ILNP: a whirlwind tour

Saleem Bhatti, University of St Andrews, UK

Outline

1. **What?**
   - **Basic information about ILNP.**
2. Why?
   - The rationale for ILNP.
3. How?
   - Basic operation of ILNP.
4. When?
   - ILNP development.

What is ILNP?

- Identifier Locator Network Protocol:
  - http://ilnp.cs.st-andrews.ac.uk/
- ILNP enhances Internet Protocol functionality through the use of crisp **naming**.
- March 2010: IRTF RRG Chairs recommend ILNP for development within the IETF: http://www.ietf.org/mail-archive/web/rrg/current/msg06356.html
- People:
  - Ran Atkinson (Cheltenham Research, US)
  - Saleem Bhatti (University of St Andrews, UK)

Identifier / Locator Network Protocol

- This is a work in progress:
  - http://ilnp.cs.st-andrews.ac.uk/
- Focus on network and transport layers (for now)
- This talk – ILNPv6 as a parallel/concurrent system on the **existing** Internet infrastructure:
  - **We take a bottom-up engineering approach.**
  - Initial idea based on Mike O'Dell's 8+8/GSE (1996/7)
    - Many enhancements compared to 8+8/GSE
    - Initial “IPv6 8+8” idea dates from emails posted by Bob Smart (02 Jun 1994) and Dave Clark (11 Jan 1995): http://www.ietf.org/mail-archive/web/rrg/current/msg02455.html

Outline

1. What?
   - Basic information about ILNP.
2. **Why?**
   - **The rationale for ILNP.**
3. How?
   - Basic operation of ILNP.
4. When?
   - ILNP development.

(New) Requirements

- We wish to try and support a ***harmonised solution to many*** **network functions**:
  - Multi-homing (host and site).
  - Mobility (host and network).
  - Multi-path capable transport protocols.
  - Localised addressing (NAT).
  - Traffic engineering capability.
  - Packet-level, end-to-end security.
- Currently, solutions for these functions remain disparate and do not function well together.

Engineering issues for ILNPv6

We wish to have an **incrementally deployable** solution that is also **backwards compatible**:

1. Core network devices and protocols should not need to change, e.g. routers, switches of today can be used without modification.
2. Reuse the existing core protocol deployment as much as possible, e.g. make use of existing IPv6.
3. Try to limit the impact on current applications (but we have to accept some applications might break).
4. The end system stack will need to change, but changes should run in parallel with current stack.

Names

- My definition of a “name”:
  A set of bits used to label an object. The semantics of the name are defined within the context of use of the object it names.
- Examples:
  - protocol name – ‘http’
  - port number – ‘80’
  - fully qualified domain name (FQDN), e.g. ‘marston.cs.st-andrews.ac.uk’
  - **IP address - ‘138.251.195.61’**

Application layer protocols

- URLs:
  https://marston.cs.st-andrews.ac.uk/
- Can also use an IP address:
  https://138.251.195.61/
- Notice, the use of **either** a DNS name or an IP address – FQDN and IP address used as synonyms.
- **IP address is overloaded**:
  - used in application protocols as a **session identifier**

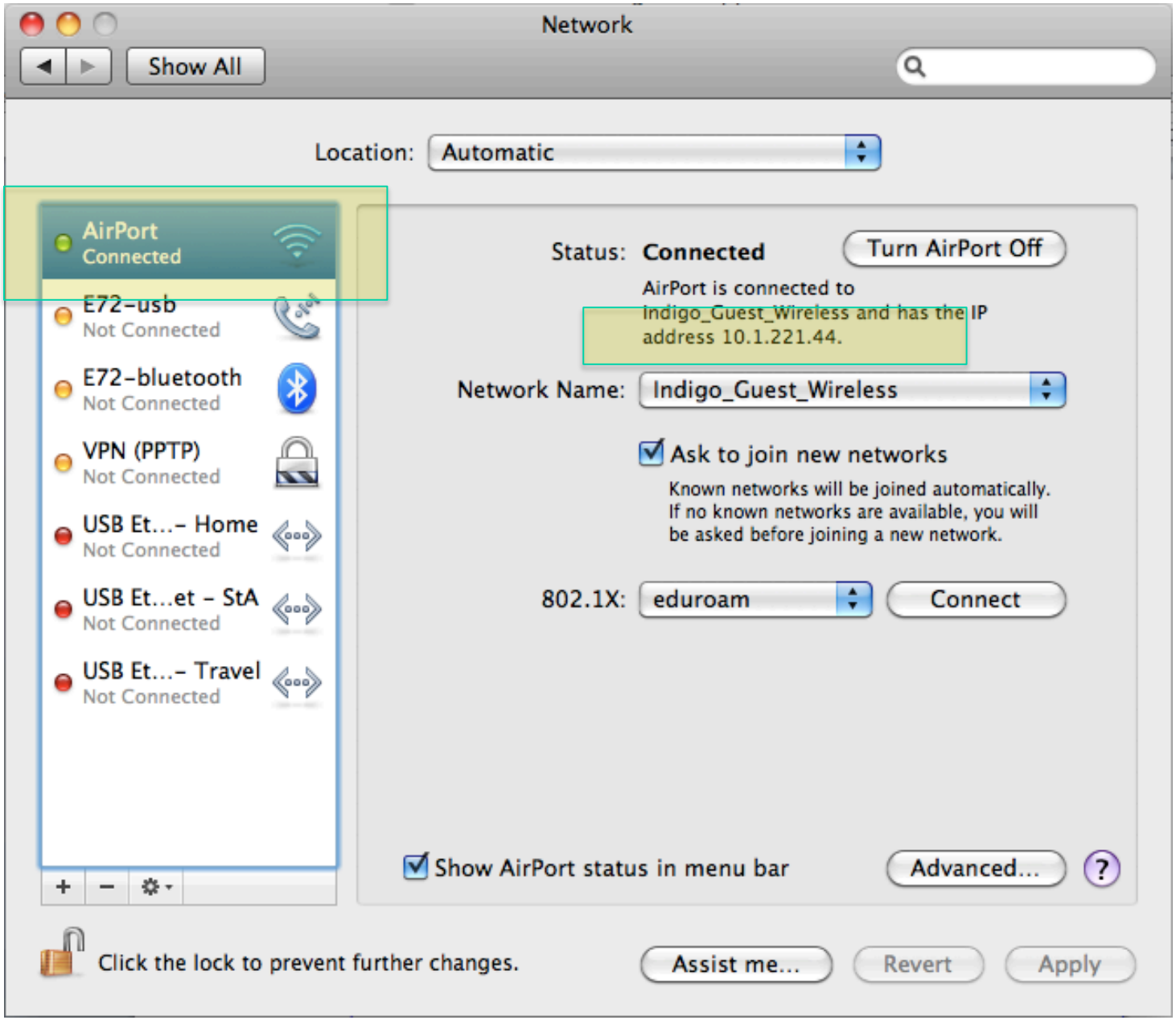
Transport protocols

- TCP uses a tuple to identify a TCP connection:
  - local IP address
  - local port number
  - remote IP address
  - remote port number
- TCP state (and the pseudo-header checksum) is bound to **all** the bits in the local and remote IP address.
- **IP address used as an Identifier.**

Network layer

- IP address bits are used in **routing**:
  - IP address (network) prefix, e.g.
    138.251.195.61/24
    means that 138.251.61 (the network prefix) is used for routing at the IP layer
- The host part of the address may be further used for sub-netting at the site:
  - IP sub-netting on host bits, e.g.
    138.251.195.61/25
    means 1 bit of the host part of the address is used
- **IP Address used as a Locator.**

Interface identifier



RFC4984 (Sep 2007)

IAB Naming and Addressing Workshop 18-19 October 2006
RFC4984, p6

.... workshop participants concluded that the so-called "locator/identifier overload" of the IP address semantics is one of the causes of the routing scalability problem as we see today. Thus, a "split" seems necessary to scale the routing system, although how to actually architect and implement such a split was not explored in detail.

RFC2101 (Feb 1997)

IPv4 Address Behaviour Today
RFC2101 pp 3-4

Identifiers should be assigned at birth, never change, and never be re-used. Locators should describe the host's position in the network's topology, and should change whenever the topology changes. Unfortunately neither of the these ideals are met by IPv4 addresses.

IEN 1 (29 July 1977)

- Section 3 ADDRESSING (pp 6-12):
  - Discusses physical vs. logical addressing
- Section 3.2 Special Topologies (pp 7-8):
  - Specifically discusses “Changes in Topology” (mobility) and “Multiply-Connected Hosts” (multi-homing)
  - Flags problems with use of IP addresses (as today).
- Lots of wisdom:
  - IENs 19, 23, 31, 46

Layers are entangled

Protocol Layer	IP
Application	FQDN or IP address
Transport	IP address (+ port number)
Network	IP address
(Interface)	IP address

Entanglement ☹

A problem for harmonising the new requirements …

Outline

1. What?
   - Basic information about ILNP.
2. Why?
   - The rationale for ILNP.
3. **How?**
   - **Basic operation of ILNP.**
4. When?
   - ILNP development.

Naming: IP vs. ILNP

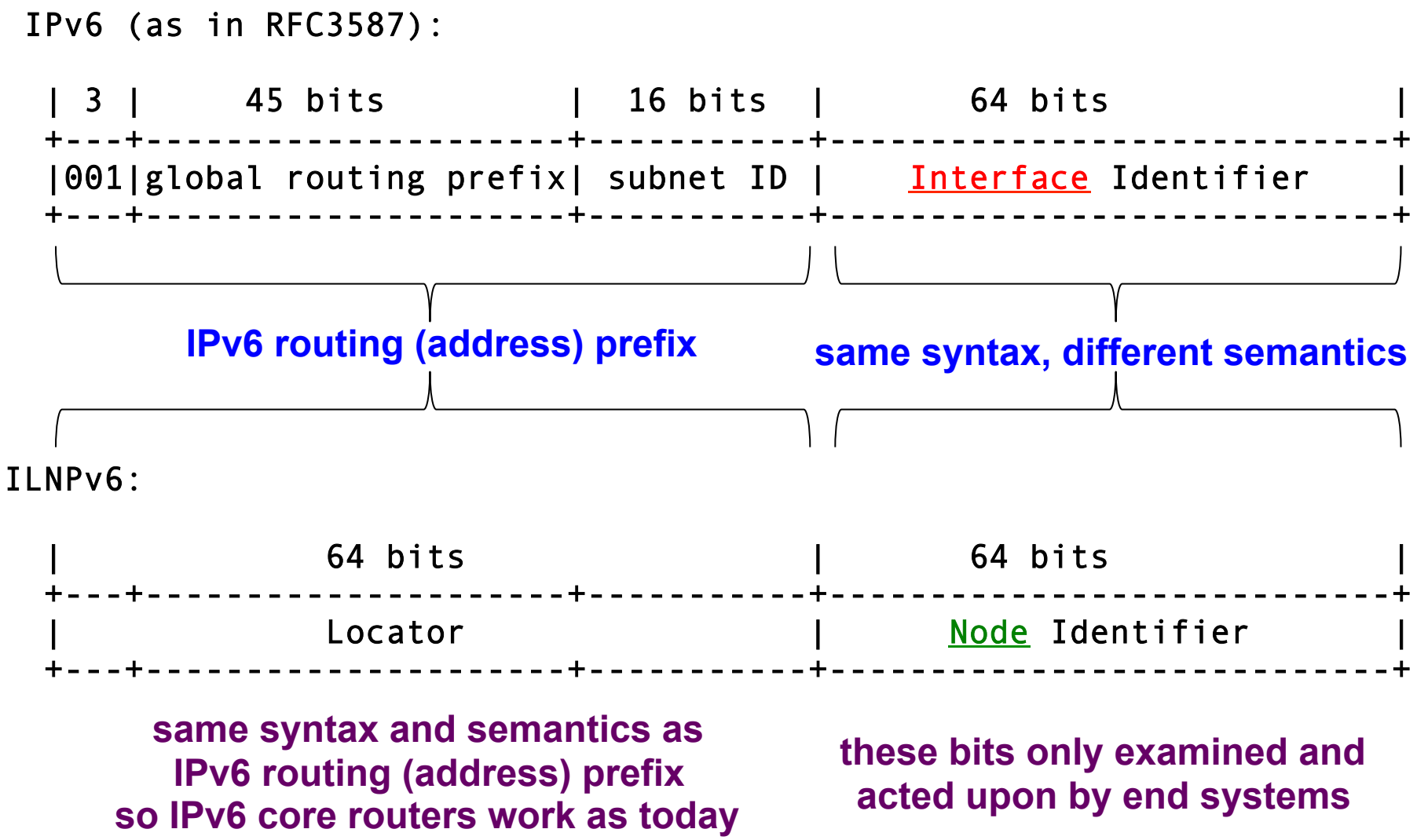
Protocol Layer	IP	ILNP
Application	FQDN or IP address	FQDN (RFC1958)
Transport	IP address (+ port number)	Identifier (+ port number)
Network	IP address	Locator
(Interface)	IP address	(dynamic mapping)
	Entanglement ☹	**Separation ☺**

