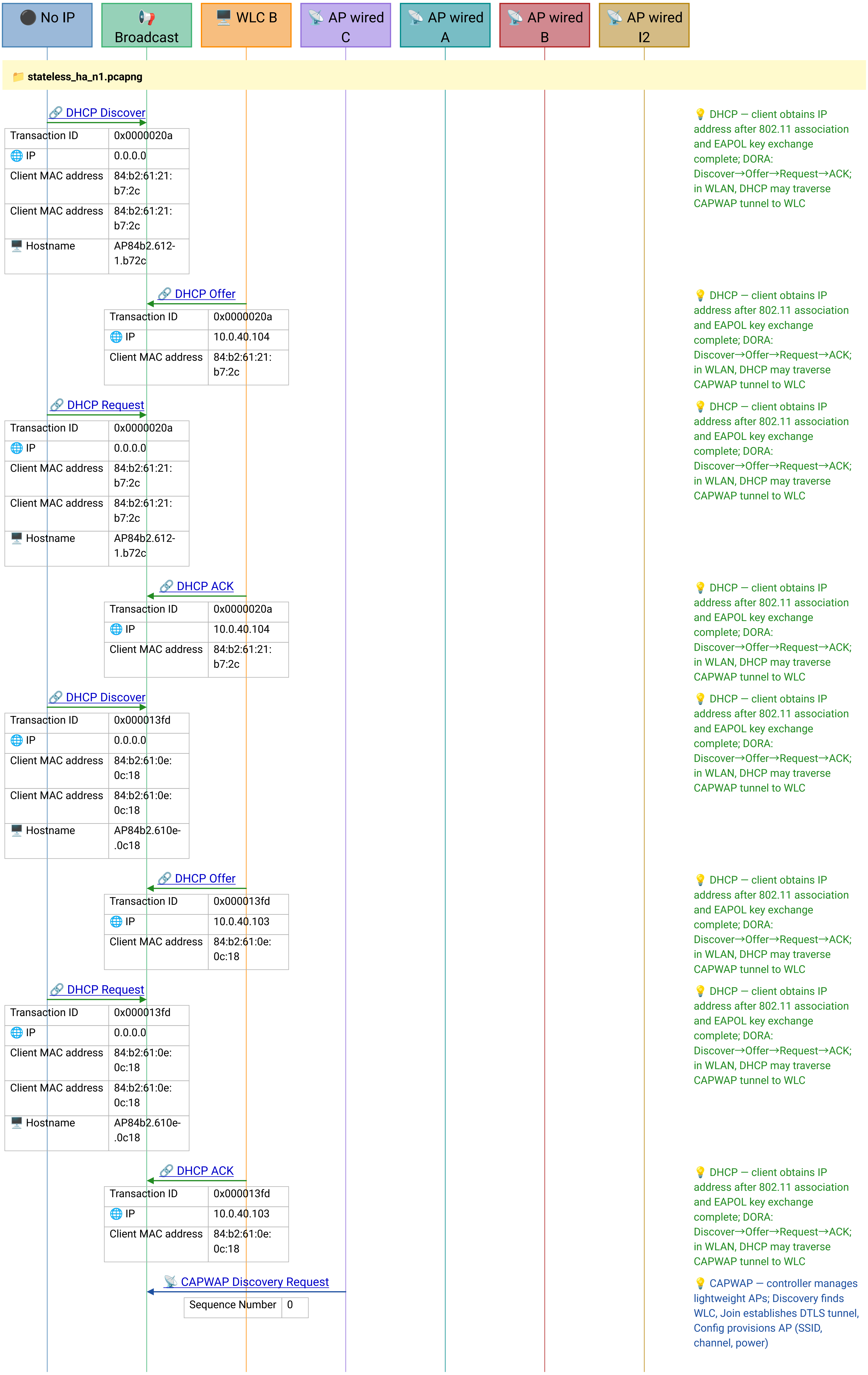
Exploring stateless_ha_n1.pcapng



⚫ No IP | 📢 Broadcast | 🖥️ WLC B | 📡 AP wired C | 📡 AP wired A | 📡 AP wired B | 📡 AP wired I2

