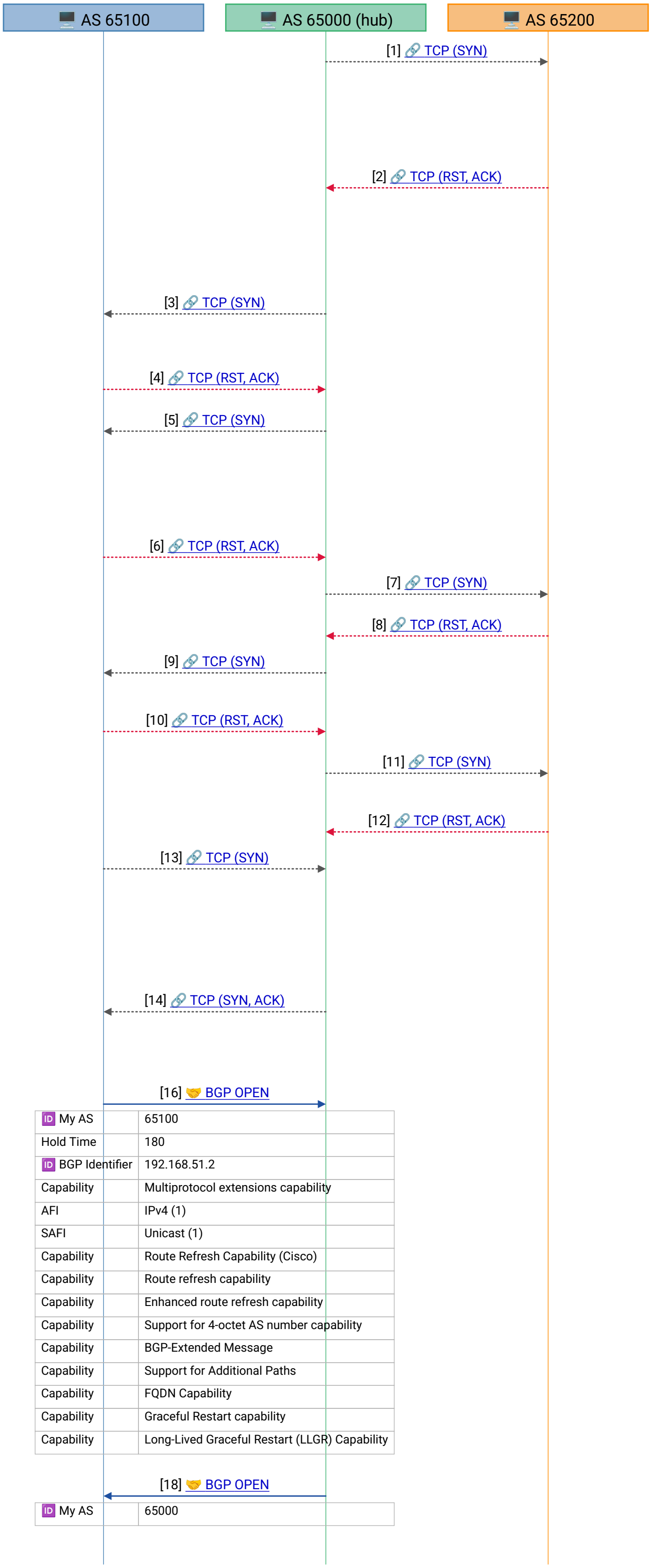
bgp-peering-and-route-exchange.annotated



[Frame 1] BGP runs over TCP, so nothing BGP-level can happen until a TCP connection exists. This is the hub **AS 65000** actively opening a connection to its AS 65200 neighbor: a TCP SYN to destination port **179** (the well-known BGP port). The source port is an incrementing ephemeral port (51416 here), which lets us track each fresh connection attempt.

[Frame 2] Connection refused. The neighbor answers the SYN with `RST, ACK` instead of `SYN, ACK`. A reset this early — before any BGP bytes — means **no process is listening on TCP/179:** the peer's BGP daemon simply isn't up yet. This is a transport-layer refusal, *not* a BGP NOTIFICATION or a policy rejection. Watch this pattern repeat until each neighbor finally comes online.

[Frame 3] At the same instant the hub also tries its **other** neighbor, **AS 65100,** over the 192.168.51.x link (SYN to 51.2:179). Same story as frame 1 — the hub is up first and is knocking on both doors.

[Frame 4] `RST, ACK` again — AS 65100's BGP isn't listening yet either. Connection refused.

