bgpd: improve packet parsing for EVPN and ENCAP/VNC #21098

Merged riw777 merged 1 commit into `FRRouting:master` from `mjstapp:fix_bgp_parse_evpn_vnc` 2 weeks ago

Conversation 8 Commits 1 Checks 21 Files changed 3

mjstapp commented 3 weeks ago — Contributor

Improve packet validation for EVPN NLRIs and for ENCAP/VNC. Validate internal ip address fields against overall message length; impose stricter validation for VNC sub-tlvs in the rfapi code.

👍 1

frrbot bot added the bgp label 3 weeks ago

github-actions bot added size/M master labels 3 weeks ago

greptile-apps bot commented 3 weeks ago • edited

Greptile Summary

This PR hardens packet parsing for EVPN (Types 2, 3, and 4) and VNC/RFAPI by adding cross-validation between the internally-declared IP address length field and the overall NLRI wire-length, and by adding stricter sub-TLV length guards in the rfapi RIB path.

Key changes:

- **bgpd/bgp_evpn.c — Type-2**: A new `psize` vs. `ipaddr_len` consistency check is added *after* `ipaddr_len` has already been validated to be `0`, `32`, or `128`, so integer division is exact and no fringe values can slip through.

- **bgpd/bgp_evpn.c — Type-3** and **bgpd/bgp_evpn_mh.c — Type-4**: Similar cross-validation checks are added, but they are placed *before* the explicit bit-length guard against `IPV4_MAX_BITLEN` / `IPV6_MAX_BITLEN`. Because `psize` is independently constrained to two legal values before these checks run, any non-byte-aligned `ipaddr_len` that passes via truncating integer division (e.g., `33/8 == 4`) will still be rejected by the subsequent explicit bit-length comparison. There is no memory safety risk, but the ordering means the new guards are not independently sufficient on their own.
- **bgpd/rfapi/rfapi_rib.c**: Two new guards prevent processing a `BGP_VNC_SUBTLV_TYPE_RFPOPTION` sub-TLV whose `pEncap->length` is less than 3 (which would allow a zero-byte allocation and a potentially out-of-bounds `value[1]` read) or whose declared option length (`value[1]`) is zero. This closes a previously identified vulnerability where `XCALLOC(..., 0)` could be called, the allocation would succeed, and a warning-and-shrink loop would produce a `hop` with a zeroed length.

Confidence Score: 4/5

- This PR is safe to merge — all new validation checks are defensive guards that reject malformed input and do not introduce new memory safety issues.
- The changes add strictly defensive input validation. The rfapi fix is clean and complete. For EVPN Types 3 and 4, the new size check uses integer division before `ipaddr_len` is validated against canonical values, meaning non-byte-aligned values can pass it — but they are caught by the subsequent explicit bit-length check, so no unsafe memory operation can occur. The Type-2 check is ordered correctly and is fully robust. Score docked one point because of the ordering inconsistency between Type-2 and Types 3/4, which leaves the new guards in Types 3/4 not independently sufficient.
- No files require special attention; the logic is correct and no regressions are expected.

Important Files Changed

Filename	Overview
bgpd/bgp_evpn.c	Adds cross-validation of `ipaddr_len` against `psize` for Type-2 (after prior ipaddr_len validation — robust) and Type-3 (before ipaddr_len validation against valid bit-lengths — relies on subsequent check to reject fringe values). No memory safety issues introduced.
bgpd/bgp_evpn_mh.c	Adds `psize` / `ipaddr_len` cross-validation for Type-4 routes. The outer `psize` check (lines 830–836) against the two legal sizes already constrains psize; the new check further ensures the internal ip-len field is consistent with that size. Safe due to subsequent explicit bit-length check.
bgpd/rfapi/rfapi_rib.c	Adds two guards for `BGP_VNC_SUBTLV_TYPE_RFPOPTION`: rejects sub-TLVs shorter than 3 bytes (preventing out-of-bounds `value[1]` access and zero-byte allocations) and rejects a declared option length of zero.

Filename	Overview
	Allocation and memcpy are now always operating on at least 1 byte. Change is correct and complete.

Flowchart

Last reviewed commit: `7676cad`

greptile-apps bot reviewed 3 weeks ago

View reviewed changes

`bgpd/rfapi/rfapi_rib.c` Show resolved

`bgpd/bgp_evpn.c` Show resolved

`bgpd: improve packet parsing for EVPN and ENCAP/VNC` ··· ✓ `7676cad`

mjstapp force-pushed the `fix_bgp_parse_evpn_vnc` branch from **4369990** to **7676cad** Compare 3 weeks ago

Jafaral commented 3 weeks ago — Member

@Mergifyio backport stable/10.6 stable/10.5 stable/10.4 stable/10.3 stable/10.2 stable/10.1 stable/10.0

👍 1

mergify bot commented 3 weeks ago • edited

> backport stable/10.6 stable/10.5 stable/10.4 stable/10.3 stable/10.2 stable/10.1 stable/10.0

✅ **Backports have been created**

▶ Details

github-actions bot added the backport label 3 weeks ago

Jafaral commented 3 weeks ago — Member

@greptile review

👍 1

riw777 approved these changes 2 weeks ago

View reviewed changes

riw777 left a comment — Member

looks good

riw777 merged commit **4825b5b** into `FRRouting:master` 2 weeks ago

20 checks passed

View details

This was referenced 2 weeks ago

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21234

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21235

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21236

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21237

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21238

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21239

Merged

bgpd: improve packet parsing for EVPN and ENCAP/VNC (backport #21098) #21240

Merged

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21240 from FRRouting/mergify/bp/stable/10.0/pr-21098` ✗ `42677ae`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21239 from FRRouting/mergify/bp/stable/10.1/pr-21098` ✗ `e61ca81`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21238 from FRRouting/mergify/bp/stable/10.2/pr-21098` ✗ `924b46e`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21237 from FRRouting/mergify/bp/stable/10.3/pr-21098` ✗ `36dfaf6`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21236 from FRRouting/mergify/bp/stable/10.4/pr-21098` ✗ `e6bf2ec`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21235 from FRRouting/mergify/bp/stable/10.5/pr-21098` ✓ `9f4a9ac`

...

riw777 added a commit that referenced this pull request 2 weeks ago

`Merge pull request #21234 from FRRouting/mergify/bp/stable/10.6/pr-21098` ✗ `571312f`

...

Reviewers

greptile-apps[bot]

riw777

Assignees

No one assigned

Labels

backport bgp master size/M

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

3 participants