📁 **stateless_ha_n1.pcapng**

🔗 DHCP Discover

Transaction ID	0x0000020a
🌐 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 — 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 Offer

Transaction ID	0x0000020a
🌐 IP	10.0.40.104
Client MAC address	84:b2:61:21:b7:2c

💡 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 Request

Transaction ID	0x0000020a
🌐 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 — 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 ACK

Transaction ID	0x0000020a
🌐 IP	10.0.40.104
Client MAC address	84:b2:61:21:b7:2c

💡 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 Discover

Transaction ID	0x000013fd
🌐 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 — 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 Offer

Transaction ID	0x000013fd
🌐 IP	10.0.40.103
Client MAC address	84:b2:61:0e:0c:18

💡 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 Request

Transaction ID	0x000013fd
🌐 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 — 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 ACK

Transaction ID	0x000013fd
🌐 IP	10.0.40.103
Client MAC address	84:b2:61:0e:0c:18

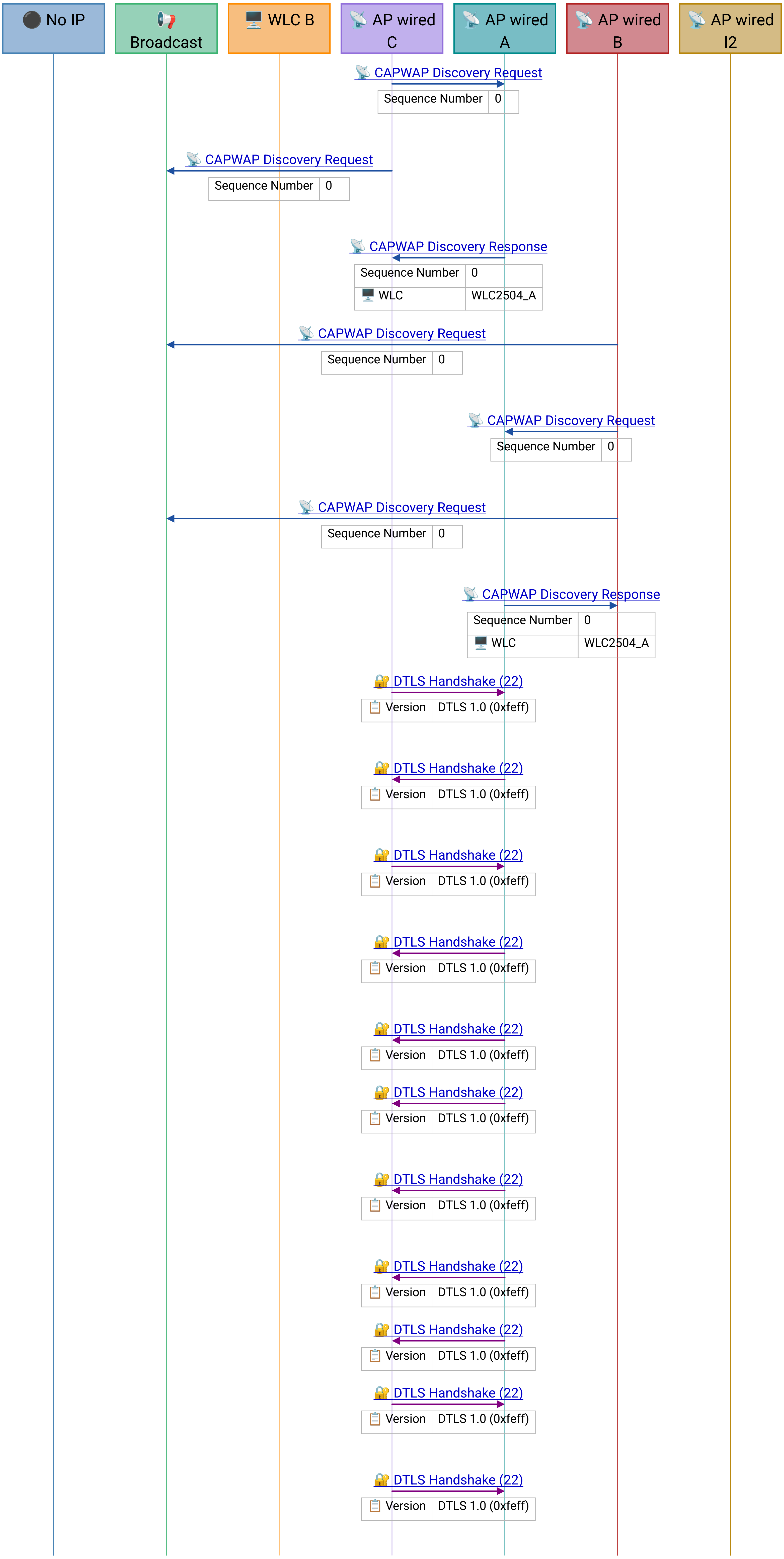
💡 DHCP — client obtains IP address after 802.11 association and EAPOL key exchange complete; DORA: Discover→Offer→Request→ACK; in WLAN, DHCP may traverse CAPWAP tunnel to WLC