[Frame 5] Retry, attempt 2 toward AS 65100. Note the timestamp: exactly ~120 s after frame 3. That cadence is BGP's **ConnectRetry timer** (default 120 s): when an active connect fails, BGP waits ConnectRetry seconds and tries again. The ephemeral source port has stepped up (55404), confirming this is a brand-new connection attempt, not a retransmission.

[Frame 6] Still refused — AS 65100 remains down.

[Frame 7] Retry, attempt 2 toward **AS 65200** (192.168.50.x link), again ~120 s after frame 1.

[Frame 8] Refused.

[Frame 9] Retry, attempt 3 toward AS 65100. This is the **last** failed attempt on the .51 link — the neighbor is about to come up (see frame 13).

[Frame 10] Refused — but only just. ~36 s later AS 65100's BGP will finally start.

[Frame 11] Retry, attempt 3 toward AS 65200 on the .50 link. AS 65200 will stay down much longer, so these retries continue for several more minutes.

[Frame 12] Refused.

[Frame 13] The .51 session comes up — and note who initiates. Now it is **AS 65100** (192.168.51.2) that sends the `SYN`, inbound to the hub's port 179. The hub *was* listening, so this direction succeeds where the hub's own outbound attempts failed. Source port 54402 -> dst 179. This is why the earlier retries never worked: the connection only completes once the *passive* side's BGP is running and it connects in.

[Frame 14] `SYN, ACK` from the hub (AS 65000) — the TCP three-way handshake is completing. (The final bare ACK isn't drawn; only the state-changing TCP segments are.) With TCP established, both ends move to the BGP **OpenSent** state.

[Frame 16] BGP OPEN from AS 65100 (RFC 4271) — the first BGP message. It carries `My AS: 65100, Hold Time: 180 s,` BGP `Identifier: 192.168.51.2,` and a capability list (4-octet AS, MP-BGP IPv4-unicast, route-refresh, graceful restart, LLGR, add-paths, FQDN). Crucially its **Graceful-Restart "Restart State" flag is set** — AS 65100 is telling the hub it is recovering from a restart, so the hub should preserve any stale routes until an End-of-RIB marker arrives.

[Frame 18] BGP OPEN from the hub, AS 65000 (`My AS: 65000,` router-ID 192.168.51.1, hold 180 s, same capability set). Its Restart-State flag is **clear** — the hub did not restart. Both sides have now OPENed with a **common 180 s hold**

AS 65100 | AS 65000 (hub) | AS 65200

Hold Time	180
BGP Identifier	192.168.51.1
Capability	Multiprotocol extensions capability
AFI	IPv4 (1)
SAFI	Unicast (1)
Capability	Route Refresh Capability (Cisco)
Capability	Route refresh capability
Capability	Enhanced route refresh capability
Capability	Support for 4-octet AS number capability
Capability	BGP-Extended Message
Capability	Support for Additional Paths
Capability	FQDN Capability
Capability	Graceful Restart capability
Capability	Long-Lived Graceful Restart (LLGR) Capability

time; the lower of the two proposals wins, and here both proposed 180. The session moves toward Established.

[20] BGP KEEPALIVE

[21] BGP KEEPALIVE

[Frame 20] KEEPALIVE from AS 65100 — a 19-byte header-only message that acknowledges the peer's OPEN. Receiving a KEEPALIVE in OpenConfirm is what promotes each side to **Established.** Steady-state KEEPALIVEs then run at ~1/3 of the hold time, i.e. ~60 s.

[Frame 21] KEEPALIVE from the hub — the reciprocal confirmation. The **AS65000 <-> AS65100 session is now Established** and route exchange can begin.

[24.1] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65100
NEXT_HOP	192.168.51.2
Path Attribute - MULTI_EXIT_DISC	0
Path Attribute - COMMUNITIES	321:654
Prefix Length	16
NLRI	10.30.0.0

[Frame 24.1] First UPDATE: AS 65100 advertises its own routes. `10.30.0.0/16` with `ORIGIN=IGP`, `AS_PATH=65100` (a one-hop path means AS 65100 is the origin), `NEXT_HOP=192.168.51.2`, `MED=0`, community `321:654`.

[24.2] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65100
NEXT_HOP	192.168.51.2
Path Attribute - MULTI_EXIT_DISC	0
Path Attribute - COMMUNITIES	123:456
Prefix Length	16
NLRI	10.40.0.0

[Frame 24.2] Second prefix in the same UPDATE burst: `10.40.0.0/16`, also `AS_PATH=65100`, community `123:456`. These two prefixes (10.30, 10.40) are AS 65100's origin routes.

[24.3] BGP UPDATE

[Frame 24.3] End-of-RIB marker. An UPDATE with *no* NLRI and *no* withdrawn routes. Because graceful restart was negotiated (and AS 65100 flagged Restart-State), this empty UPDATE signals "I have finished sending my initial routing table" (RFC 4724) — the hub may now flush any stale 65100 routes it was holding.

[26.1] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65000
NEXT_HOP	192.168.51.1
Path Attribute - MULTI_EXIT_DISC	0
Path Attribute - COMMUNITIES	123:456
Prefix Length	16
NLRI	10.10.0.0

[Frame 26.1] The hub answers with its full table toward AS 65100. `10.10.0.0/16`, `AS_PATH=65000` (hub-originated), community `123:456`.

[26.2] BGP UPDATE

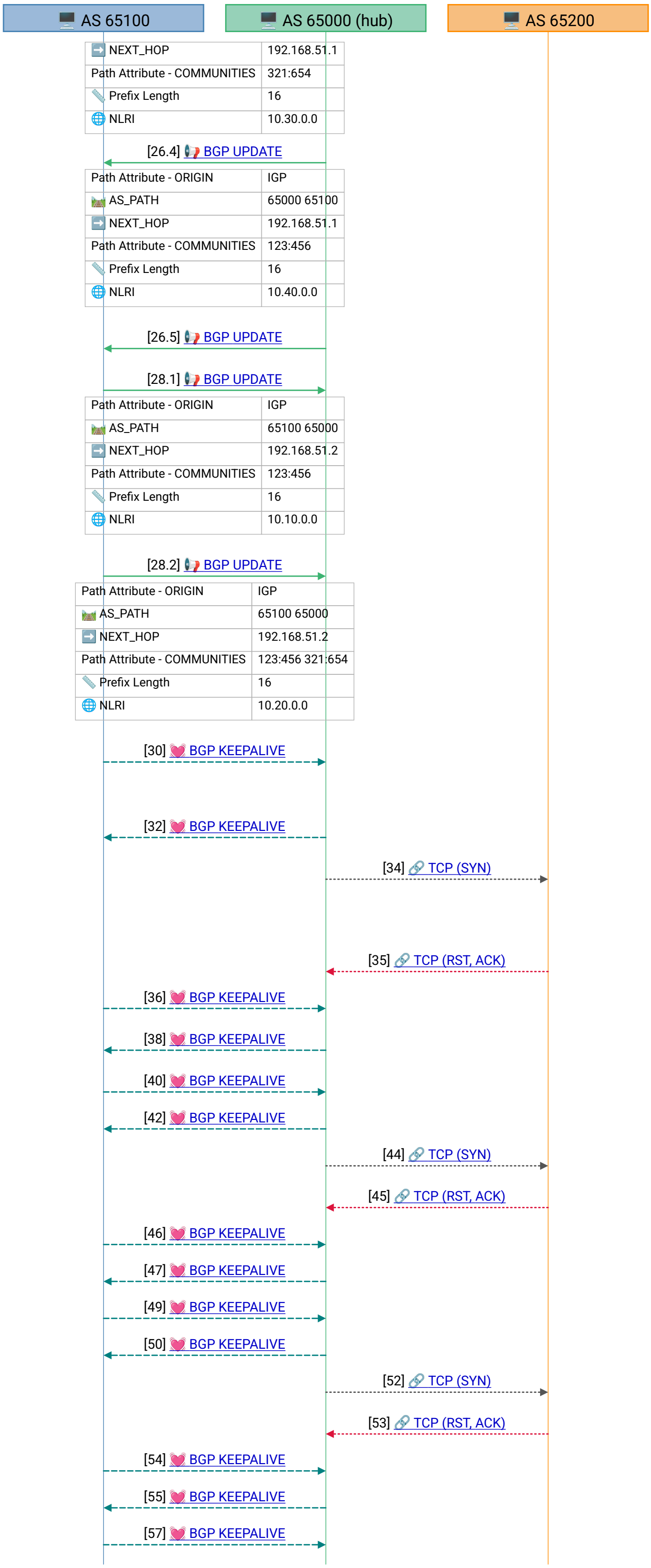
Path Attribute - ORIGIN	IGP
AS_PATH	65000
NEXT_HOP	192.168.51.1
Path Attribute - MULTI_EXIT_DISC	0
Path Attribute - COMMUNITIES	123:456 321:654
Prefix Length	16
NLRI	10.20.0.0

[Frame 26.2] `10.20.0.0/16`, `AS_PATH=65000` (hub-originated), communities `123:456` `321:654`.

