IETF95

Further Mitigating Router ND Cache Exhaustion DoS Attacks Using Solicited-Node Group Membership

Mark Smith
markzzzsmith@gmail.com

Problem

Router neighbor cache state resource exhaustion denial-of-service attack.

Remote attacker attempts to consume router ND cache resources by sending packets to non-existent destinations on the link.

Described in RFC3756, further described RFC6583 with suggested mitigations.

Solicited Node Multicast Group Mitigation

Proposal:

Use nodes' Solicited Node Multicast Group membership to reduce and try to make harder to find a router's exploitable Neighbor Cache resources.

Solicited Node Multicast Group (SNMG) Membership

Per RFC4291 and RFC6434, nodes are to join SNMGs for each of their addresses, using MLDv1 or MLDv2.

Solicited Node Multicast Group:

FF02:0:0:0:0:1:FF00::<lower 24 bits IPv6 address>

where address is either unicast or anycast.

Absence of SNMG

Absence of SNMG means no addresses on-link that map to it.

No point performing ND NS for an address that would map to an absent SNMG.

Method

1) Router collects on-link present SNMGs using MLDv1 or MLDv2.
2) Packet with unresolved D.A. arrives, SNMG for unresolved address is calculated.
3) Calculated SNMG is compared with the list of on-link SNMGs.
4) Perform ND NS if SNMG is present. If not, drop ND trigger packet. **ND Cache DoS further mitigated**.

MLD reliablity?

Two Modes:

Strict Mitigation Mode – drop ND NS trigger packets for unknown SNMGs when MLD known to be reliable.

e.g., enterprise, content provider networks.

Relaxed Mitigation Mode – drop ND NS trigger packets for unknown SNMGs only when DoS looks to be happening.

e.g., default for home gateways, public networks

Survey of MLD for SNMGs

The following send MLDv1 or MLDv2 joins for SNMGs:

- Linux: Chromecast v1 & v2, Android 6, Chrome OS, Fedora 20
- Windows: XP, Vista, 7, 8.1
- Apple: Mac OS X, iPhone (thanks to Fred Baker and Mark Prior)

“MLD Considered Harmful”, Antonios Atlasis et. al., presentation says MLD disabled for OpenBSD. (of course, I don't agree with the title :-))

Limitations

2^40 addresses maps to each SNMG in a /64 prefix. Attacker guessing or discovering an on-link SNMG can consume ND cache resources because ND NSes will be sent.

Privacy Addresses or Stable Opaque IIDs, both with random IIDs, mitigates guessing.

Discovery only possible if attacker is not blind to the success of the DoS. A link with /104 or shorter prefix length will have 2^24/16 million possible SNMGs.

So certainly not perfect!

Could be useful in addition to mitigations in RFC6583.

Next Steps

Current thinking on an implementation:

- pim6sd or mrd6 to collect SNMGs via MLD
- Linux kernel ND code modified to use multicast route table for SNMG checks

Suggestions appreciated (I'm certainly not an expert kernel hacker!)

Not specifically seeking v6ops WG adoption, certainly fine with that if that happens.

Some other thoughts

Hosts could implement this method if they were willing to participate in MLD as a “router”. ND cache attacker would be user on multi-user host – DoS is on other users of the host.

As MLDv2 does not supress reports, LL source addresses could be used to enumerate all on-link nodes' LL addresses (except OpenBSD!).

Address registation protocol by combining MLDv2 LL source collection with hosts supporting Node Information Queries or Inverse ND?