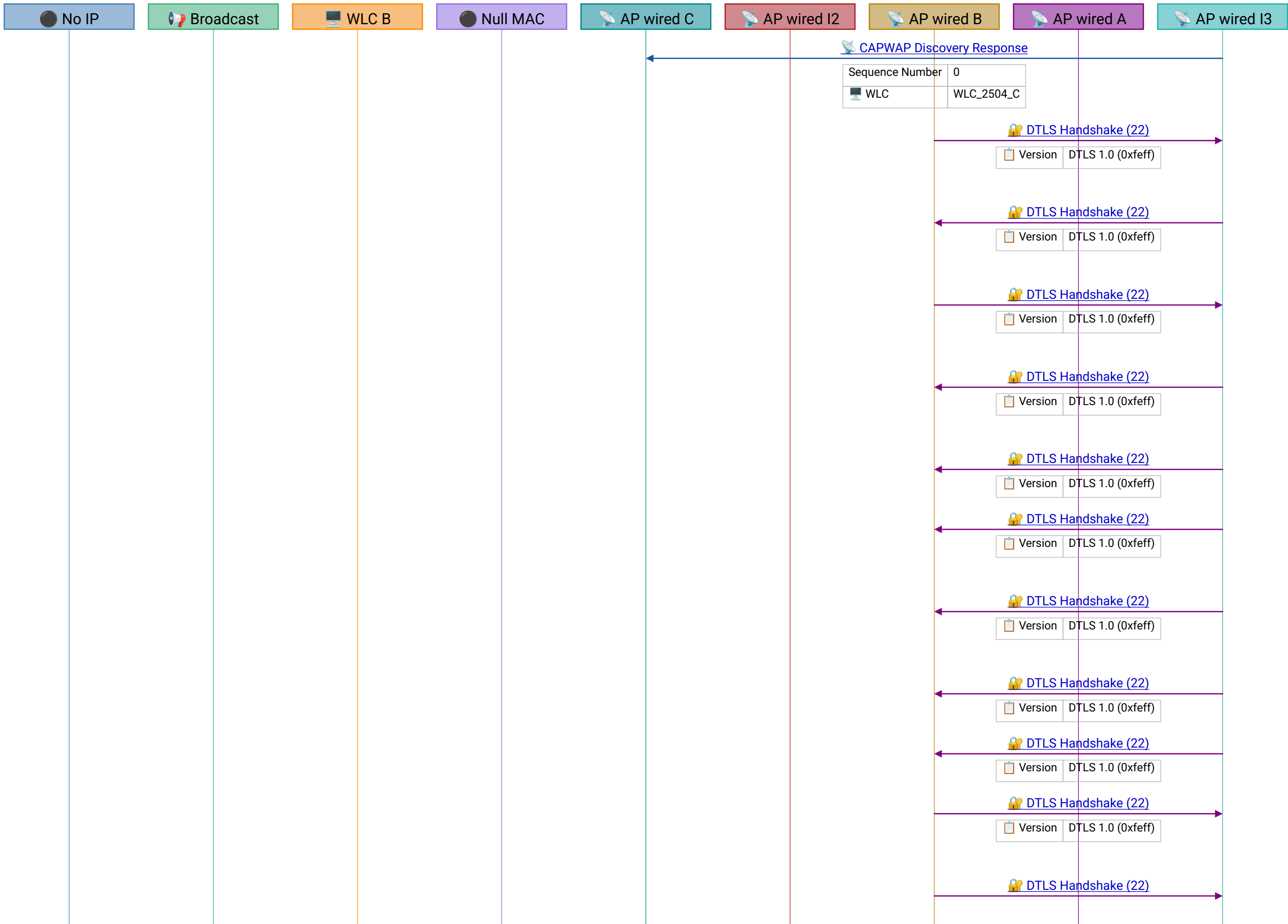
💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)



💡 CAPWAP – controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

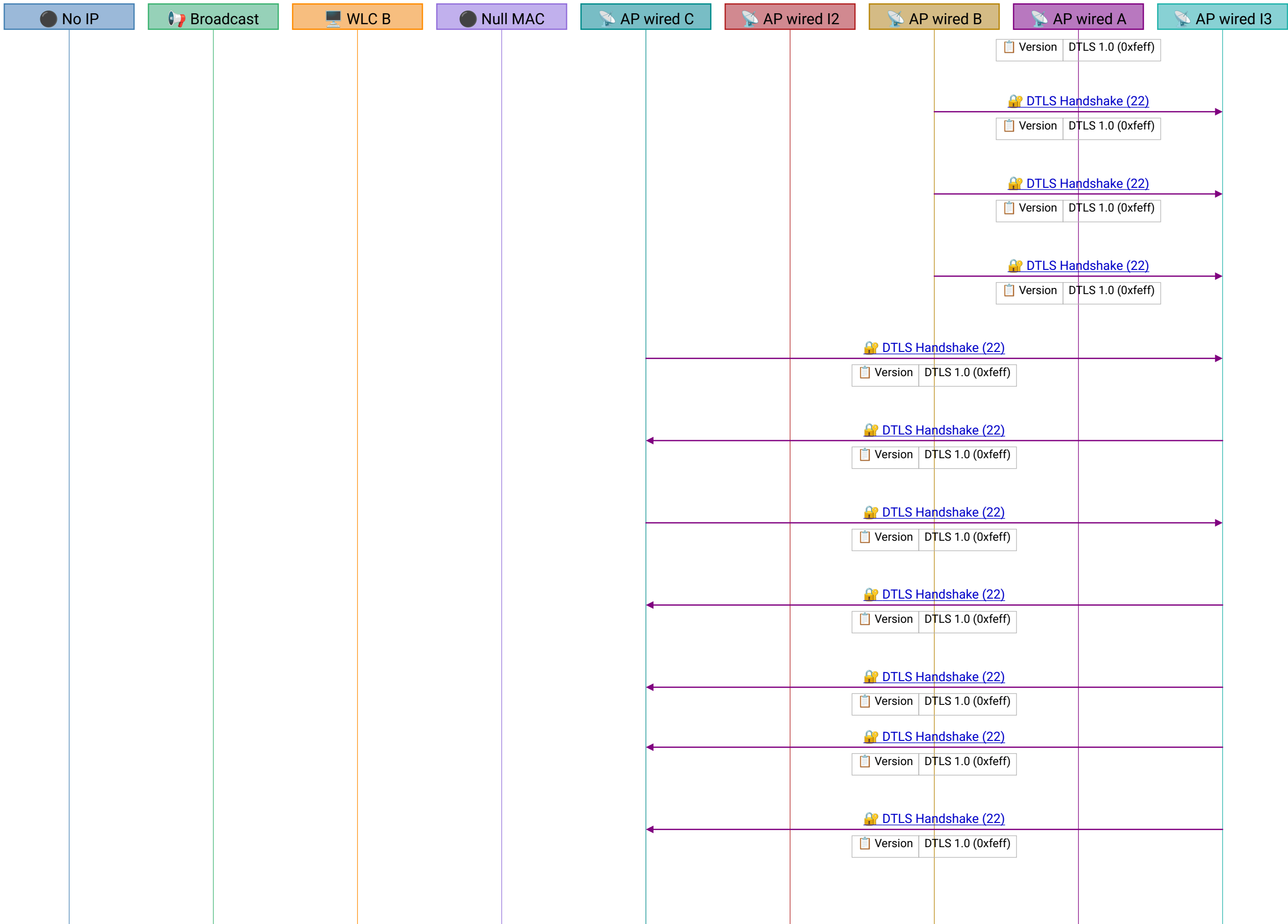
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and



WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

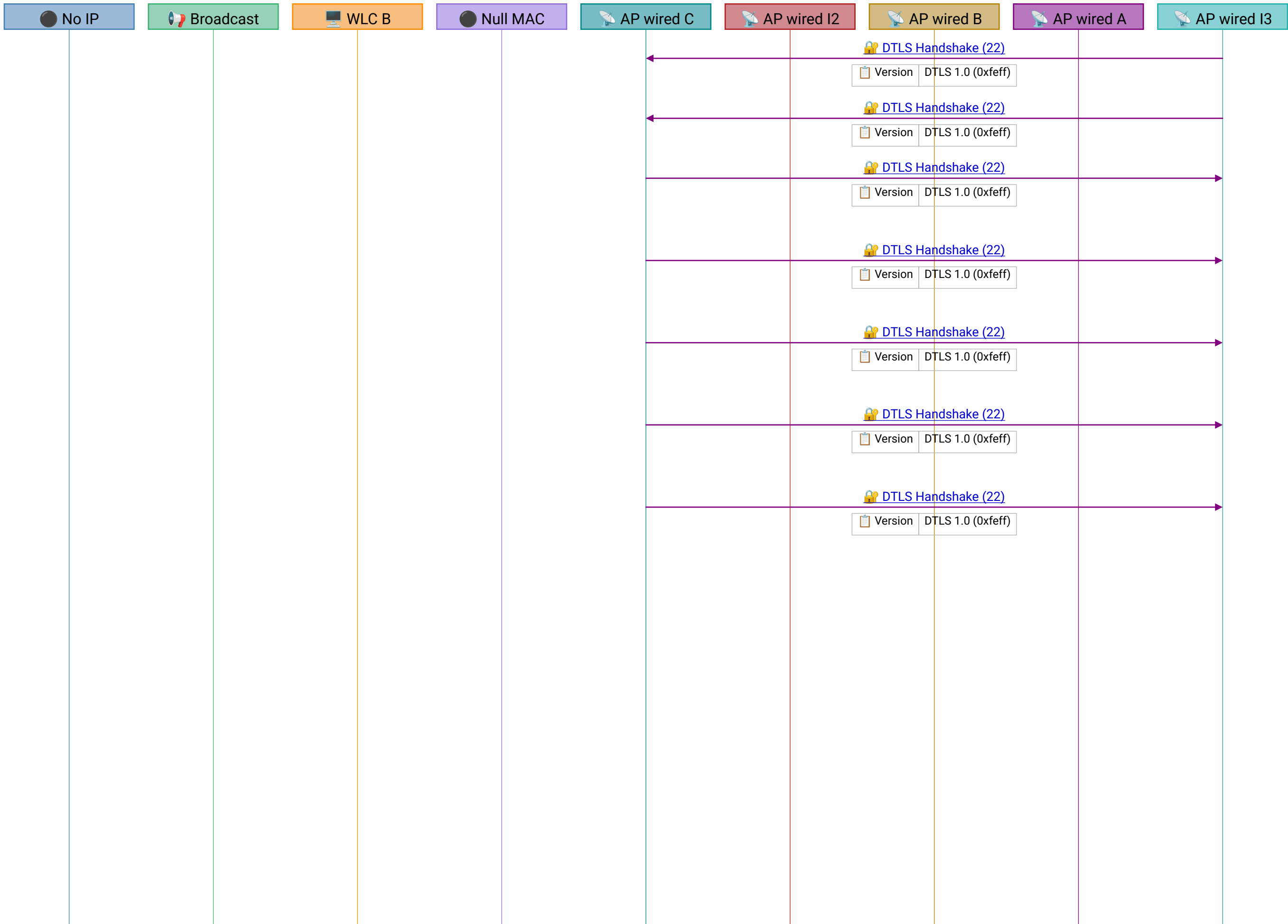
💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport



💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 DTLS Handshake – establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport