# IPv6

gianluca.verin@libero.it

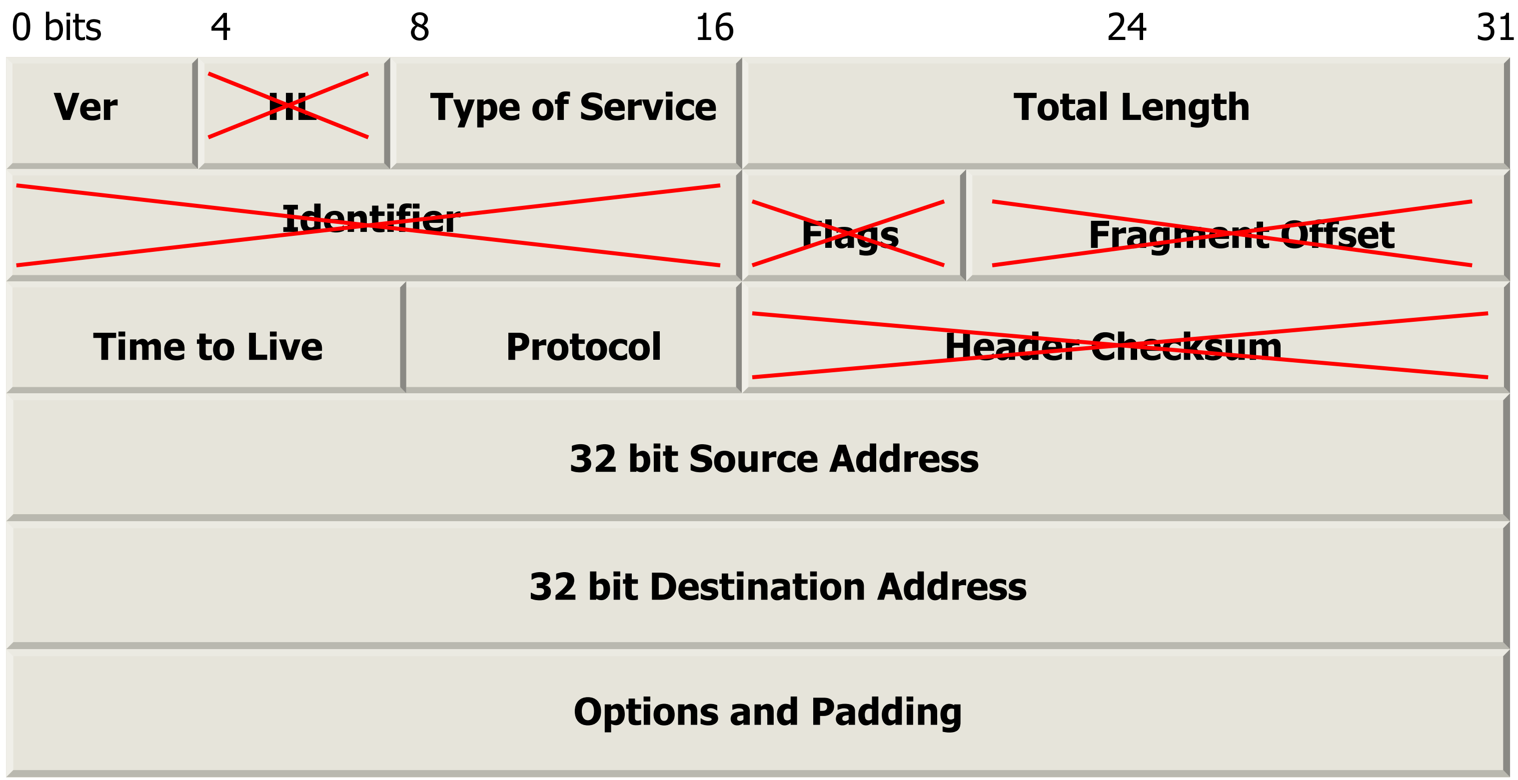
# Agenda

- IPv6 Basics
- Connecting to 6Bone
- Why do we need IPv6?
- IPv6 Introduction-Transition
- IPv6 and open source community
- Future applications
- Summary