📡 CAPWAP Discovery Request

Sequence Number	0

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

Exploring stateless_ha_n1.pcapng



📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Response

Sequence Number	0
🖥️ WLC	WLC2504_A

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Response

Sequence Number	0
🖥️ WLC	WLC2504_A

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

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

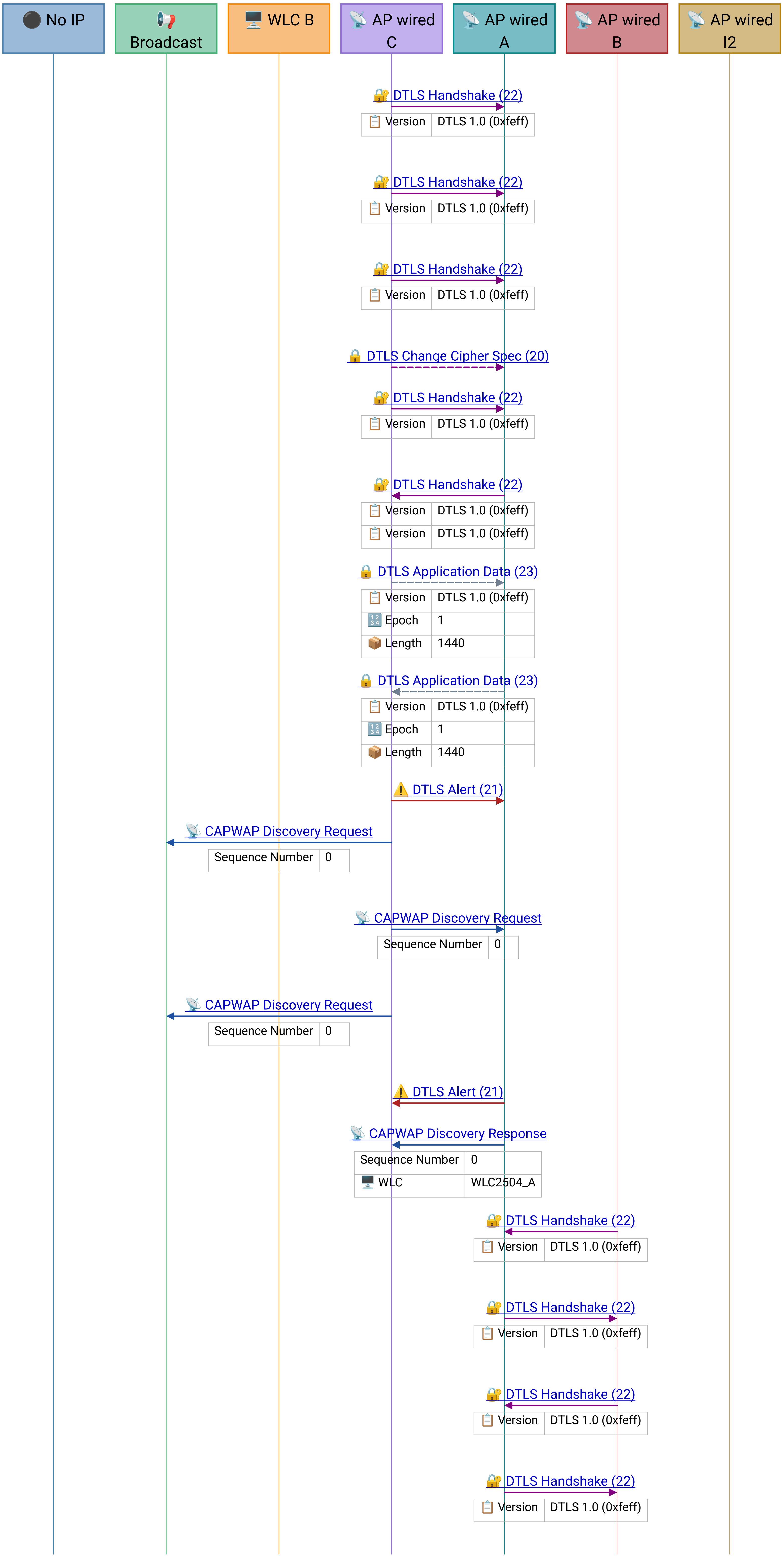
📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 Change Cipher Spec (20)

Frame 28 | 2017-01-21T13:15:39.306442Z

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)
📋 Version	DTLS 1.0 (0xfeff)

💡 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 Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	1440

Frame 31 | 2017-01-21T13:15:39.445501Z

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	1440

Frame 32 | 2017-01-21T13:15:39.446944Z

⚠️ DTLS Alert (21)

Frame 38 | 2017-01-21T13:15:41.230738Z

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

⚠️ DTLS Alert (21)

Frame 44 | 2017-01-21T13:15:41.945304Z

📡 CAPWAP Discovery Response

Sequence Number	0
🖥️ WLC	WLC2504_A

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

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

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

)

Exploring stateless_ha_n1.pcapng

with CAPWAP's UDP transport

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 Change Cipher Spec (20)

Frame 57 | 2017-01-21T13:15:43.4186Z

🔐 DTLS Handshake (22)

📋 Version	DTLS 1.0 (0xfeff)

💡 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 (22)

📋 Version	DTLS 1.0 (0xfeff)
📋 Version	DTLS 1.0 (0xfeff)

💡 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 Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	1440

Frame 60 | 2017-01-21T13:15:43.549534Z

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	1440

Frame 61 | 2017-01-21T13:15:43.551293Z

⚠️ DTLS Alert (21)

Frame 71 | 2017-01-21T13:15:45.403735Z

📡 CAPWAP Discovery Request

Sequence Number	0

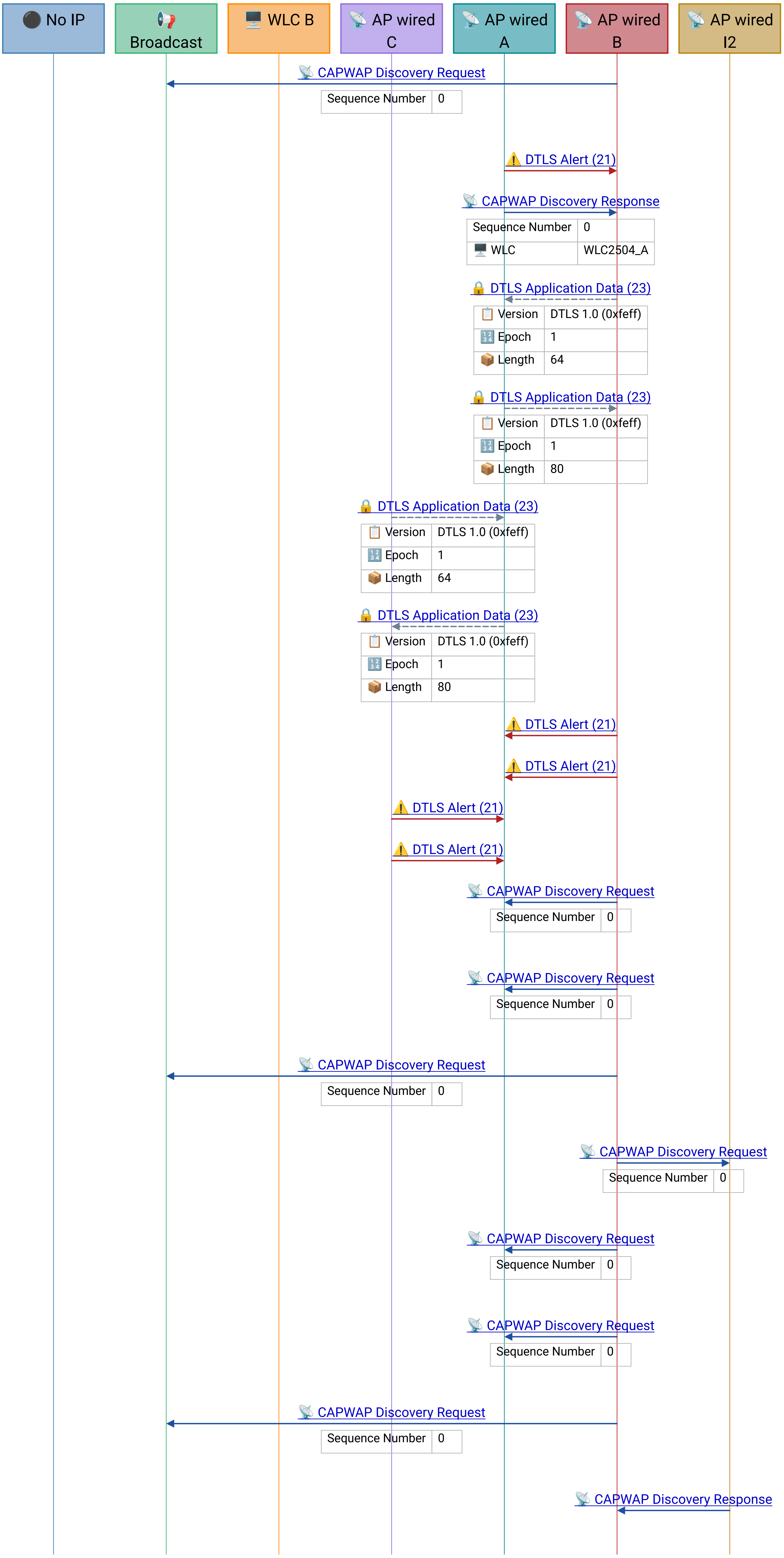
💡 CAPWAP — controller manages lightweight APs; Discovery finds WLC, Join establishes DTLS tunnel, Config provisions AP (SSID, channel, power)

📡 CAPWAP Discovery Request

Sequence Number	0

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

Exploring stateless_ha_n1.pcapng



📡 CAPWAP Discovery Request

Sequence Number	0

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

⚠️ DTLS Alert (21)

Frame 77 | 2017-01-21T13:15:46.116501Z

📡 CAPWAP Discovery Response

Sequence Number	0
🖥️ WLC	WLC2504_A

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

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	64

Frame 79 | 2017-01-21T13:16:16.22915Z

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	80

Frame 80 | 2017-01-21T13:16:16.229474Z

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	64

Frame 81 | 2017-01-21T13:16:16.945799Z

🔒 DTLS Application Data (23)

📋 Version	DTLS 1.0 (0xfeff)
🔢 Epoch	1
📦 Length	80

Frame 82 | 2017-01-21T13:16:16.94606Z

⚠️ DTLS Alert (21)

Frame 110 | 2017-01-21T13:16:35.229236Z

⚠️ DTLS Alert (21)

Frame 111 | 2017-01-21T13:16:35.229268Z

⚠️ DTLS Alert (21)

Frame 113 | 2017-01-21T13:16:35.945731Z

⚠️ DTLS Alert (21)

