ASPA-based AS path verification [1] examples

Kotikalapudi Sriram, Oliver Borchert, and Maria Matejka
Email: ksriram@nist.gov

August 2025

[1] "BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects," https://datatracker.ietf.org/doc/draft-ietf-sidrops-aspa-verification/

C2P = Customer to Provider
p2p = peer to peer (lateral peers)

ASPAs:

A, {C, D}
B, {E}
C, {F}
D, {F, G}
G, {AS0}

E and F have no ASPAs

The topology shown is ground truth, not deduced from ASPAs.

If ASPA exists for an AS, it only asserts if another AS is its Provider or Not Provider.

Topology

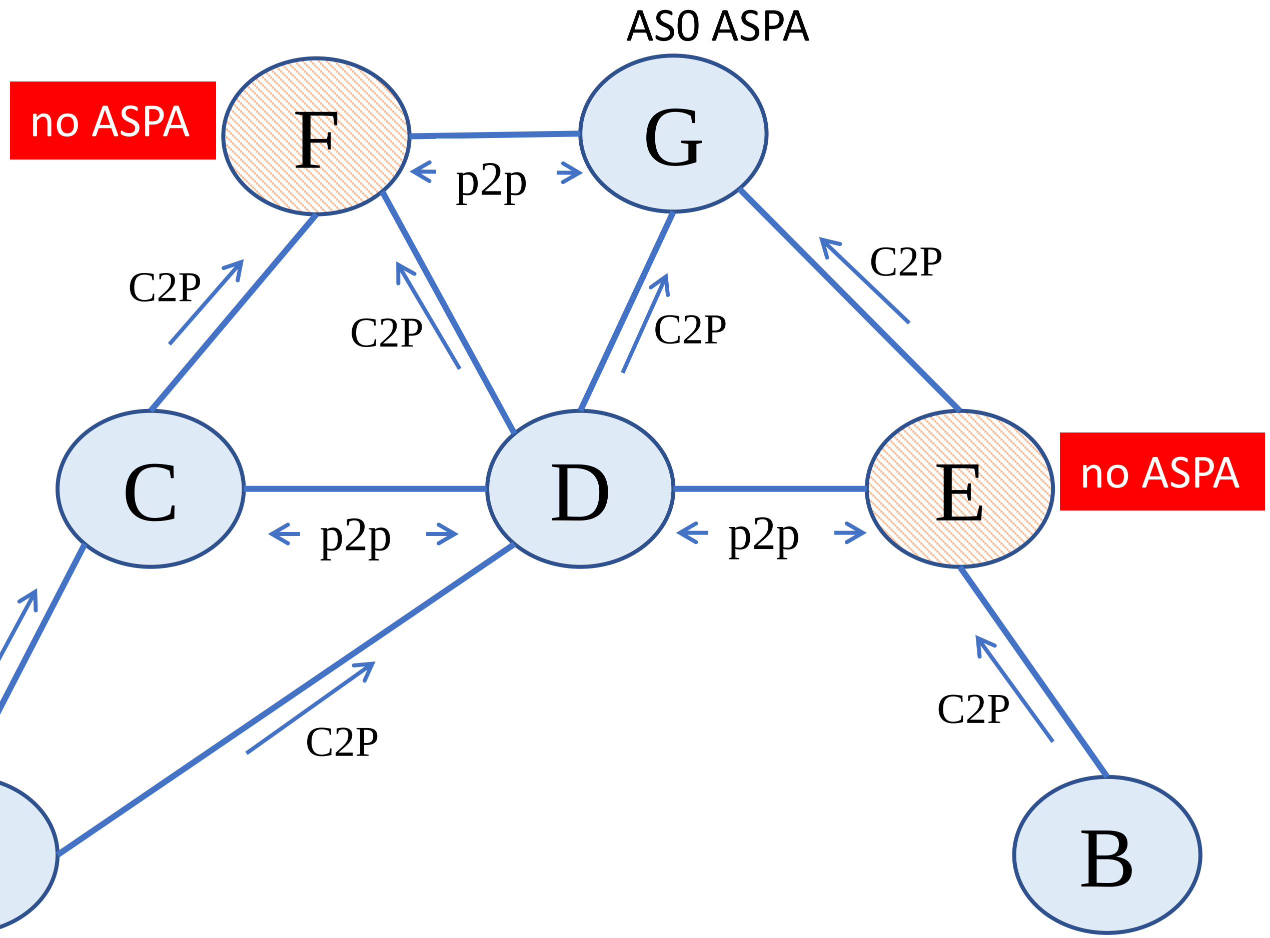