FQDN = fully qualified domain name

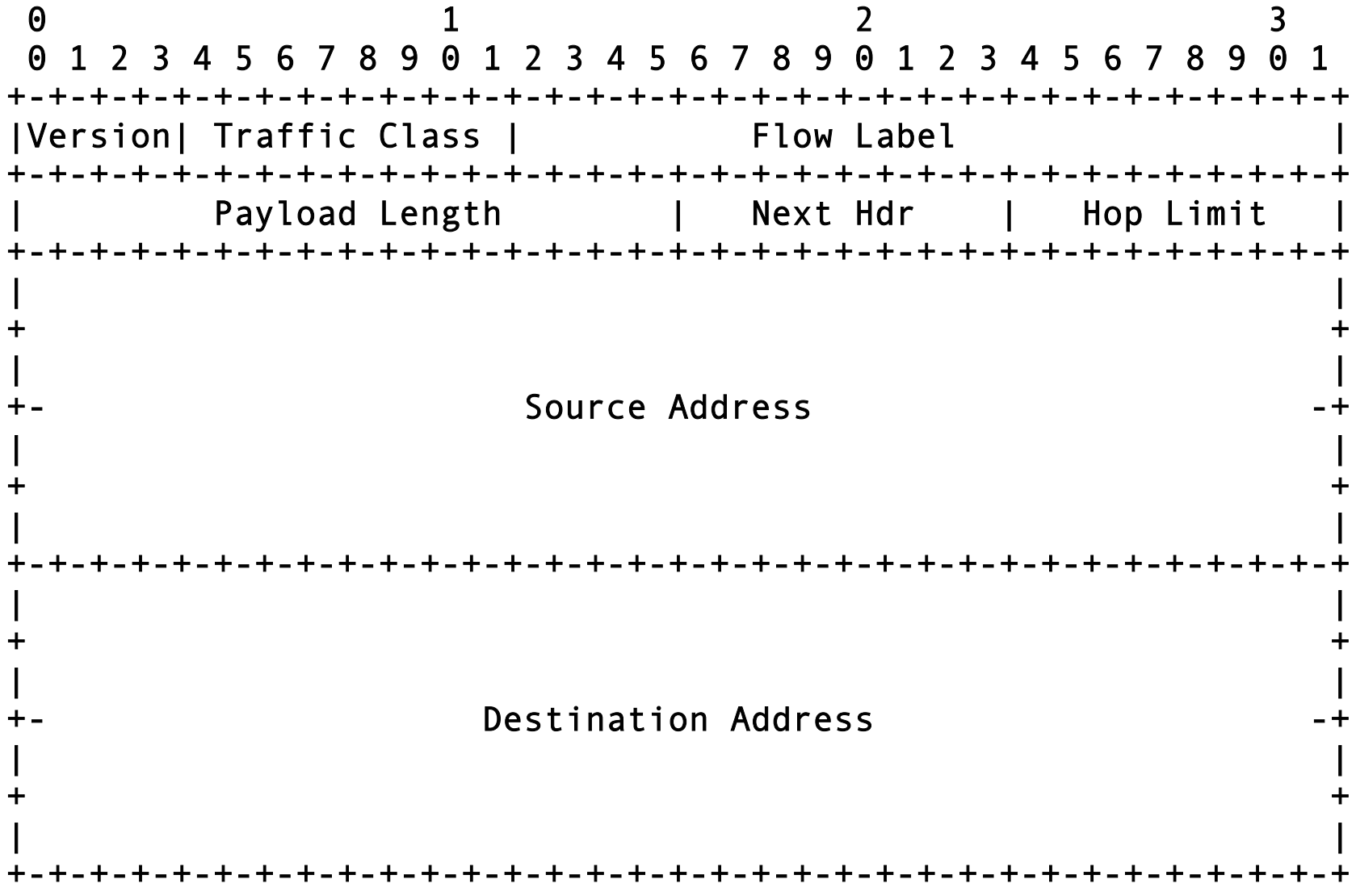
ILNPv6

- Can be seen as a set of 'extensions' to IPv6:
  - Uses same packet format as IPv6 in network core.
  - IPv6 core routers do not need to change.
  - Incrementally deployable on IPv6 core.
  - Backwards compatible with IPv6.
- Split 128-bit IPv6 address:
  - **64-bit Locator (L) - network name.**
  - **64-bit Identifier (I) - node name.**
- Could also be retro-fitted to IPv4 (but messy).