[26.3] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65000 65100

[Frame 26.3] `10.30.0.0/16`, `AS_PATH=65000 65100`. This is 65100's *own* route being reflected back with 65000 prepended — 65100 will ignore it because its own AS is already in the path (loop prevention).



[Frame 26.4] `10.40.0.0/16, AS_PATH=65000 65100` — same reflect-back as sub 3.

[Frame 26.5] End-of-RIB marker from the hub — its initial advertisement to AS 65100 is complete.

[Frame 28.1] AS 65100 re-advertises the hub's origin routes back, now with itself prepended. `10.10.0.0/16, AS_PATH=65100 65000`. (The hub, in turn, drops these — 65000 is already in the path.) This is the normal BGP "advertise everything you use to your neighbor" behavior; the AS_PATH loop rule keeps it from cycling.

[Frame 28.2] `10.20.0.0/16, AS_PATH=65100 65000` — same as sub 1 for the second hub prefix. The AS65000<->AS65100 tables are now fully synchronized.

[Frame 30] First **periodic KEEPALIVE** on the AS65000<->AS65100 session (from 65100), ~60 s after the session came up. Each KEEPALIVE resets the peer's 180 s hold timer; miss ~3 in a row and the peer is declared dead.

[Frame 32] Reciprocal periodic KEEPALIVE from the hub. The .51 session is healthy and idle (no route changes).

[Frame 34] Meanwhile, the hub is still hammering AS 65200. Active-open retry (attempt 4) on the .50 link — AS 65200's BGP is *still* not up, so this will be refused. Note the .51 link no longer generates these; only the .50 neighbor is still missing.

[Frame 35] `RST, ACK` — AS 65200 still down.

[Frame 36] Periodic KEEPALIVE, AS 65100 -> hub (.51 session).

[Frame 38] Periodic KEEPALIVE, hub -> AS 65100 (.51 session).

[Frame 40] Periodic KEEPALIVE, AS 65100 -> hub.

[Frame 42] Periodic KEEPALIVE, hub -> AS 65100.

[Frame 44] Hub -> AS 65200 active-open **retry attempt 5** (.50 link). Still ~120 s ConnectRetry spacing.

[Frame 45] Refused.

[Frame 46] Periodic KEEPALIVE, AS 65100 -> hub.

[Frame 47] Periodic KEEPALIVE, hub -> AS 65100.

[Frame 49] Periodic KEEPALIVE, AS 65100 -> hub.

[Frame 50] Periodic KEEPALIVE, hub -> AS 65100.

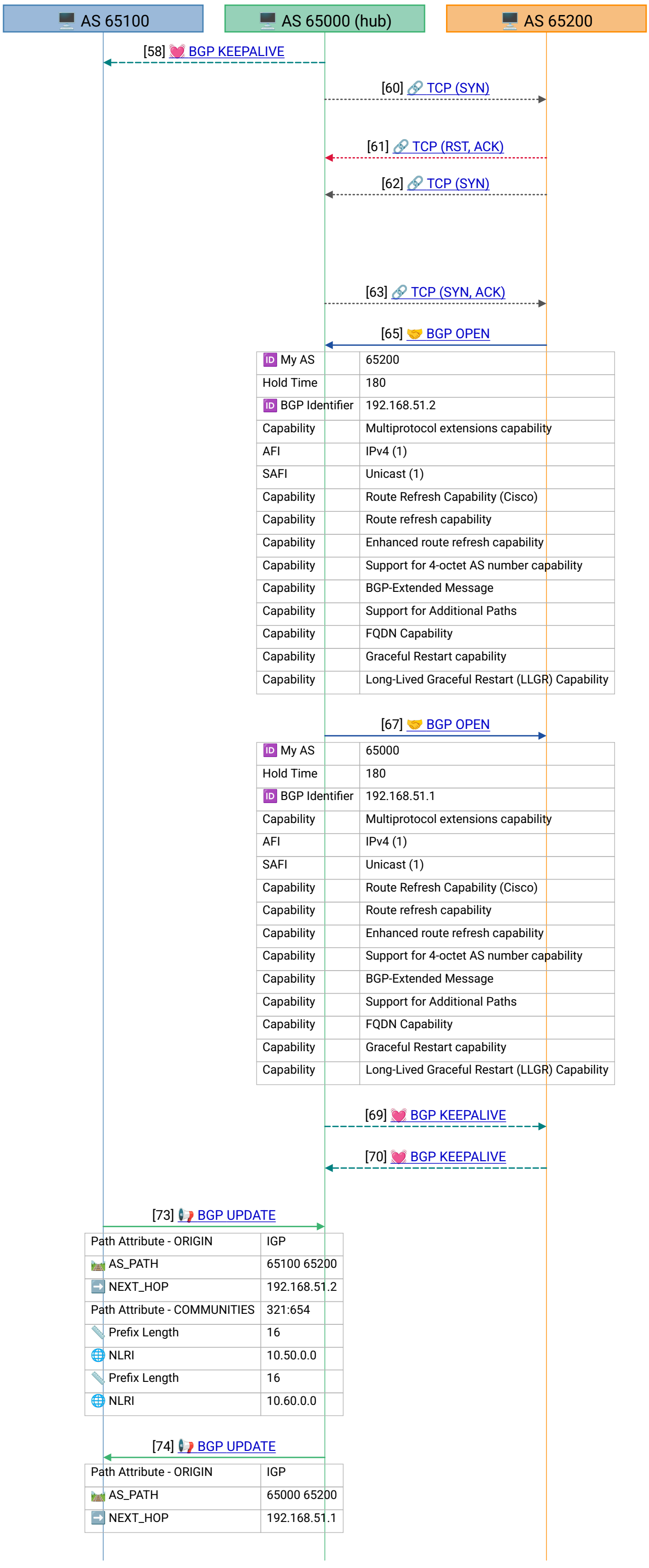
[Frame 52] Hub -> AS 65200 active-open **retry attempt 6** (.50 link).

[Frame 53] Refused.

[Frame 54] Periodic KEEPALIVE, AS 65100 -> hub.

[Frame 55] Periodic KEEPALIVE, hub -> AS 65100.

[Frame 57] Periodic KEEPALIVE, AS 65100 -> hub.



[Frame 58] Periodic KEEPALIVE, hub -> AS 65100.

[Frame 60] Hub -> AS 65200 active-open **retry attempt 7** (.50 link) — the last failed attempt. AS 65200 is about to come up.

[Frame 61] Refused — for the last time on this link.

[Frame 62] The .50 session finally establishes — again, the spoke initiates. AS 65200 (192.168.50.2) sends the inbound SYN to the hub's port 179 (src port 60314). Just like AS 65100 before it, the connection only completes once AS 65200's BGP is running and connects *in* to the always-listening hub.

[Frame 63] `SYN, ACK` from the hub — TCP handshake completing on the .50 link.

[Frame 65] BGP OPEN from AS 65200 — `My AS: 65200`, router-ID 192.168.51.2, hold 180 s, and again the **Graceful-Restart Restart-State flag is set.** So AS 65200, like AS 65100, is coming back from a restart and will end its initial advertisement with an End-of-RIB marker.

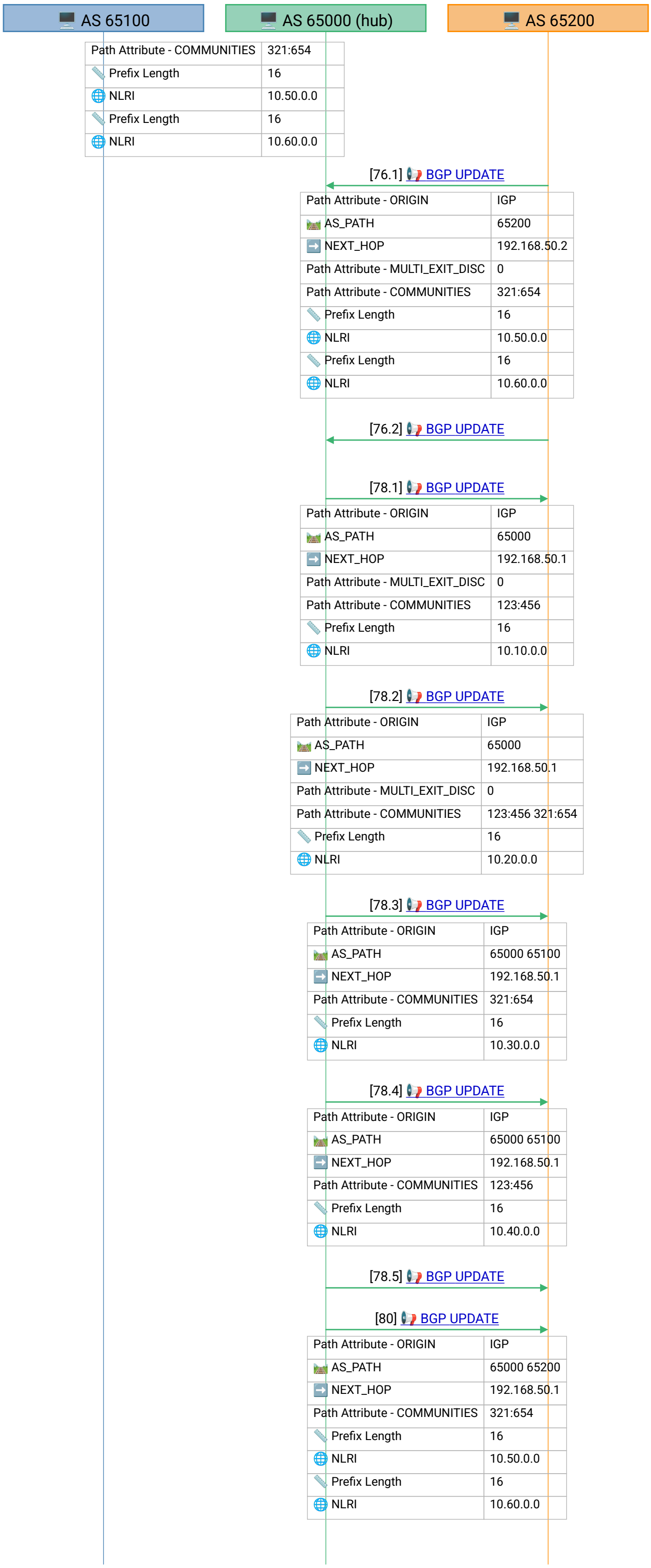
[Frame 67] BGP OPEN from the hub, AS 65000 toward AS 65200 (`My AS: 65000`, router-ID 192.168.51.1, Restart-State clear). Both sides OPENed with hold 180 s; the AS65000<->AS65200 session moves to Established.

[Frame 69] KEEPALIVE from the hub confirming AS 65200's OPEN.

[Frame 70] KEEPALIVE from AS 65200 — the AS65000<->AS65200 session is now **Established.** Both of the hub's peerings are up.

[Frame 73] A new AS (65200) just joined, so its prefixes ripple across the *existing* .51 session too. Here AS 65100 advertises `10.50.0.0/16` and `10.60.0.0/16` (AS 65200's origin routes) toward the hub with `AS_PATH=65100 65200`. All of frames 73–82 happen within ~50 ms at 17:21:13, triggered by the .50 session coming up; frame-number order on the diagram does not strictly match their sub-millisecond timing.

[Frame 74] The hub propagates AS 65200's new routes to AS 65100. `10.50.0.0/16` and `10.60.0.0/16` with `AS_PATH=65000 65200` — the hub prepends its own AS to the path it learned from AS 65200. This is BGP doing its core job: making AS 65200's reachability visible to AS 65100 without those two ever peering directly.



[Frame 76.1] On the new .50 session, AS 65200 advertises its origin routes. `10.50.0.0/16` and `10.60.0.0/16`, `AS_PATH=65200` (one hop = origin), `NEXT_HOP=192.168.50.2`, community `321:654`.

[Frame 76.2] End-of-RIB marker from AS 65200 — its initial table (just 10.50/10.60) is complete. This is the marker its Restart-State flag in frame 65 promised.

[Frame 78.1] The hub sends AS 65200 everything it knows. `10.10.0.0/16`, `AS_PATH=65000` (hub origin).

[Frame 78.2] `10.20.0.0/16`, `AS_PATH=65000` (hub origin).

[Frame 78.3] `10.30.0.0/16`, `AS_PATH=65000 65100` — a route the hub learned from AS 65100, now handed to AS 65200. AS 65200 gains reachability to AS 65100's prefixes via the hub.

[Frame 78.4] `10.40.0.0/16`, `AS_PATH=65000 65100` — the second AS 65100 prefix, transited to AS 65200.

[Frame 78.5] End-of-RIB marker from the hub toward AS 65200.

[Frame 80] The hub also echoes `10.50.0.0/16` & `10.60.0.0/16` back toward AS 65200 with `AS_PATH=65000 65200`. AS 65200 **discards** these on receipt — its own AS (65200) is already in the AS_PATH, so the loop-prevention rule rejects them. Harmless, and a clean illustration of why AS_PATH exists.