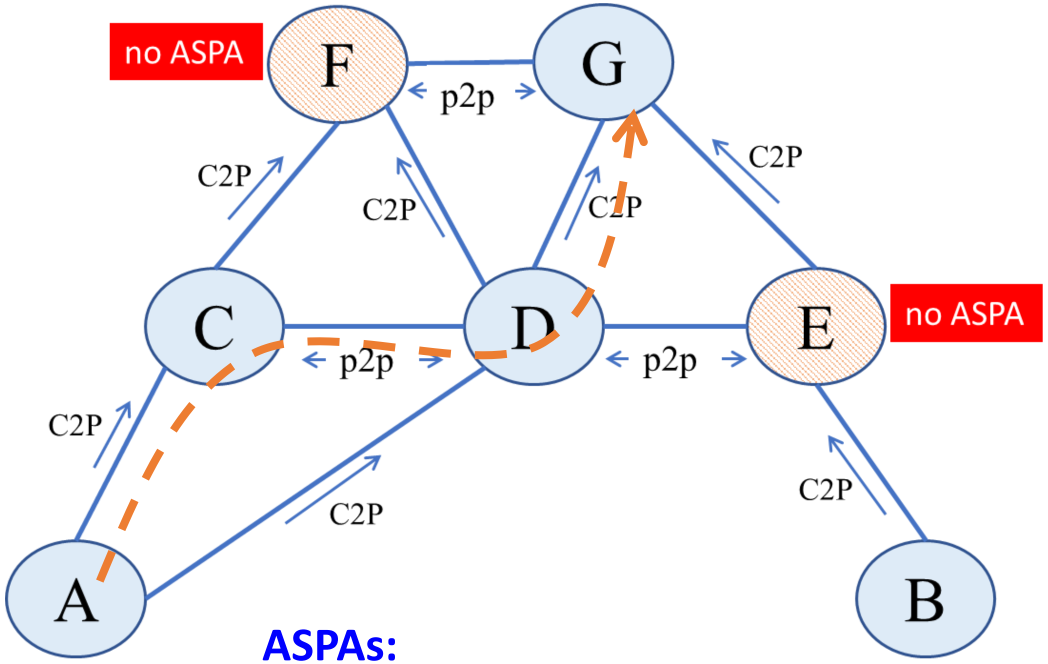
Table 1: Examples of Upstream Path Verification

(BGP update received from a customer or lateral peer)

#	Receiving/ Verifying AS	AS path (unique ASes)	AS path length	max_up_ramp	min_up_ramp	Upstream Path Verification Result
1	G	F C A	3	3	3	Valid
2	G	D C A	3	2	2	Invalid
3	G	D F C A	4	4	3	Unknown
4	C	D E B	3	3	2	Unknown
5	C	A D E B	4	3	2	Invalid
6	C	A D G E B	5	3	2	Invalid
7	D	A C F	3	2	1	Invalid
8	D	A C F G	4	1	1	Invalid
9	D	E B	2	2	2	Valid

Working out example #2 in Table 1 for upstream path

(BGP update received from customer or lateral peer)



ASPAs:

