

NLSR - Bug #5139

NLSR not reacting correctly on neighbor down

12/22/2020 10:10 AM - Ashlesh Gawande

Status:	Closed	Start date:	
Priority:	High	Due date:	
Assignee:	Ashlesh Gawande	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	v0.6.0		

Description

Regression was introduced in [#5009 \(https://gerrit.named-data.net/c/NLSR/+5745/11/src/hello-protocol.cpp\)](https:// Gerrit:https://gerrit.named-data.net/c/NLSR/+5745/11/src/hello-protocol.cpp).

Due to conflicting if and else:

[1](#)

[2](#)

the neighbor is not marked as INACTIVE.

Will need a unit test to stop this from happening in the future.

Furthermore, even if this was correct, in Hyperbolic routing mode the routing table calculation is not scheduled so NLSR takes a long time to reflect the status of the neighbor in the FIB.

History

#1 - 12/22/2020 10:10 AM - Ashlesh Gawande

- Description updated

#2 - 12/22/2020 08:03 PM - Ashlesh Gawande

- Subject changed from NLSR not reacting correctly neighbor down to NLSR not reacting correctly on neighbor down

- Status changed from New to Code review

- % Done changed from 0 to 100

#3 - 01/04/2021 08:48 PM - Ashlesh Gawande

- Status changed from Code review to Closed

#4 - 02/15/2021 10:28 AM - Ashlesh Gawande

Seems like for HR, it is not straight forward to remove the FIB entry completely in a short time.

Example:

c---a---b---d:

Once successfully converged, the routing table at **a** will have the following routes:

Routing Table:

Destination: /ndn/b-site/%C1.Router/cs/b

NextHop(Uri: udp4://10.0.0.2:6363, Cost: 0)

NextHop(Uri: udp4://10.0.0.6:6363, Cost: 1.41926)

Destination: /ndn/c-site/%C1.Router/cs/c

NextHop(Uri: udp4://10.0.0.6:6363, Cost: 0)

NextHop(Uri: udp4://10.0.0.2:6363, Cost: 1.41926)

Destination: /ndn/d-site/%C1.Router/cs/d

NextHop(Uri: udp4://10.0.0.2:6363, Cost: 0.935617)

NextHop(Uri: udp4://10.0.0.6:6363, Cost: 2)

FIB at **a** will contain the following prefixes:

/ndn/c-site/c nexthops={faceid=266 (cost=0), faceid=264 (cost=1419)}

/ndn/d-site/d nexthops={faceid=264 (cost=936), faceid=266 (cost=2000)}

/ndn/b-site/b nexthops={faceid=264 (cost=0), faceid=266 (cost=1419)}

After NLSR on **b** goes down, **a** will have the following routes - trying to route through **c** to **b**:

Routing Table:

Destination: /ndn/c-site/%C1.Router/cs/c

NextHop(Uri: udp4://10.0.0.6:6363, Cost: 0)

Destination: /ndn/d-site/%C1.Router/cs/d

NextHop(Uri: udp4://10.0.0.6:6363, Cost: 2)

Destination: /ndn/b-site/%C1.Router/cs/b
NextHop(Uri: udp4://10.0.0.6:6363, Cost: 1.41926)

And FIB:

```
/ndn/c-site/c nexthops={faceid=266 (cost=0)}  
/ndn/d-site/d nexthops={faceid=266 (cost=2000)}  
/ndn/b-site/b nexthops={faceid=266 (cost=1419)}
```

HR at **a** does not know that **c** is routing to **b** through **a** itself

(In LS, **a** will update its adjacency LSA and **c** will calculate accordingly).

So in the case of HR, for the route to be completely gone NLSR must wait for the expiration of Coordinate LSA (30 minutes by default).

(Note that on Router d the route will be completely gone in a short time).