AS 65100 | AS 65000 (hub) | AS 65200

[82.1] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65200 65000
NEXT_HOP	192.168.50.2
Path Attribute - COMMUNITIES	123:456
Prefix Length	16
NLRI	10.10.0.0

[Frame 82.1] AS 65200 re-advertises the routes it now uses back to the hub, prepending itself. `10.10.0.0/16`, `AS_PATH=65200 65000` (hub drops it — 65000 already in path).

[82.2] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65200 65000
NEXT_HOP	192.168.50.2
Path Attribute - COMMUNITIES	123:456 321:654
Prefix Length	16
NLRI	10.20.0.0

[Frame 82.2] `10.20.0.0/16, AS_PATH=65200 65000.`

[82.3] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65200 65100
NEXT_HOP	192.168.50.2
Path Attribute - COMMUNITIES	321:654
Prefix Length	16
NLRI	10.30.0.0

[Frame 82.3] `10.30.0.0/16, AS_PATH=65200 65100.` Note the path does **not** contain 65000 even though the route came via the hub — an artifact of how this lab capture was built; in a live network you would expect `65200 65000 65100`.

[82.4] BGP UPDATE

Path Attribute - ORIGIN	IGP
AS_PATH	65200 65100
NEXT_HOP	192.168.50.2
Path Attribute - COMMUNITIES	123:456
Prefix Length	16
NLRI	10.40.0.0

[Frame 82.4] `10.40.0.0/16, AS_PATH=65200 65100.` With this, both peerings have fully converged: every AS can now reach all six /16 prefixes. **Route exchange is done; the rest of the capture is pure keepalive heartbeat.**

[84] BGP KEEPALIVE

[Frame 84] Steady state begins. From here to the end of the capture there are no route changes — only ~60 s KEEPALIVE heartbeats on each session, every one resetting the sender's 180 s hold timer. This is AS 65100 -> hub on the .51 session.

[85] BGP KEEPALIVE

[Frame 85] Steady-state KEEPALIVE, hub -> AS 65100 (.51 session).

[87] BGP KEEPALIVE

[Frame 87] Steady-state KEEPALIVE, AS 65200 -> hub (.50 session) — the first heartbeat on the newer peering.

[89] BGP KEEPALIVE

[Frame 89] Steady-state KEEPALIVE, hub -> AS 65200 (.50 session).

[91] BGP KEEPALIVE

[Frame 91] KEEPALIVE, AS 65100 -> hub (.51).

[92] BGP KEEPALIVE

[Frame 92] KEEPALIVE, hub -> AS 65100 (.51).

[94] BGP KEEPALIVE

[Frame 94] KEEPALIVE, AS 65200 -> hub (.50).

[96] BGP KEEPALIVE

[Frame 96] KEEPALIVE, hub -> AS 65200 (.50).

[98] BGP KEEPALIVE

[Frame 98] KEEPALIVE, AS 65100 -> hub (.51).

[99] BGP KEEPALIVE

[Frame 99] KEEPALIVE, hub -> AS 65100 (.51).

[101] BGP KEEPALIVE

[Frame 101] KEEPALIVE, AS 65200 -> hub (.50).

[103] BGP KEEPALIVE

[Frame 103] KEEPALIVE, hub -> AS 65200 (.50).

[105] BGP KEEPALIVE

[Frame 105] KEEPALIVE, AS 65100 -> hub (.51).

[106] BGP KEEPALIVE

[Frame 106] KEEPALIVE, hub -> AS 65100 (.51).

[108] BGP KEEPALIVE

[Frame 108] KEEPALIVE, AS 65200 -> hub (.50).

[109] BGP KEEPALIVE

[Frame 109] KEEPALIVE, hub -> AS 65200 (.50).