# Introduction to IPv6

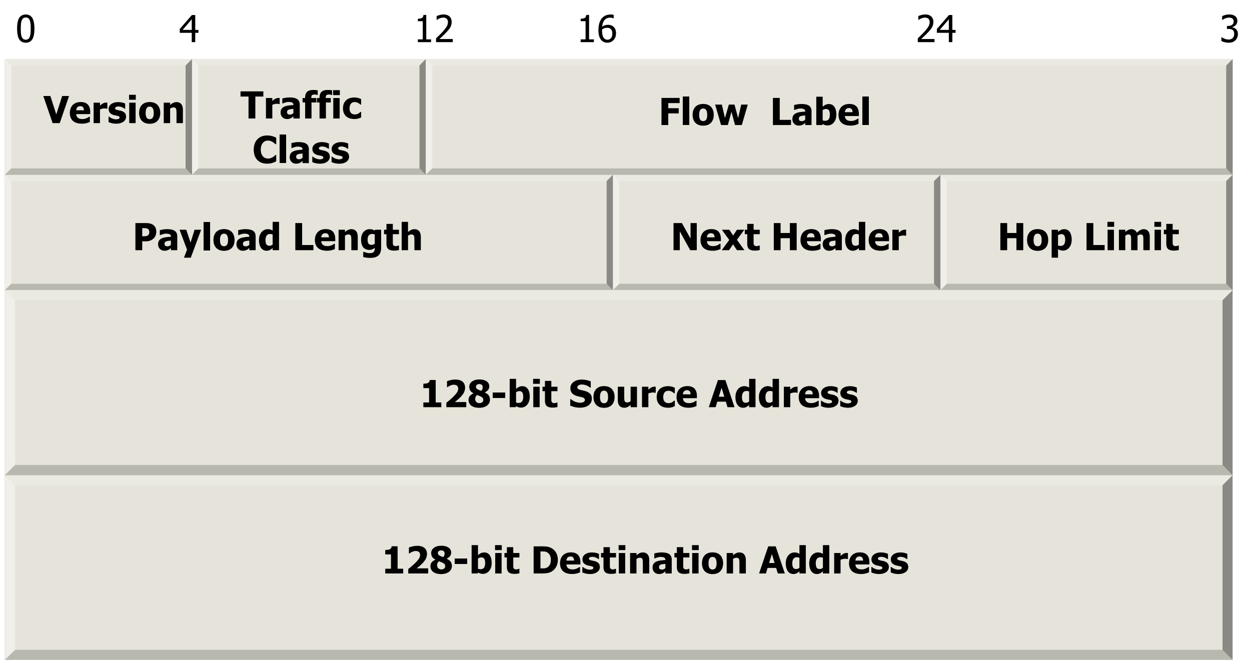
- IP Standards Body - IETF (Internet Engineering Task Force)
- IETF recognised the need to work on an improved Internet Protocol
- IP Next Generation (IPng) Directorate formed in 1991
- More than 10 years of work on IPng resulted in IPv6
- IPv6 is now mature – all new IP protocols developed in IETF are now expected to support IPv6
- IETF/IAB has recommended the use of IPv6 for wireless (billions of devices)
- IPv6 adopted by 3GPP for IP Multimedia services
- Government Actions
  - Japan (incentives for IPv6) and China (NGN network)
  - U.S. DoD (Network-centric warfare)
  - EU IPv6 Task Force (IPv6 is part of the e-Europe Broadband Action)

# The IPv4 Header (>= 20 bytes)




- Variable header length (e.g. for options) and fragmenting makes it harder to do fast processing in software
- Checksum redundant since error checks always done at layer-2