A, {C, D}
B, {E}
C, {F}
D, {F, G}
G, {AS0}

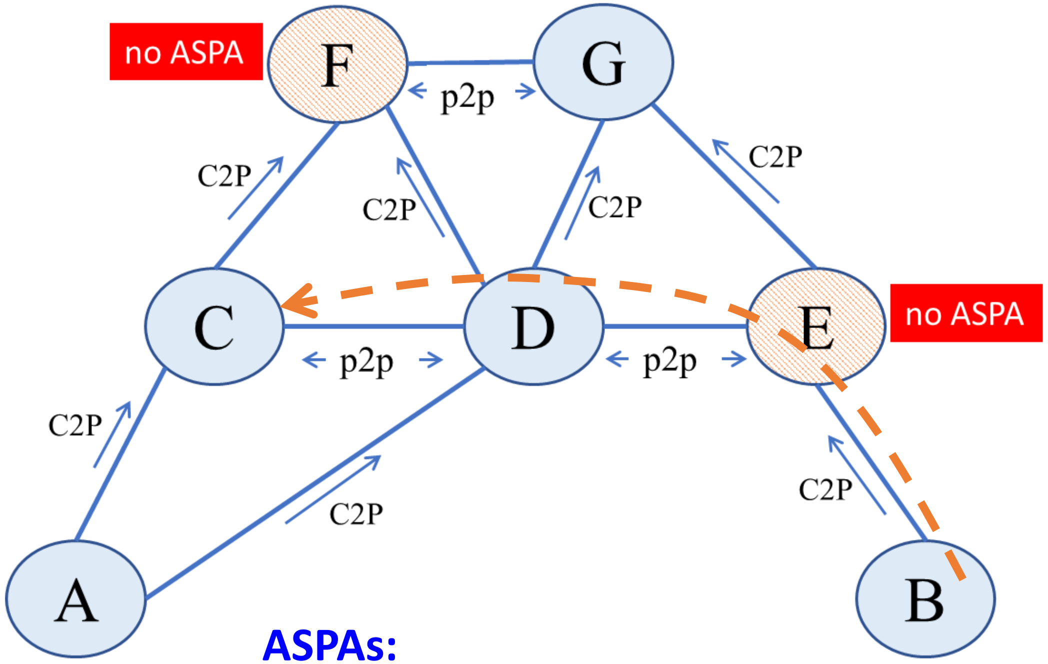
E and F have no ASPAs

- G is the receiving/verifying AS
- G receives BGP update from D with AS path: D C A (simplified to contain only unique ASes)
- AS path length = 3
- Using the given ASPAs and the method in Section 6.1 of [1]:
  - max_up_ramp = 2 (A C)
  - min_up_ramp = 2 (A C)
- Using the procedure in Section 6.2 of [1]:
  - max_up_ramp < AS path length; hence verification outcome: Invalid

[1] "BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects," https://datatracker.ietf.org/doc/draft-ietf-sidrops-aspa-verification/

Working out example #4 in Table 1 for upstream path

(BGP update received from customer or lateral peer)



ASPAs:
A, {C, D}
B, {E}
C, {F}
D, {F, G}
G, {AS0}

E and F have no ASPAs

- C is the receiving/verifying AS
- C receives BGP update from D with AS path: D E B (simplified to contain only unique ASes)
- AS path length = 3
- Using the given ASPAs and the method in Section 6.1 of [1]:
  - max_up_ramp = 3 (B E D)
  - min_up_ramp = 2 (B E)
- Using the procedure in Section 6.2 of [1]:
  - max_up_ramp is not less than AS path length; hence, AS path is not Invalid
  - min_up_ramp < AS path length; hence, verification outcome: Unknown

