Teredo Extensions

draft-thaler-v6ops-teredo-extensions-05.txt

Dave Thaler
dthaler@microsoft.com
History

- Teredo documented in RFC 4380
- At IETF71, URL for Microsoft doc posted to the list
- This documented what Windows XP/Vista/Windows 7 already does
- It contains several things:
  - Security updates to RFC 4380:
    - Now in draft-krishnan-v6ops-teredo-update
  - Support for more NAT types:
    - Now in draft-thaler-v6ops-teredo-extensions
  - Efficiency (server load) improvement:
    - Now in draft-thaler-v6ops-teredo-extensions
Current Status

• Track: AD-sponsored individual submission through INTAREA
• Last presented in V6OPS at IETF 73
• WG Last call done in V6OPS due to history
• Several issues raised:
  – Length computation
  – Handling of unrecognized trailers
  – State refresh constraints
  – Various typos and wording fixes
Length Computation

• RFC4380 said receiver checks that IPv4 UDP datagram length and IPv6 packet length are “consistent”
  – Rémi Denis-Courmont pointed out this was ambiguous and implementations might interpret as “equal”, causing interop problems
  – Clarified that “consistent” means ≤, to allow for trailers
Unrecognized Trailer Type Handling: How do you handle unrecognized Type values?

<table>
<thead>
<tr>
<th>Type</th>
<th>Length</th>
<th>Value (variable)</th>
</tr>
</thead>
</table>

- Similar problem to unrecognized IPv6 options (except no ICMPv6 errors since this is below IPv6)
  - Separate type ranges for skip vs discard
  - Allows for future extensibility
- Discard if high two bits of Type == 01, else skip
- Currently all defined types start with 00
State Refresh

- Purpose of state refresh is to keep mappings in your NAT(s) and hence keep Teredo address when idle
- RFC4380 required a nominal interval of 30 seconds, with optional procedure to learn a longer interval to reduce Teredo server load
  - Actually randomized within 75-100% of above, to avoid sync
- Problem 1: (rare) NATs flush state < 30 seconds
  - Resolution: nominal interval SHOULD be configurable and MUST default to 30
- Problem 2: some implementations randomize across a larger range (e.g. 50-150%, per classic Van Jacobson paper)
  - Resolution: MAY use an implementation-specific range that MUST fall within 50-150%
Next Steps

• Proceed with AD (Jari) sponsored submission through INTAREA