[111] BGP KEEPALIVE

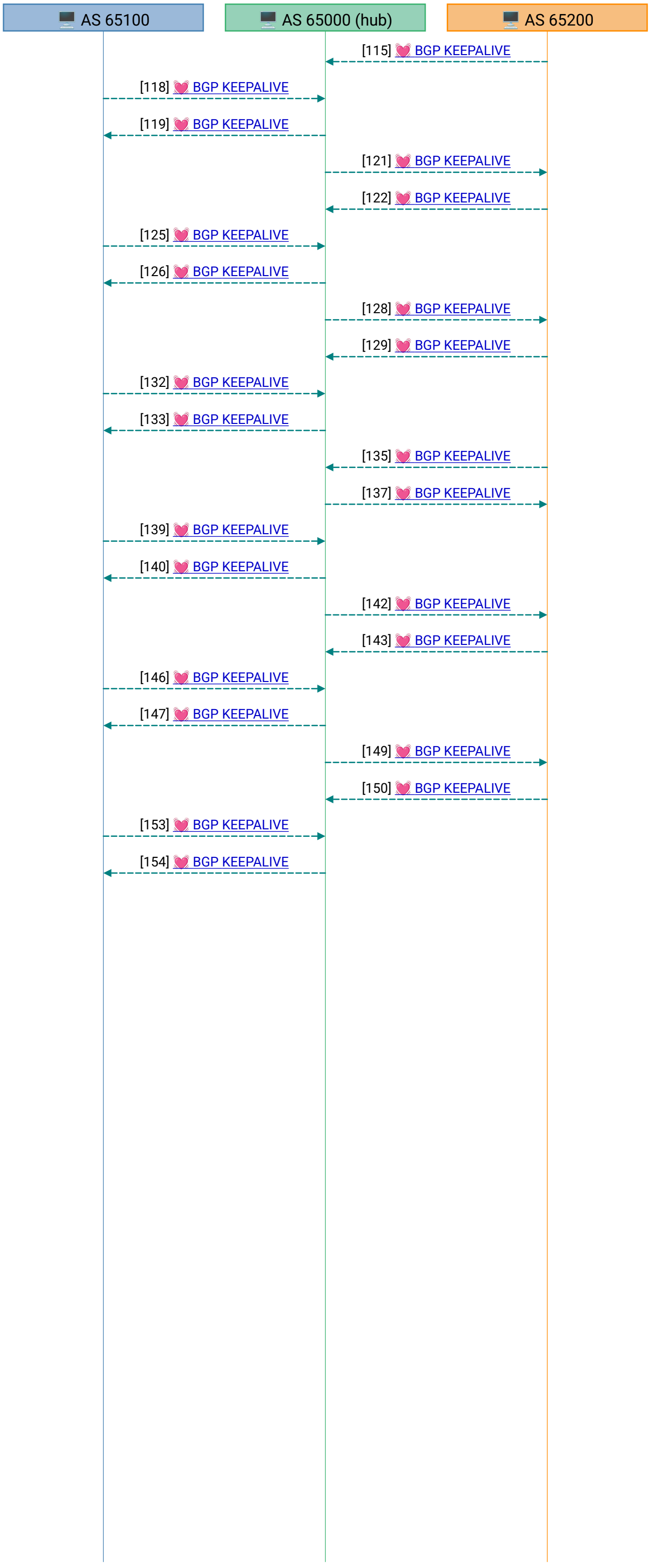
[Frame 111] KEEPALIVE, AS 65100 -> hub (.51).

[112] BGP KEEPALIVE

[Frame 112] KEEPALIVE, hub -> AS 65100 (.51).

[114] BGP KEEPALIVE

[Frame 114] KEEPALIVE, hub -> AS 65200 (.50).



[Frame 115] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 118] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 119] KEEPALIVE, hub -> AS 65100 (.51).

[Frame 121] KEEPALIVE, hub -> AS 65200 (.50).

[Frame 122] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 125] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 126] KEEPALIVE, hub -> AS 65100 (.51).

[Frame 128] KEEPALIVE, hub -> AS 65200 (.50).

[Frame 129] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 132] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 133] KEEPALIVE, hub -> AS 65100 (.51).

[Frame 135] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 137] KEEPALIVE, hub -> AS 65200 (.50).

[Frame 139] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 140] KEEPALIVE, hub -> AS 65100 (.51).

[Frame 142] KEEPALIVE, hub -> AS 65200 (.50).

[Frame 143] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 146] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 147] KEEPALIVE, hub -> AS 65100 (.51).

[Frame 149] KEEPALIVE, hub -> AS 65200 (.50).

[Frame 150] KEEPALIVE, AS 65200 -> hub (.50).

[Frame 153] KEEPALIVE, AS 65100 -> hub (.51).

[Frame 154] Last frame in the capture — a routine KEEPALIVE (hub -> AS 65100, .51 session). Both peerings are still Established and healthy; the capture simply **ends here,** with no NOTIFICATION and no TCP FIN/RST teardown. That is why the session tracker labels these two sessions `incomplete::no-close` — they never failed, the recording just stopped mid-flight.