Table 2: Examples of Downstream Path Verification

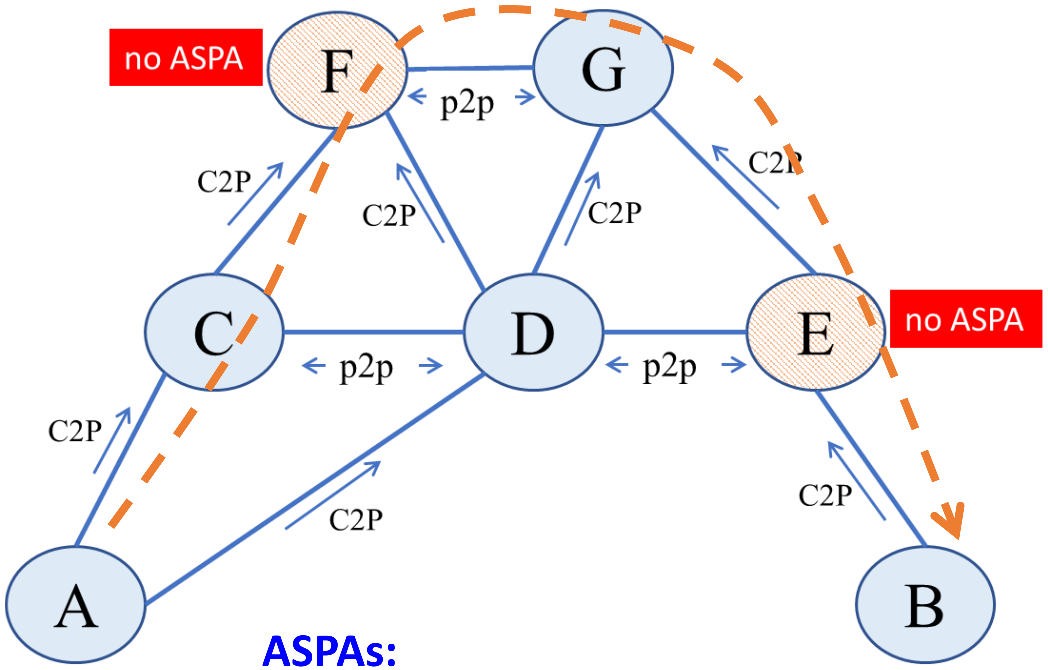
(BGP update received from a provider)

#	Receiving /Verifying AS	AS path (unique ASes)	AS path length	max_up_ramp	max_down_ramp	min_up_ramp	min_down_ramp	Downstream Path Verification Result
1	B	E G F C A	5	4	2	3	1	Unknown
2	B	E G D A	4	3	2	3	1	Valid
3	B	E D C A	4	2	2	2	1	Unknown
4	B	E G D C A	5	2	2	2	1	Invalid
5	A	C F D G	4	1	4	1	2	Unknown
6	A	D G E B	4	3	2	2	2	Valid
7	A	C D G E B	5	3	1	2	1	Invalid
8	D	F C A	3	3	2	3	1	Valid
9	B	E A	2	1	2	1	1	Valid*
10	B	E C A	3	2	2	2	1	Valid**

*Forged-origin and **forged-path-segment hijacks (path manipulations) by a provider (E) towards a customer (B) are undetectable by ASPA (see Section 8.2 “Security Considerations” in [1]).

Working out example #1 in Table 2 for downstream path

(BGP update received from a provider)



ASPAs:

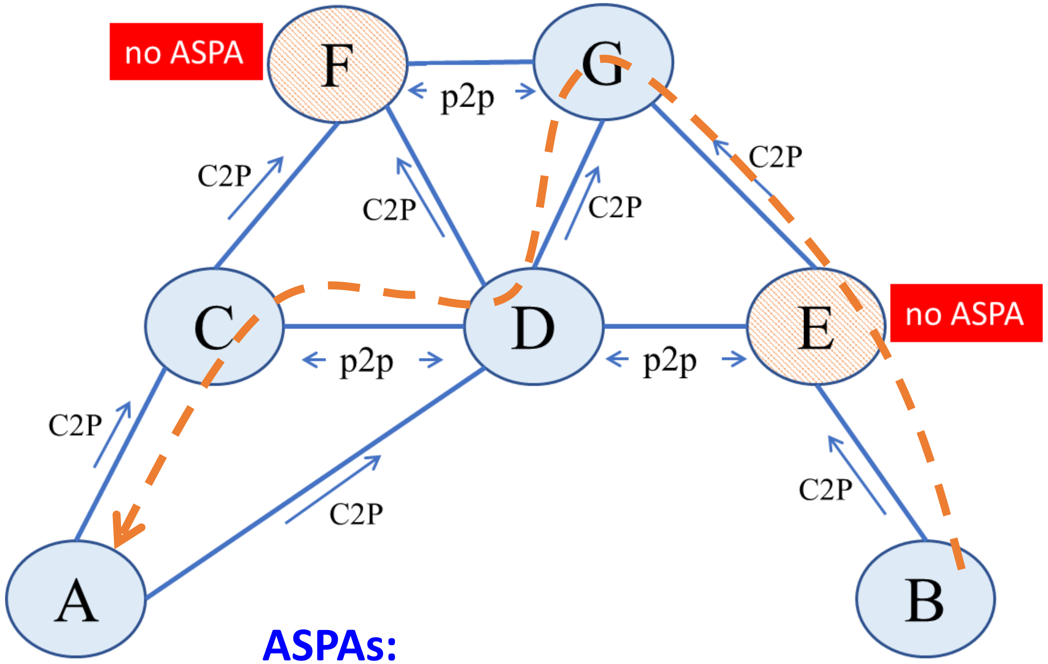
A, {C, D}
B, {E}
C, {F}
D, {F, G}
G, {AS0}

