# draft-ietf-pim-eckert-rfc1112bis-06

## PIM-WG, IETF124 Montreal

Steve Deering, retired

Toerless Eckert, Futurewei USA (tte@cs.fau.de) (Editor)

# Status update

- 2nd WGLC started since -05 / IETF123. Received review feednback from Stig
  - (thanks a lot)

# -05 -> -06

- Removed erroneous statement about IGMPv2/MLDv3 and UDP api

- Note to reviewers (to-be-removed) – reinforce notion we want no editorial improvements on existing text sections, just technical necessities.

- All of Stigs feedback
  - Clarified IP multicast address means ASM Host Group address OR SSM channel Destination Address ONLY ACTUAL TECHNICAL/SEMANTIC ENHANCEMENT!
  - Fixed hopefuly all occurances of „host group“ -> IP multicast address where this applies
  - Various text nits
  - Acknowledgements
  - Security considerations…

- 6 new pages of Security Considerations
- 2 pages new appendix sections for Application Socket Security Considerations
  - Changed order of authors so SteveD is not primarily blamed for all the new text…

# Security Considerations (1)

- Really the best/only document to elaborate on the issues. Did not find any other document that covers them comprehensively
  - All the core stuff „unicast“ stack/application developer can do wrong with multicast – and did.
  - Host stack beyind IP module out of scope -> See new appendix section
- 12.1 Network forwarding issues
  - Only host stack, ASM service model and some L2 issues. Multicast routing issues out of scope
  - Example given (shameless BIER marketing): State attacks against network forwarding only for stateful forwarding protocols but not e.g.: L2 or BIER
- 12.2 Senders can not control who can receive their traffic
  - „Secrecy of address“ (IPv6)
  - Encryption+key distribution to authenticated receivers (GDOI)
  - Filtering in the network (as used for high-value multicast in e.g.: pay-per-view IPTV)
  - Against group key sharing: TEE / DRM

# Security Considerations (2)

- 12.3 Sender control – receivers can not control/limit sender
  - Address privacy – but no analysis for IP multicast like RFC7721 for unicast
  - SSM solves this problem
  - SSM is even better than unicast: stops unwanted traffic at the source (with appropriate network forwarding)
- 12.4 Packet spoofing
  - Who needs packet spoofing when you can just be another ASM sender!
  - ASM model also makes discovery of address (sender IP/MAC) easy to discover
  - Network forwarding details may change this (when IP multicast is not broadcast)
  - Outside scope: Stateful network forwarding makes spoofing often harder than unicast spoofing because RPF check is mandatory
  - Authentication of sender harder than unicast. Public key signing computationally expensive, shared key cryptography does not work.
  - Asked on MSEC and collected/listed known IETF attempts to solve: TESLA (RFC4084) plus many newer RFC/drafts.

# Security Considerations (3)

- 12.5 Address managmenet
  - Unique ASM group required per ASM application. Static allocation, old RFC (MASC, MADCAP, ...) not widely adopted
- 12.5.1 Waste traffic when ASM groups are re-used by apps
  - Host must receive traffic for all apps and somehow filter unwanted app traffic
  - If thats not expected, compatible app formats may even cause app malfunction / security issues
  - May also lead to network path overload when traffic planning does not expect this issue but assumes only desired app traffic reaches hosts.
- 12.5.2 Waste traffic due to layer 2 to layer 3 mapping
  - Explains with expplicit IPv4 address patterns given (224.128.0.251 -> 224.0.0.251)
  - Example: high-bitrate video overloads low power IoT device, breaks e.g.: mDNS on it
- 12.5.3 Multiple application instances
  - If not designed for, traffic from one instance will leak into the other instance
  - Example: OSPF. No instance notion. Mis-wiring of supposedly disjoint OSPF instance networks causes network routing breakdown. Solution: isolate instances by different authentication password (actually the only use of otherwise really insecure password in OSPF).

# Security Considerations (4)

- 12.5.3 Multiple application instances
  - If not designed for, traffic from one instance will leak into the other instance
  - Example: OSPF. No instance notion. Mis-wiring of supposedly disjoint OSPF instance networks causes network routing breakdown. Solution: isolate instances by different authentication password (actually the only use of otherwise really insecure password in OSPF).
- 12.6 MAC filters
  - DoS against total number of sockets: Like unicast except for hardware resources (MAC filters).

# Application Socket Security Considerations (1)

- New appendix because this is really out of scope – but there is no better place than this RFC to capture it.
- Legacy ASM host stack would forward all joined groups to all UDP sockets – because there was no IP Multicast support in socket layer
- IGMPv2/MLDv2 fix this – If implementers correctly implement one well hidden sentence:

  *After a multicast packet has been accepted from an interface by thebIP layer, its subsequent delivery to the application or process that listens on a particular socket depends on the multicast listening state of that socket…*

- Level 2L
  - Would have to do the same as Level 2 (IGMPv3/MLDv2)
  - Should be a requirement but will not make it one because it raises additional new requirement, which risks (Full) Internet Standard status. All work so far on RFC1112bis was to capture all known well deployed practices/RFCs only!

# Application Socket Security Considerations (2)

- Application Socket issues
  - Slide Rant Summary: BSD socket model was never overhauled well for IP Multicast security. It sucks!
    - Source Port, Destination Port and bind() have different effects on Unicast vs. Multicast
    - Designed for Unicast, suxx for Multicast!
  - Binding to (UDP) port on client/receiver can prohibit receiving IP multicast from sender/server – not possible in unicast
    - SO_REUSEADDR/SO_REUSEPORT allows to avoid this, but you can not force a malicious app to use it!
  - Multicast Sender can not prohibit that a malicioous IP multicast sender app on same host re-uses same IP-address and sends to same multicast group. Unicast server/sender do not have this issue. They can bind to „server port“ and be guaranteed that clients will never connect to another app on the server for this (address,port) combinatio.
    - Applications could look in multicast at different sender ports but don‘t do it (e.g.: RTP)
    - Sender app can not even see malicious application traffic itself because IP_MULTICAST_LOOP allowsto prohibit sending packets from one IP multicast app on a host to any other app on the host.

# Next Steps

- Before closing WGLC, i suggest to get EARLY reviews
  - So that the doc is not yet out of WG hands

  - SECDIR
    - Can constrain itself to primarily Security Considerations + Application Socket Security Considerations
  - INTDIR
    - this doc is primarily an extension to IPv4/IPv6 protocol. This is really the realm of INT: 6MAN(IPv6) / INTAREA(IPv4)
    - Even though whole doc applies, these two new security sections might also be good starting point
  - Transport Area DIR
    - they mostly care about congestion control but they are really also responsible for the aspects in „Application Socket Security Consideations“ ((de)multiplexing multiple applications traffic in transport layer)
    - Specifically validate claim in draft: The following considerations are not covered by [RFC8085] or resolved trough the requirements specified by [TAPS] RFCs.

# Next Steps

- Suggested reviewers in review request
  - SECDIR: Brian Weis, Kyle Rose, Dave Thaler
  - TSVDIR: Michael Tüxen, Kyle Rose
  - INTDIR: Jen Linkova, Rick Taylor, Pascal Thubert, Michael Tüxen, Dave Thaler
- Not clear if requested reviews would be early

# The End