# The IPv6 Header (40 bytes)




| | | | |
|---|---|---|---|
| **Version** | 6 | **Payload Length** | Length of IPv6 payload |
| **Traffic Class** | Priority of IPv6 packets | **Next Header** | Type of header following the IPv6 header |
| **Flow Label** | Special handling of packet | **Hop Limit** | Decremented by 1 in each router |

# Extension Headers

| | |
|---|---|
| **IPv6 Header**<br>Next Header = TCP | **TCP Header + Data** |

| | | |
|---|---|---|
| **IPv6 Header**<br>Next Header = Hop-by-Hop | **Hop-by-Hop Header**<br>Next Header = UDP | **UDP Header + Data** |

| | | | | |
|---|---|---|---|---|
| **IPv6 Header**<br>Next Header = Destination | **Destination Header**<br>Next Header = Routing | **Routing Header**<br>Next Header = ESP | **ESP Header**<br>Next Header = TCP | **TCP Header + Data** |

# IPv6 Addresses

- There are three different types of addresses:

  Unicast - An identifier for a single interface (e.g. Aggregatable Global Unicast)

  Anycast - An identifier for a set of interfaces. A packet sent to an Anycast address is ***delivered to one of the interfaces*** identified by that address. Anycast addresses could be used to reach the nearest node for a certain service.

  Multicast - An identifier for a set of interfaces. A packet sent to a multicast address is ***delivered to all interfaces*** identified by that address.

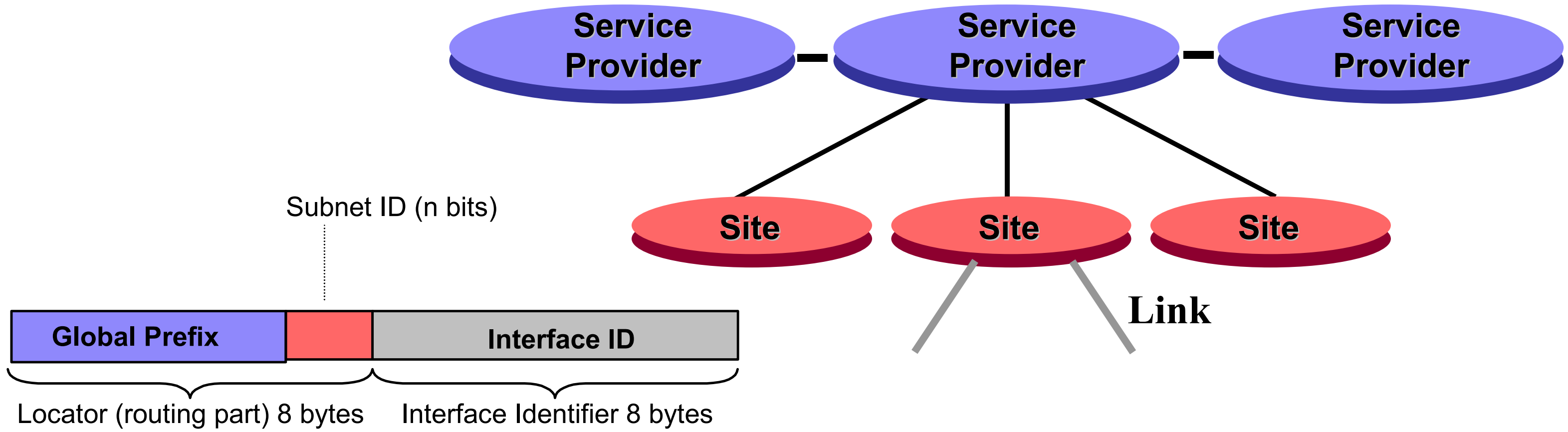
# IPv6 Unicast Address scopes

- **Link-local**
  Only unique/valid on same link (ALL DEVICES NEED THIS)
  Autconfigured at startup. Packets are not forwarded beyond the link when the source or destination is link-local

- ~~Site-local (**DEPRECATED**)
  Only unique/valid within the same site (operator-defined)
  May be autoconfigured. Packets are not forwarded beyond the “site” when the source or destination is site-local~~

- **Global**
  Globally unique. May be autoconfigured.