E and F have no ASPAs

- B is the receiving/verifying AS
- B receives BGP update from E with AS path: E G F C A (simplified to contain only unique ASes)
- AS path length = 5
- Using the given ASPAs and the method in Section 6.1 of [1]:
  - max_up_ramp = 4 (A C F G)
  - max_down_ramp = 2 (E G)
  - min_up_ramp = 3 (A C F)
  - min_down_ramp = 1 (E)
- Using the procedure in Section 6.3 of [1]:
  - max_up_ramp + max_down_ramp is not less than AS path length; hence, AS path is not Invalid
  - min_up_ramp + min_down_ramp < AS path length; hence, verification outcome: Unknown

Working out example #7 in Table 2 for downstream path

(BGP update received from a provider)



ASPAs:

A, {C, D}
B, {E}
C, {F}
D, {F, G}
G, {AS0}

E and F have no ASPAs

- A is the receiving/verifying AS
- A receives BGP update from C with AS path: C D G E B (simplified to contain only unique ASes)
- AS path length = 5
- Using the given ASPAs and the method in Section 6.1 of [1]:
  - max_up_ramp = 3 (B E G)
  - max_down_ramp = 1 (C)
  - min_up_ramp = 2 (B E)
  - min_down_ramp = 1 (C)
- Using the procedure in Section 6.3 of [1]:
  - max_up_ramp + max_down_ramp < AS path length; hence, verification outcome: Invalid