Frame 114 | 2017-01-21T13:16:35.945764Z

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

Sequence Number	0

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

📡 CAPWAP Discovery Request

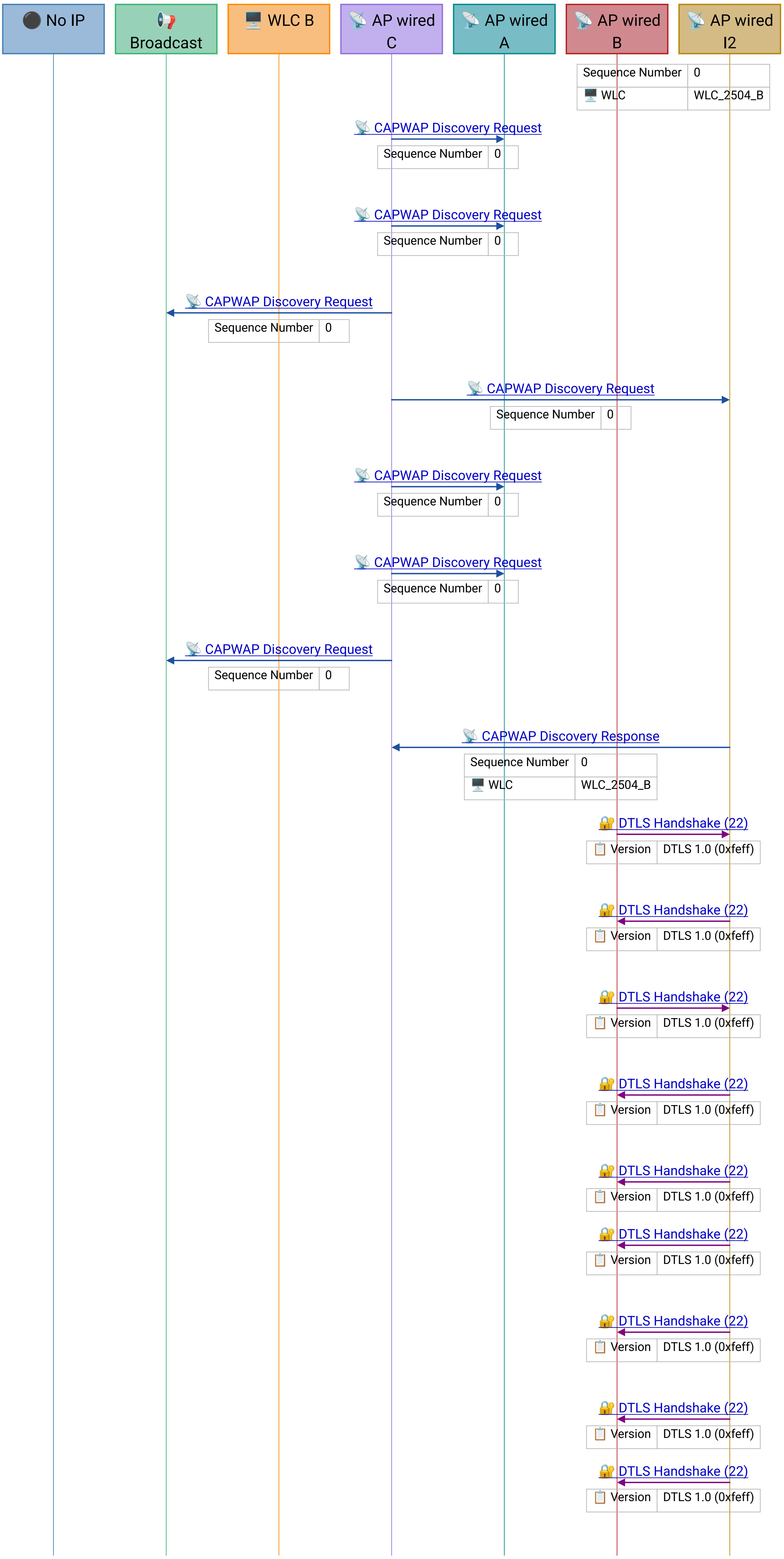
Sequence Number	0

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

📡 CAPWAP Discovery Response