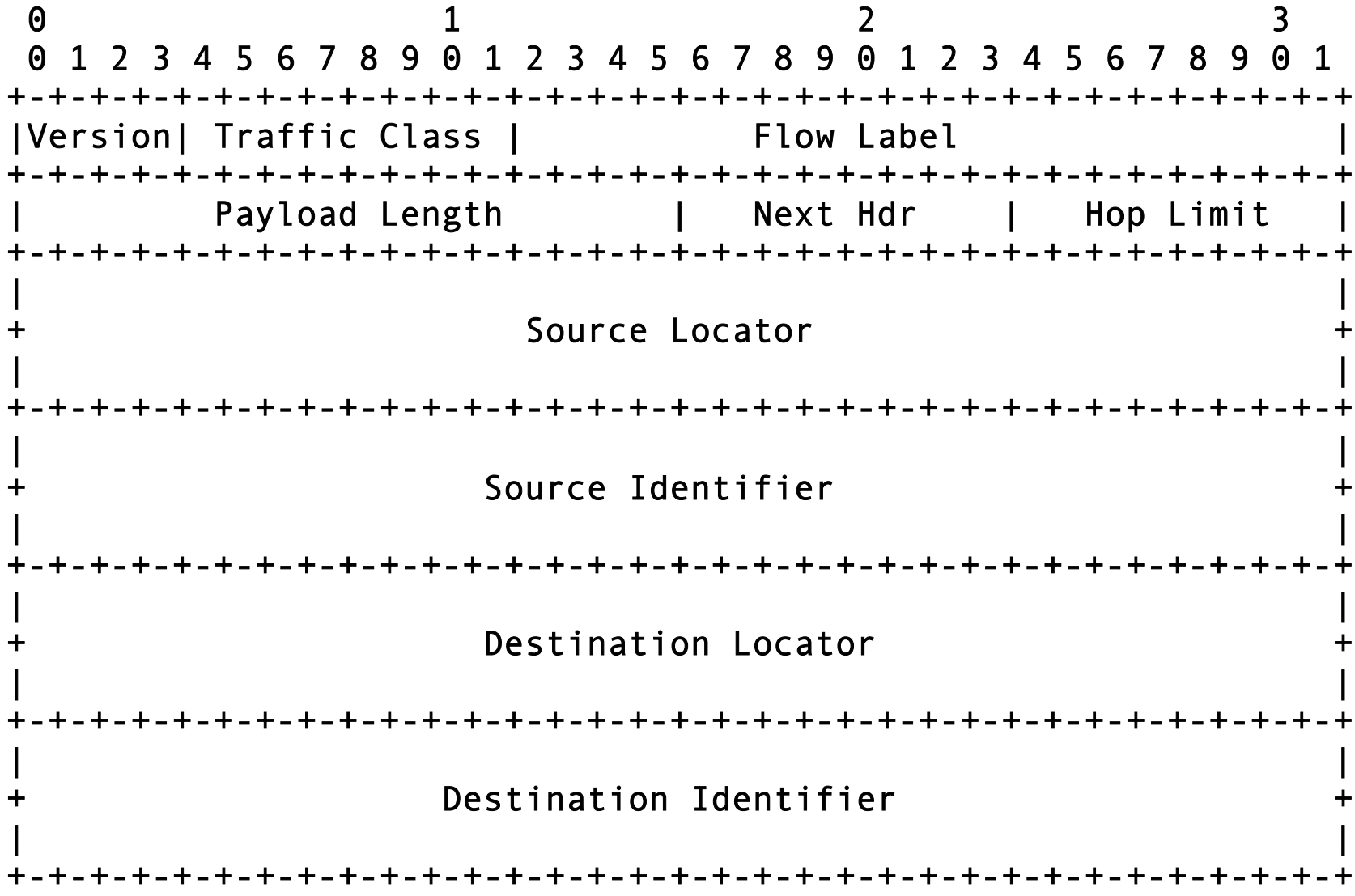
IPv6 addresses and ILNPv6



IPv6 packet header



ILNPv6 packet header



Locators and Identifiers [1]

