Thoughts on a New Namespace

Ran Atkinson
Presented by Steve Blake
“Standing on the Shoulders of Giants”

• Computer Science sometimes has been accused of blindly reinventing the wheel.

• We actively tried to avoid that, so credit to:
  ▶ Dave Clark for [c.1995] email to a public mailing list proposing to split the IP address into two pieces.
  ▶ Mike O’Dell for two early proposals (8+8, GSE), in the 1990s.
  ▶ The IRTF Name Space RG (NSRG), c. 1999-2002.

• This work extends and enhances those early ideas:
  ▶ Like HIP, this work dates back to the author’s participation in the IRTF NSRG early this decade.
Architectural Claim

If we provide a richer set of namespaces then the Internet Architecture can better support mobility, multi-homing, and other important capabilities:

- provide a broader set of namespaces than at present.
- reduce/eliminate names with overloaded semantics.
- provide crisp semantics for each type of name.
Effects of APIs

• Most C programmers still use the BSD Sockets API
  ▸ Sockets API does not itself support DNS
  ▸ This forces Applications to call into DNS Resolver, hence forces them to be aware of IP addresses and other low-level details

• Most Java programmers use a DNS-aware API
  ▸ Java designers carefully used data-hiding and abstraction in their API design
  ▸ Applications are aware of DNS names, but not aware of IP addresses or other low-level details
  ▸ Encourages more abstract application protocol design
What to do?

- Revisit the naming architecture of the Internet
  - Applying what we have learnt over 2+ decades
  - The IRTF Namespace RG focused on this topic.
- Consider adding additional namespaces
  - Network-layer host identifiers (not used for routing)
  - Service Names
  - Others also, perhaps.
- This talk focuses on how Network-layer host identifiers can help solve some parts of the architectural gap.
Some Existing Namespaces

- IP Address
  - 128.60.80.2
- IP Subnetwork
  - 128.60.80.0/24
- Domain Name
  - itd.nrl.navy.mil
- Communication Endpoint ("Socket")
  - TCP port 25 at itd.nrl.navy.mil
- Mailbox
  - username@itd.nrl.navy.mil
- URL
Routing RG
Issues
Scalability

- Growth in prefixes inside the Default Free Zone (DFZ) is at least geometric at present.
- Primary cause is growth in site multi-homing, which is also at least geometric at present.
- Primary goal of multi-homed sites is higher availability.
- Important reference for the above data:
Multi-Homing

- A fundamental issue is that current site multi-homing creates additional entropy in the DFZ RIB/FIB
- Why?
  - We multi-home sites using Longest Prefix Match
  - Each multi-homed site adds more-specific prefixes to DFZ
- Why this approach for multi-homing?
  - Transport-layer pseudo-header checksums include location information, not just host identity
- The real fix is to de-couple the transport protocol state from the network location.
Mobility

- Actually, mobility is just highly dynamic multi-homing
  - Want transport-layer session(s) to remain up
  - But want to change the network location of participant(s)
- Again, the cleanest fix is to de-couple the transport session state from the network location(s)
  - Mobile IP\{v4, v6\} try to hide the real network location through Home Address, Tunnelling, and other mechanisms.
    - An assumption for Mobile IP was that one could not change the architecture.
    - ILNP assumes the architecture can be changed.
Heresy

• The Internet’s routing architecture is actually just fine.
• The problem is that we are (ab)using routing to work-around limitations in the Internet’s naming architecture.
• If we can sort out the naming architecture, then the existing routing protocols and techniques will be fine and don’t need to change.
ILNP:
An 8+8 Approach
What is 8+8?

- 1) Name of an addressing architecture that split the IP address into a separate Locator and Identifier.
  - from Mike O’Dell in the middle 1990s.
- 2) An specific proposal on how to enhance IPv6; sometimes this is also called “GSE”.
  - Also from Mike O’Dell in the 1990s
- 3) A class of IP architectures that is based on the original concept from (1) above
  - In this talk, we are using definition (3) just above.
The 8+8 Architecture

- Separate the high-order bits (“Routing Prefix”) of an IPv6 address into a Locator field, 64 bits wide.
- Separate the low-order bits of an IPv6 address into an Identifier field, 64 bits wide.
- Transport session state contains only the Identifier.
- IP packet forwarding/routing uses only the Locator.
- One can imagine a range of networking protocols, different in various details, that use this architecture.
**ILNPv6**

- We propose an set of enhancements to IPv6, which we call **ILNPv6**:
  - provides full backwards compatibility with IPv6.
  - provides full support for incremental deployment.
  - **IPv6 routers do not need to change.**
- ILNPv6 “splits” the IPv6 address in half:
  - **Locator (L)**: 64-bit name for the subnetwork
  - **Identifier (I)**: 64-bit name for the host
- Same architecture can work for IPv4 (ILNPv4),
  - but a shortage of bits makes the engineering ugly
IPv6 Packet Header

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |   Next Hdr    |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
|Version| Traffic Class |           Flow Label                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Payload Length        |   Next Hdr    |   Hop Limit   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
Source Address

Destination Address
```
### ILNPv6 Packet Header

<table>
<thead>
<tr>
<th>Version</th>
<th>Traffic Class</th>
<th>Flow Label</th>
<th>Payload Length</th>
<th>Next Hdr</th>
<th>Hop Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Source Locator**
- **Source Identifier**
- **Destination Locator**
- **Destination Identifier**
Locators vs. Identifiers

- **Locator (L):**
  - uses the existing “Routing Prefix” bits of an IPv6 address.
  - names a single subnetwork (/48 allows subnetting).
  - **topologically significant, so the value of L changes as subnetwork connectivity changes.**
  - only used for routing and forwarding.

- **Identifier (I):**
  - Replaces the existing “Interface ID” bits of an IPv6 address
  - **Names a (physical/logical/virtual) host, not an interface.**
  - Remains constant even if connectivity/topology changes.
  - uses IEEE EUI-64 syntax, which is the same as IPv6:
  - only used by transport-layer [and above] protocols.
A Bit More Detail

• All ILNP nodes:
  ‣ have 1 or more Identifiers at a time.
  ‣ Identifiers are independent of the network interface
  ‣ only Identifiers are used at the Transport-Layer or above.

  ‣ have 1 or more Locators at a time.
  ‣ only Locators are used to route/forward packets.

• An ILNP “node” might be:
  ‣ a single physical machine,
  ‣ a virtual machine,
  ‣ or a distributed system.
# Naming Comparison

<table>
<thead>
<tr>
<th>Protocol Layer</th>
<th>IP</th>
<th>ILNP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>FQDN or IP address</td>
<td>FQDN</td>
</tr>
<tr>
<td>Transport</td>
<td>IP address (+ port number)</td>
<td>Identifier (+ port number)</td>
</tr>
<tr>
<td>Network</td>
<td>IP address</td>
<td>Locator</td>
</tr>
<tr>
<td>Link</td>
<td>MAC address</td>
<td>MAC address</td>
</tr>
</tbody>
</table>
ILNP:
Transport Layer Changes

• CRITICAL CHANGE:
  ▶ Transport-layer pseudo-header only includes IDENTIFIER, never the LOCATOR.

• IMPLICATIONS:
  ▶ We can multi-home nodes/sites without impacting routing.
  ▶ Mobility just became a built-in/native capability.
  ▶ Need a way to tell correspondents when we move.
  ▶ Historically, IETF concerned about authenticating location changes and providing equivalent security to current IPv6.
ILNP: DNS Enhancements

• New resource records (forward lookups)
  ▶ I: Identifier[s], unsigned 64-bit value, EUI-64 syntax
  ▶ L: Locator[s], unsigned 64-bit value, topological
  ▶ Each of these has a preference value, as with MX records.
  ▶ Nota Bene: DNS permits per-resource-record TTL values
    - Expect I values to be relatively longer-lived in all cases.
    - Expect L values to be relatively shorter-lived if mobile/multihomed.

• One performance optimisation
  ▶ LP: Locator Pointer; points to an L record
  ▶ Also has a preference value.

• Reverse lookups can work as they do today
## DNS Enhancements

<table>
<thead>
<tr>
<th>NAME</th>
<th>DNS Type</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifier</td>
<td>I</td>
<td>Names a Node</td>
</tr>
<tr>
<td>Locator</td>
<td>L</td>
<td>Names a subnetwork</td>
</tr>
<tr>
<td>Locator Pointer</td>
<td>LP</td>
<td>Forward pointer from FQDN to an L Record</td>
</tr>
</tbody>
</table>
Generating a Packet

- Source performs DNS lookup on destination’s FQDN.
- Source learns the set of I and L values for destination.
  - Like MX records, I and L records have preference values.
  - All valid I and L records are stored in local session cache
- Source selects the Source Locator and the Source ID to use for its own packet(s) to this destination.
- Source selects the Destination Locator and Destination ID to use.
- Source creates the packet and sends it out.
Mobility Approach
Naming and Mobility

• With MIP (v4 and v6), IP addresses retain their dual role, used for both location and identity:
  ▸ overloaded semantics creates complexity, since all IP addresses are (potentially) topologically significant.

• With ILNP, identity and location are separate:
  ▸ new Locator used as node moves:
    - reduces complexity: only Locator changes value.
  ▸ constant Identifier as node moves:
    - agents not needed and triangle routing never occurs.
  ▸ upper-layer state (e.g. TCP, UDP) only uses Identifier.
    - Recall that an Identifier names a node, not an interface.
Mobility has 2 Primary Aspects

- **1) Rendezvous**
  - How initially to find a node’s location to start a new session

- **2) Location Updates**
  - How to maintain existing communications sessions as one or more end nodes for that session change location

- ILNP uses DNS for initial rendezvous
- ILNP primarily uses control traffic for updates,
  - can fall back to DNS if this is ever necessary.
Mobility Implementation

• Implementation in correspondent node:
  ▶ uses DNS to find MN’s set of Identifiers and Locators.
  ▶ only uses Identifier(s) in transport-layer session state.
  ▶ uses Locator(s) only to forward/route packets.

• Implementation in mobile node (MN):
  ▶ accepts new sessions using currently valid I values.
  ▶ With ILNPv6, when the MN moves:
    - MN uses ICMP Locator Update (LU) to inform other nodes of the revised set of Locators for the MN.
    - LU can be authenticated via IP Security (or Nonce).
    - MN uses Secure Dynamic DNS Update (RFC-3007) to revise its Locator(s) in its Authoritative DNS server.
ILNPv6 Network Handoff

<table>
<thead>
<tr>
<th>MN</th>
<th>Mobile Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>Router serving MN</td>
</tr>
<tr>
<td>DNSR</td>
<td>DNS Server (reverse)</td>
</tr>
<tr>
<td>DNSH</td>
<td>DNS Server (forward)</td>
</tr>
<tr>
<td>CN</td>
<td>Correspondent Node</td>
</tr>
</tbody>
</table>
Multi-Homing
Multi-Homing with ILNP

- ILNP supports both site multi-homing & host multi-homing – and provides resilience/availability for both.
- ICMP Locator Update mechanism handles uplink changes (e.g. fibre cut/repair).
- ILNP reduces size of RIB in DFZ:
  - more-specific routing prefixes are no longer used for this.
- In turn, this greatly helps with BGP scalability.
- New optional DNS Locator Pointer (LP) record can enhance DNS scalability (e.g. for site multi-homing).
- Same approach also supports mobile networks.
Network Realms
(Scoped Addressing & “NAT”)
ILNPv6: “NAT” Integration

• IP Address Translation (NAT/NAPT) is here to stay:
  ‣ many residential IP gateways use NAT or NAPT.
  ‣ often-requested feature for IPv6 routers is NAT/NAPT.

• ILNPv6 reduces issues with these deployments:
  ‣ With ILNPv6, we have “Locator Translation”, instead.
  ‣ Identifiers don’t change when Locators are translated.
  ‣ Upper-layer protocol state is bound to I only, never to L.
  ‣ Translation is now invisible to upper-layer protocols.

• ILNPv6 IPsec is not affected by NAT:
  ‣ Security Association is bound to Identifiers, not Locators.
  ‣ ILNP AH covers Identifiers, but does not cover Locators.
  ‣ ILNP IPsec and “NAT” work fine together \(\text{w/o extra code}\)
Security Considerations
Security Mechanisms

• IP Security with ILNP:
  › can use IPsec AH and ESP for cryptographic protection
  › ILNP AH includes I values, but excludes L values
  › IPsec Security Association (SA) bound to value of I, not L

• New IPv6 Destination Option - Nonce:
  › contains clear-text 64-bit unpredictable nonce value
  › protects against off-path attacks on a session [child proof]
    - existing IP without IPsec is vulnerable to on-path attacks
    - So Nonce use is affordable, yet provides equivalent protection as today
  › primarily used to authenticate control traffic:
    - e.g. ICMP Locator Update (LU) message

• Existing IETF DNS Security can be used as-is
Operational Considerations
Incremental Deployment

- ILNPv6 is a set of extensions to IPv6
- No changes to:
  - IPv6 routing protocols,
  - IPv6 forwarding (no silicon or software changes),
  - IPv6 Neighbour Discovery (ND)
- Implications:
  - Existing IPv6 networks already support ILNPv6 packets.
  - No upgrades needed to routers.
- ILNPv6 enhances host TCP/IPv6 stacks
  - Host OSs will need to be upgraded over time.
Backward Compatibility

• How does an initiating node know whether the remote node is ILNPv6 enabled or not?
  ▶ ILNPv6 DNS records (I, L) will be returned on DNS lookup, in addition to usual IPv6 (or IPv4) DNS records.

• How does a responding node know whether the remote node is ILNPv6 enabled or not?
  ▶ ILNPv6 Nonce is present in received packet from remote node that is initiating a new UDP/TCP/SCTP session.

• If either node doesn’t support ILNPv6, the other node falls back to using existing ordinary IPv6.

• No loss of connectivity/reachability during evolution.
ILNPv6: No Free Lunch

- No globally-routable network interface name:
  - potential impact on SNMP MIBs, e.g. to get interface counters from a particular interface.
- A few legacy apps might remain problematic, not sure yet.
  - Probably should test with FTP
- DNS reliance is not new, but is more explicit:
  - at present, users perceive “DNS fault” as “network down”.
  - ILNP creates no new DNS security issues.
  - Existing IETF DNS standards work fine without alteration.
Research Status
Next steps

• Demo implementation of ILNPv6 in BSD UNIX
  ▸ which is in progress now.

• Plan to use the demo implementation in experiments to test feasibility of ILNPv6:
  ▸ verify compatibility with IPv6 routers.
  ▸ wide area testing on UK SuperJANET connectivity
    - initially between St Andrews (Scotland) and London (England).
  ▸ later extend to international testing over IPv6 backbone.

• Fine-tune ILNP design and implementation based on experimental results.

• Would like to examine ILNP for MANET deployments
Summary

- ILNP treats the IP Address as consisting of separate Identifier & Locator values.
- This enables native Mobility (without agents).
- Also, Multi-Homing, NAT, and Security are well integrated with Mobility.
- Improvements in the Naming Architecture enable simpler protocol approaches and ILNP is consistent with the wider goals of the future direction of the Internet architecture.
Thank you!

- Three very drafty Internet-Drafts are online:
  - “ILNP Concept of Operations”, draft-rja-ilnp-intro-01.txt
  - “Nonce Destination Option”, draft-rja-ilnp-nonce-00.txt
  - “Additional DNS Records”, draft-rja-ilnp-dns-00.txt

- For more, please contact:
  - Ran Atkinson rja@extremenetworks.com