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

Exploring stateless_ha_n1.pcapng



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

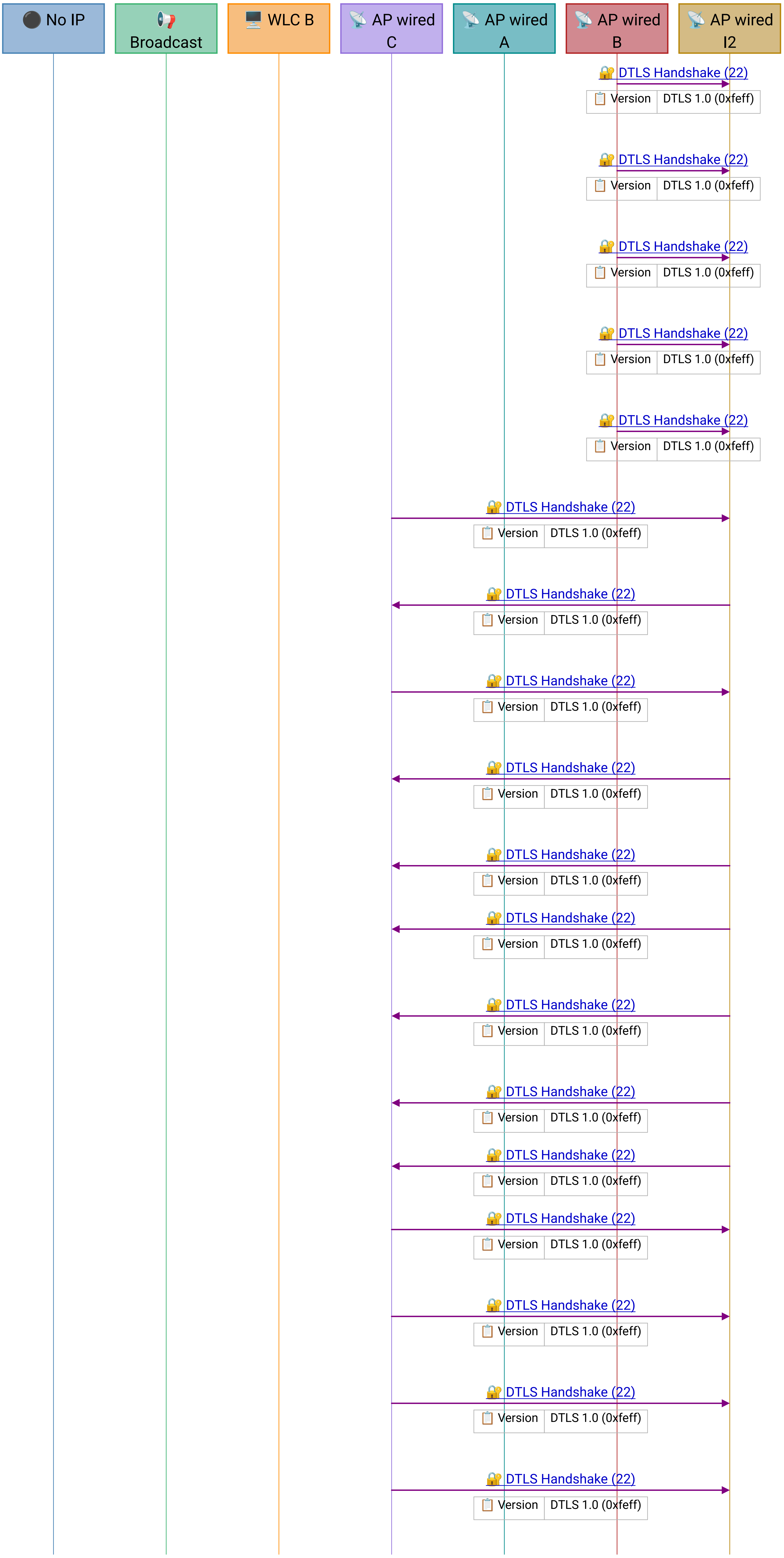
💡 DTLS Handshake — establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 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