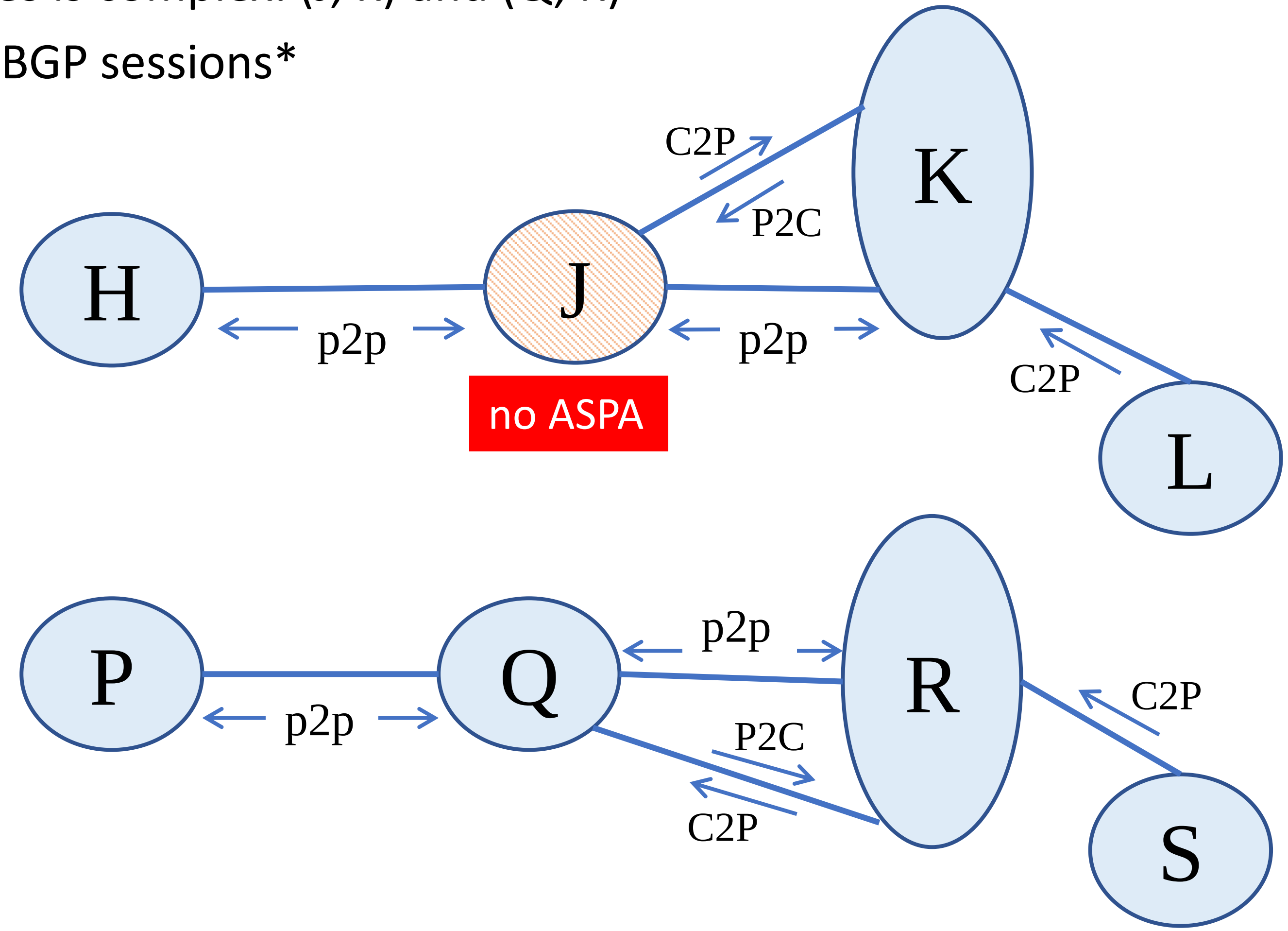
Examples - Topology Contains Complex BGP Relationships

- BGP relationship between neighbor ASes is complex: (J, K) and (Q, R)
- Different BGP relations exist in distinct BGP sessions*

ASPAs:

H, {AS0}
K, {AS0}
L, {K}
P, {AS0}
Q, {AS0}
R, {Q}
S, {R}

J has no ASPA



* See discussion in Sec. 7.4 of the Draft: https://datatracker.ietf.org/doc/draft-ietf-sidrops-aspa-verification/

Table 3: Examples - Topology Contains Complex BGP Relationships

#	Receiving/ Verifying AS	AS path (unique ASes)	Rec-to-Sender session type	Type of path verification	AS path length	max_up _ramp	max_down _ramp	min_up _ramp	min_down _ramp	Path Verification Result
1	K	J H	P2C	Upstream	2	1	NA	1	NA	Invalid
2	K	J H	p2p	Upstream	2	1	NA	1	NA	Invalid
3	L	K J H	C2P	Downstream	3	1	1	1	1	Invalid*
4	R	Q P	p2p	Upstream	2	1	NA	1	NA	Invalid
5	R	Q P	C2P	Downstream	2	1	1	1	1	Valid
6	S	R Q P	C2P	Downstream	3	1	2	1	2	Valid*

* Subsequent ASes in the path after a complex-relationship hop (e.g., AS L and AS S) can only go by the ASPAs. They do not know the existence of complex relations in the AS path. The ASPA method is designed to give a conservative answer in such cases (at subsequent ASes) rather than declare a route Invalid when it might be Valid.