IPv6/UDP
Zero-Checksum

Magnus Westerlund
Gorry Fairhurst
draft-ietf-6man-udpzero-01
Overview

› UDP for IPv6
› Should we change the behaviour?
› Checks required if relaxing checksum
› Next Steps
UDP for IPv6

- **Not** a solution to "just" make IPv6 like IPv4!
  - Specified only for tunnels

UDP with zero checksum does not always meet goals:
- *May*, get through firewalls, NAT
- Restricts deployability to systems that can be changed

- Impacts other systems and applications:
  - Reduced delivery protection (e.g. for other applications)
  - Not comparable with IPv4/UDP without checksum usage
Should we change the behaviour?

Section 1.2.4: What if zero UDP-checksum is used?
- What types of middleboxes need to be crossed (NAT, firewalls, etc.).
- How will those middleboxes deal with these packets?
  - What do IPv6 routers do today with zero-checksum UDP packets?
  - What other IPv6 middleboxes exist today?
  - What would they do?

Section 1.2.5
- Would ECMP be suitable for load-balancing LISP/AMT?

- The IETF should carefully consider constraints on sanctioning the use of the zero checksum mode
  - Current draft recommends UDP or UDP-Lite
Checks required if relaxing checksum

› 1. MUST verify integrity of inner (tunneled) packet
› 2. Non-IP inner (tunneled) packets MUST have a CRC or other mechanism for checking packet integrity
› 3. MUST define handling for default nodes (i.e. discard)
› 4. MUST NOT allow host fragmentation
› 5. MUST implement tunnel egress rules
  – Includes MUST NOT allow recursive fragmentation
› 7. Nodes MUST by default use original behaviour, probably requires a host “API” change to allow zero-checksum.
› 8. API SHOULD NOT wild-card the source \{any,dst\} ?
Next steps

› Next revision will:
  – Looking for inputs on middlebox behavior
  – Clarify ground rules (previous slide)

› WG may now “understand” the issues and caveats:
  – do we *wish* to go ahead and make the recommendation to allow this for consenting applications?

› Please read and comment on the draft
Extra Slides
Why is this being discussed?

› There is a proposal is to allow turning off the UDP checksum for IPv6, i.e. set it to 0.
  – Only for specific applications, especially tunneling usage.

› This was a result of two IETF protocols under development:
  – Automatic IP Multicast Without Explicit Tunnels (AMT) (draft-ietf-mboned-auto-multicast)
  – Locator/ID Separation Protocol (LISP) draft-ietf-lisp

› A checksum change was/is proposed in:
  – draft-eubanks-chimento-6man-00

Note: A more detailed presentations was previously made to 6man saying why this draft is needed.
Perceived needs of LISP and AMT

› LISP and AMT are both tunneling mechanisms
  – Don’t require the UDP checksum to verify data corruption of inner packet, because that will be verified at delivery after de-capsulation

› IP in IP tunneling would work if not for the additional requirements:
  – ECMP
  – Firewall traversal – \textit{BUT uncertain whether v6 Firewalls of NATs would currently support a zero checksum}

› UDP-Lite would work,
  – \textit{BUT} limited firewall traversal (especially for IPv6)

  – \textit{midbox traversal may need to be defined for any UDP Update !!!}
Understanding the Impact

› UDP is an end-to-end transport working on host nodes
› Impact of outer IP header corruption with zero UDP-checksum
  – Corrupted destination delivers to random host, different stack
  – Corrupted source makes it look like it comes from a different source
    › Impact depends on application and OS stack.

› Issues and recommendations described in current WG draft.
AMT

› Uses UDP tunnels between an AMT relay router and an AMT gateway
   – AMT Gateway is either a site gateway router or host

› UDP chosen for FW traversal

› The issue is the encapsulated multicast data in UDP + AMT header
   – Substantial amounts of data
   – Some routers can’t calculate a UDP checksum over a complete packet
     › Don’t have access to the complete packet when encapsulating
LISP

▪ Encapsulates any IP packet in an IP/UDP/LISP packet between the Ingress Tunnel Router (ITR) and Egress Tunnel Router (ETR).

▪ The ITR and ETR can be at different locations from site boundary to last hop routers.

▪ Reasons for using UDP:
  – To allow deployment on routers that can’t access the whole packet when doing encapsulation
  – Equal Cost Multi-Path (ECMP) operations
    ▶ IPv6 Flow label is seen as difficult to use for this purpose
    ▶ UDP ports are a part of the hash
IPv4 vs IPv6

› RFC 2460, section 8 says:
  – Unlike IPv4, when UDP packets are originated by an IPv6 node, the UDP checksum is not optional. That is, whenever originating a UDP packet, an IPv6 node must compute a UDP checksum over the packet and the pseudo-header, and, if that computation yields a result of zero, it must be changed to hex FFFF for placement in the UDP header. IPv6 receivers must discard UDP packets containing a zero checksum, and should log the error.

› Using zero-checksum is allowed in v4, but not in v6:
  – The removed IP header checksum resulted in loss of
    › delivery protection, i.e. ensuring that it is delivered to the correct right destination address and with correct source address
    › verification of next header field
  – In v6, the above are verified through the transport checksum pseudo header at the end of the delivery, rather than for each hop.
Usages
END HOST Impact

› A packet with a corrupted destination arrives at its new target
  – Where it is processed by the UDP stack:
    › This will likely drop it as it has an illegal checksum value
      - Assuming an unchanged host.
    › If the IP and UDP layer is not well-integrated or the receiving host has been changed, it will be forwarded to application
    › Depending on application, possibly may determine this as corrupt data it will (or will not) process.
    › Depending on application, may also modify/create protocol state.

› A host that turns off checksum as a result of allowing this:
  – Has lost its delivery protection
  – Will be 32000 times more likely to get unintended packets delivered to applications
Tunnel USAGE Impact

› Uncertain that IPv6/UDP with zero checksum will be passed by firewalls:
   – Packet is not according to RFC2460 and may therefore be considered dangerous or a waste of bandwidth by middlebox

› Turning off the checksum in some host operating systems/routers/CPEs is not possible or affects the whole system:
   – Margaret Wasserman said on LISP mailing list that this applies to major host operating systems and most checksum offloading hardware in hosts or CPEs.
   – Does not apply to all router cases, but the egress for some use cases are CPE or end-user hosts