Arrangement of IPv6 Hop-by-hop options

draft-krishnan-ipv6-hopbyhop-00.txt

Suresh Krishnan

Ericsson

Vulnerabilities

- HBH options are especially suitable for launching DoS attacks because
  - All the nodes on the path need to process them
  - Unrecognized options don’t always result in ICMP errors
  - There is no limitation on the number of times an option can occur

The attack

- Send a datagram with a large number of option TLVs
- The option type identifiers need to be in the range 0x02 to 0x63 to avoid ICMP errors
- Low bandwidth requirement. Easier to overwhelm the control processor than the forwarding elements

Solution

- Restrict each option type to occur only once
- Proposed arrangement of options to optimize check for duplicates. Very low processor and memory overhead
- 2x-8x performance increase in worst case scenario
- Applicable to Destination options as well

Deployment issues

- Affects all existing IPv6 nodes.
- Not too many HBH options supported yet.
- Alternate solutions
  - Rate limiting (least impact)
  - Limit number of HBH options per HBH option header (lower impact)

Questions

- Should we worry about this?
- Where does this work belong?

Quantitative Analysis

The impact on existing nodes caused by this draft has to be viewed in light of the quantitative analysis of the improvement in processing cycles in the worst case scenario.The maximum size of the hop-by-hop option header is limited by the 8 bit Hdr Ext Len field. This field contains the length of the HBH options header in 8 octet units excluding the first 8 octets. So we end up with a theoretical maximum size of (2^8)*8+8==>2056 octets. The efficiency gains also depend on the number of valid options which are assigned by the IANA in the 0x65-0xff range.
Let's consider 'm' option type identifiers have been allocated in this range.

Common Parameters
Max hop-by-hop option header size : 2056 octets
Space available for options : 2054 octets (2 octets used for next hdr and header ext len)
Space occupied by smallest possible option : 2 octets (No Option Data!!)

Plain RFC2460 Hop-by-hop options header
Maximum number of options : 1027
CPU Cycles : 1027 * k

where k is a constant which denotes the number of cpu cycles required to process one option

Quantitative Analysis

After implementing the suggestions in this draft
All the legal options (m) can occur first then the attack options can occur. Since the attack options are in the range 0x02-0x63 there can at most be 62 of them. Each of these options can have at most 1 pad option after them. Therefore

Max number of options : 2 * (m+62) 0<m<192
CPU Cycles : 2 * (m+62) * k

The best gains are observed when m is low and decrease when more option types are allocated

m=0:
Processing speed= 1027/(2*62)=8.28x

m=192:
Processing speed= 1027/(2*254)=2.02x