Extended Shim6 Design for ID/Locator Split and Traffic Engineering

draft-nordmark-shim6-esd-01.txt

Erik Nordmark
erik.nordmark@sun.com

Three separate pieces

- Complete ID-locator separation
  - Unreachable Upper-layer IDs
- Traffic engineering
  - Managing semi-static TE per site
  - More dynamic TE control
- Running over IPv4 locators

Why discuss this draft?

- To determine whether the proposed extensions could increase the applicability of shim6
- Understand relationship to shim6 proxies
- As input to a rechartering discussion?

Complete ID-Locator Separation

- The shim6 protocol mechanisms don't assume the ULID is reachable
  - But it is a key optimization and necessary for deferred context establishment
- If we had a unreachable ULID format that fits in 128 bits, then the existing (socket) APIs can be reused
- If we also could lookup a ULID to find a set of locators, the application could use referrals and callbacks as today
- Many possible details – have examples

Unreachable ULID format

- Pick a relatively short prefix from the IPv6 address architecture
- We only know how to do scalable lookups from a hierarchically allocated “name”
  - Think of 10^15 hosts using this scheme
- The result is something very similar to the centrally assigned unique-local addresses
  - Don't know if CULA will be resurrected
  - We would be using HBA/CGA for the bottom 64 bits to handle security

Need for Lookup of ULID?

- If a ULID is reachable we can just send packets to it to find out the set of locators
- For unreachable, we need a way to get packets flowing by first finding some locators for peer
- We could piggyback this on the DNS lookup of www.example.com. But that is insufficient since
  - The shim6 state might be lost and the ULPs just have the ULID
  - Application referrals, callbacks and long-lived application handles

Example: Using DNS

- Host looks for ID RRtype for www.example.com
  - Result is a 128 bit unroutable identifier
  - If no ID RR, looks for AAAA just like with shim6 today
- The ULID is mapped to locators using a reverse lookup in e.g., in the ip6.arpa tree
  - Could be creative and place AAAA records in the reverse tree
  - Could be even more creative and place SRV records in the reverse tree in order to express static priority and weight

Example: using DNS

- Syntax:

_Service._Proto.Name SRV Priority Weight Port Target

- Example:

```
$ORIGIN 10.6c.36.fe.ff.6b.0b.02.bc.00.9a.00.78.56.34.12
.ip6.arpa.
; 3/4 on to fastpath locator, 1/4 on slowpath
_shim6._ip      SRV 0 1 0 slowpath-www.example.com.
                SRV 0 3 0 fastpath-www.example.com.
; fallback if the above are broken
_shim6._ip      SRV 1 0 0 fallback-www.example.com.
```

Walkthrough (1)

- Application calls getaddrinfo() which finds ID RR
  - returns this as the IPv6 address to the application
- Application calls connect/sendto
- TCP/UDP sends packet to IP
- Shim looks at packet and sees the "unreachable ULID prefix"
  - Looks for shim6 context state
  - If none found, must setup context state before sending TCP/UDP packet

Walkthrough (2)

- Shim6 does DNS lookup of ULID to find set of locators
  - Can take priority and weight into account if we have a SRV like capability
- Shim6 uses new ULID-pair option
  - No other changes to shim6 protocol; sends I1 etc
- If one locator doesn't work for the context setup, then try other locators at the shim
- Once the context is established, again shim6 works unchanged
  - Might need to carry ULID-pair option on keepalive and probe messages etc.

Shim6 Traffic Engineering

- Can already carry priority and weight (defined as for DNS SRV records) for the locators once the context is established
- But no way for the host to know what values to use for its locators
  - And manual configuration not likely to be sufficient
- Could easily define a DHCPv6 option to allow side-wide configuration
  - Might be useful
  - Can use with stateless address autoconfiguration

Semi-Static Traffic Engineering

- Need some TE input before the shim6 context is established
- Possible to use DNS SRV for the application protocol
  - E.g., _http._tcp type SRV records
  - Requires application changes in most cases
- If non-routable ULID, see previous slides
- Combined with the DHCPv6 option, this provides the site with the ability to specify static load spreading wights and primary/fallback locators

Dynamic Traffic Engineering?

- A possibility would be to add support for routers rewriting (source) locators on shim6 packets
  - Based on idea in Mike O'Dell's GSE draft
- Shim6 (more or less by accident) allows this on packets that have the Payload Extension header
- We could add this for shim6 control messages
  - I-D has example “Sent locator-pair” and “Received locator-pair” options so hosts can learn from routers
  - These are used on I1, R1*, I2*, R2 and perhaps other shim6 control messages

Locator rewriting by routers

- Routers would be free to rewrite every packet with
  - nextheader == IPPROTO_SHIM6
  - thus every ULP packet should have payload ext hdr
- If the ULID is CGA, then the hosts can learn new locators from the routers based on the rewriting
- There are issues around which locator to use
  - probe mechanism says that A1 works and A2 fails
  - routers rewrite the source to be A2
  - Need hysteresis?

IPv4 addresses as locators

- Observation
  - If apps are using IPv6 socket API, and ULID is CGA
  - Then the locator can be anything (that is known to the local host and meaningful to the peer)
- Thus we could easily define a way to carry IPv4 addresses as locators
  - Could e.g., be IPv4-mapped address format
- Note: this does not “solve IPv4”, since IPv4 is likely to have NATs
  - But might be useful with proxies, if proxies have globally unique IPv4 addresses

Conclusions

- Using non-routable ULIDs doesn't place any new requirements on the shim6 mechanism
  - Need to discuss DNS vs. some other lookup system
- Would it be useful for TE to have router-rewriting?
  - If so, we can work on details (hysteresis)
- IPv4 locators (with CGA ULID) would be easy
  - But leave NAT discussions outside the door (in some other WG please)
  - Are they useful?