Exploring stateless_ha_n1.pcapng



💡 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

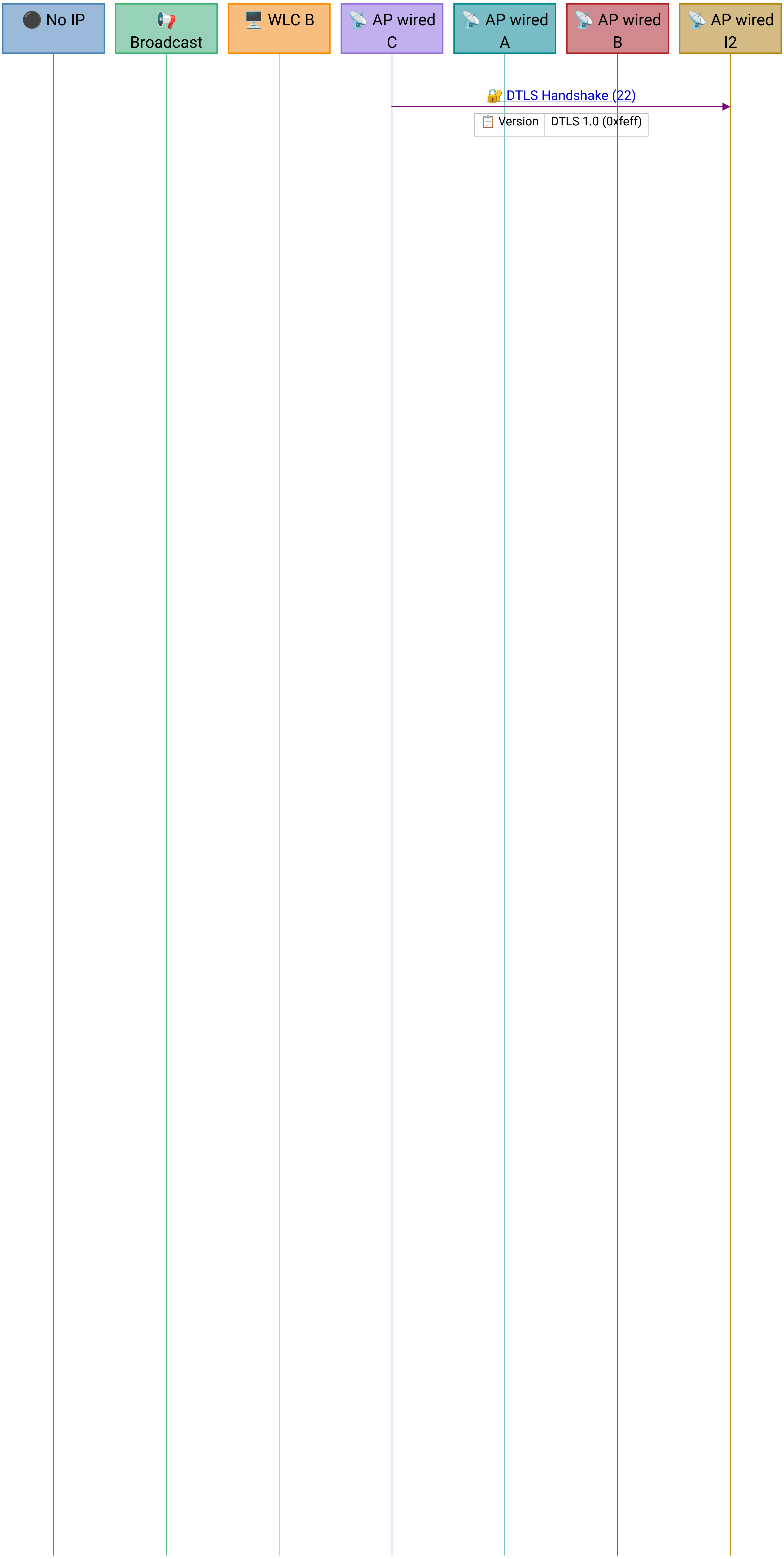
💡 DTLS Handshake — establishes encrypted tunnel between AP and WLC for CAPWAP control/data; uses UDP (not TCP) so it works with CAPWAP's UDP transport

💡 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