- **Locator, L:**
  - **Topologically significant.**
  - Names a (sub)network (as today's **network prefix**).
  - Used only for routing and forwarding in the core.
- **Identifier, I:**
  - **Is not topologically significant.**
  - Names a logical/virtual/physical node, does **not** name an interface (value ala RFC4291 Sec 2.5.1).
- **Upper layer protocols bind only to Identifier.**

Locators and Identifiers [2]

- Locator, L:
  - **Can change** value during the lifetime of a transport session (mobility, site-controlled traffic engineering).
  - Multiple Locators can be used simultaneously (multi-homing, multi-path transport protocols).
- Identifier, I:
  - **Remains constant** during the lifetime of a transport session (localised addressing, IPsec.).
  - Multiple Identifiers can be used simultaneously by a node, but not for the same session.

DNS enhancements required

Name	DNS Type	Definition
Identifier	ID	Names a Node
Locator	L64	Names a subnet
Reverse Locator	PTRL	FQDN for the DNS Server responsible for subnet L
Reverse Identifier	PTRI	FQDN for the I that is present at subnet L
Locator Pointer	LP	Forward pointer from FQDN to an L record

FQDN = fully qualified domain name

Outline

1. What?
   - Basic information about ILNP.
2. Why?
   - The rationale for ILNP.
3. How?
   - Basic operation of ILNP.
4. **When?**
   - **ILNP development.**

Development options

- Simulation:
  - Good control, high-scalability, reproducibility of experiments etc.
- Emulation:
  - e.g. use of an overlay network is feasible (Masters student project, 2009), with constraints.
  - OneLab, PlanetLab (control + mgmt + monitoring?)
- **Test-bed – implementation in OS stack:**
  - **prototype Linux (~Q3 2011?)**
  - **prototype FreeBSD (~Q4 2011?)**

No free lunch

- DNS support – not new, but explicit in ILNPv6:
  - New RRs + **zero TTL for some DNS records**.
  - Secure DNS Dynamic Update for Locator changes.
- Renumbering + address management at sites.
- No globally routeable interface name, which may impact some applications such as SNMP.
- Some legacy applications may break, e.g. FTP.
- Interworking scenarios (IPv6, IPv4).

Comparison with LISP [1]

- **LISP**: customer focused, practically-directed engineering solution, with a goal of minimal cost to end sites, employing network upgrades that would be invisible to the end users, and reduce the burden of routing state on "core network". Objective is to provide a product-based solution.
- **ILNPv6**: research vehicle to explore the current use of addressing and examine fundamental architectural issues of how naming can be used to enable new functionality. Objective is to give a proof-of-concept implementation in order to demonstrate that ILNP could be made to work as described.

Comparison with LISP [2]

	LISP	ILNPv6
What changes?	network	host
Architecture	map-and-encap	naming
Site renumbering	no	optional
End-host changes	no	yes
New network entities required	yes	no
Backbone MTU > access MTU	yes	no
BGP & DFZ state reduction	yes	yes
State 'displacement'	EID-RLOC mapping	DNS lookups
Working code	yes	in progress
'Well-behaved' applications work without modification	yes	yes
IPv6	yes	yes
IPv4	yes	possibly[1]

[1] Technically possible, deployability unclear.

Comparison with LISP [3]

	LISP	ILNPv6
Site multi-homing	yes	yes
Host multi-homing	not currently defined[1]	yes
Multicast	yes	yes
Traffic engineering options	yes	yes
Localised addressing (NAT)	in progress[2]	yes
Harmonised functionality	in progress[2]	yes
Mobile hosts	in progress[2]	yes
Mobile networks	not currently defined[1]	yes
Multi-path transport	no	yes

[1] Technically possible, deployability unclear.
[2] Internet draft document available.

Thank You!

- More information on ILNP:
  - http://ilnp.cs.st-andrews.ac.uk/
- Contact information:
  - Saleem Bhatti <saleem@cs.st-andrews.ac.uk>