# IPv6 Aggregatable Global Unicast Addressing




- Geographical significance of IPv6 unicast addressing allows efficient route aggregation
- Example IPv6 assignment: /32 => 32-bit subnet id
- Stateless example: Each /32 can be broken up into > 4.2 Billion /64s distributed to links/hosts

# IPv6 Addresses

- ::   unspecified address
- ::1 loopback address
- fe80::<64bits interface id> Link local address
- 2001:923f:200:101::/64. Globally routable IPv6 prefix assigned to hosts connected to a router's interface
- 2001:923f:200::/48. Typical prefix assigned to networks. Space for 2^16 = 65K /64 prefix

Special format for globally routable address

- 2002:v4_addr::/48 6-to-4 prefix
- </64 prefix>:0000:5efe:v4_addr ISATAP address

# Applying for IPv6 Addresses

- Operator should apply for a /32 IPv6 Global prefix to its local Regional or National Internet Registry (RIR/NIR): RIPE for Europe RIPE (www.ripe.net)
- ftp://ftp.ripe.net/ripe/docs/ripe-267.txt
- Else follow step by step instruction in SixXS in www.sixxs.net/main/.
- SixXS offers tunnel broker service and upon request you usually get a /48 prefix.

# Connecting to 6Bone

- If your ISP does not offer native IPv6 connectivity you may register with 6Bone and use a tunnel broker such as SixXS. From the tunnel broker you get: IPv4 address of tunnel broker end point, IPv6 address ISP end point, your IPv6 address. Optionally a /48 routable prefix
- FreeBSD configuration example:

# ifconfig gif0 create
# ifconfig gif0 tunnel [Your IPv4 Endpoint] [POP IPv4 Endpoint]
# ifconfig gif0 inet6 [Your IPv6 Endpoint] [POP IPv6 Endpoint] prefixlen 128

Route your IPv6 traffic via the SixXS POP:
# route add -inet6 default [POP IPv6 Endpoint]

If you plan to use your machine as a router:
# sysctl -w net.inet6.ip6.forwarding=1

Send RA to interface (prefix contained in file /etc/rtadvd.conf )

#rtadv [interface name]

# What do IPv6 addresses look like?

Tunnel Interface

IPv4 pop address

- ifconfig gif0
- gif0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1280
- tunnel inet 192.168.1.3 --> 212.224.0.188
- inet6 2001:6f8:900:43f::2 --> 2001:6f8:900:43f::1 prefixlen 128
- inet6 fe80::208:74ff:fee9:baa8%gif0 prefixlen 64 scopeid 0x5

Global Unicast

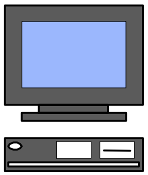
Link-Local Unicast

# IPv6 Address Configuration

- Stateless (server-less)
  - Does not depend on communication with a server (reliability increased)
  - IPv6 hosts automatically configure addresses without the need for user intervention (Concept is described in RFC2462)
  - For those services where user authentication & accounting is needed, RADIUS server may be used also to allocate prefixes
- Stateful
  - Automatic address configuration given to hosts by DHCPv6 server
  - Requries a DHCPv6 server in the network
  - DHCPv6 client in host, first-hop router as DHCPv6 relay

# IPv6 Neighbor Discovery (Stateless Addressing)

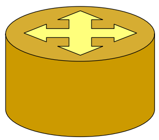
1. Host forms Link-local address (combining link-local prefix and interface Id)
   PPP, Ethernet



**IPv6 Host**

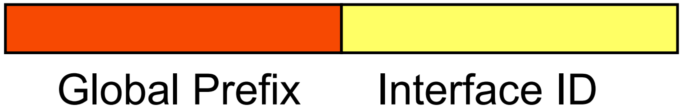
2. Host optionally sends Router Solicitation (link-local multicast)

3. Router sends periodic multicast Router Advertisement or unicast in response to solicitation



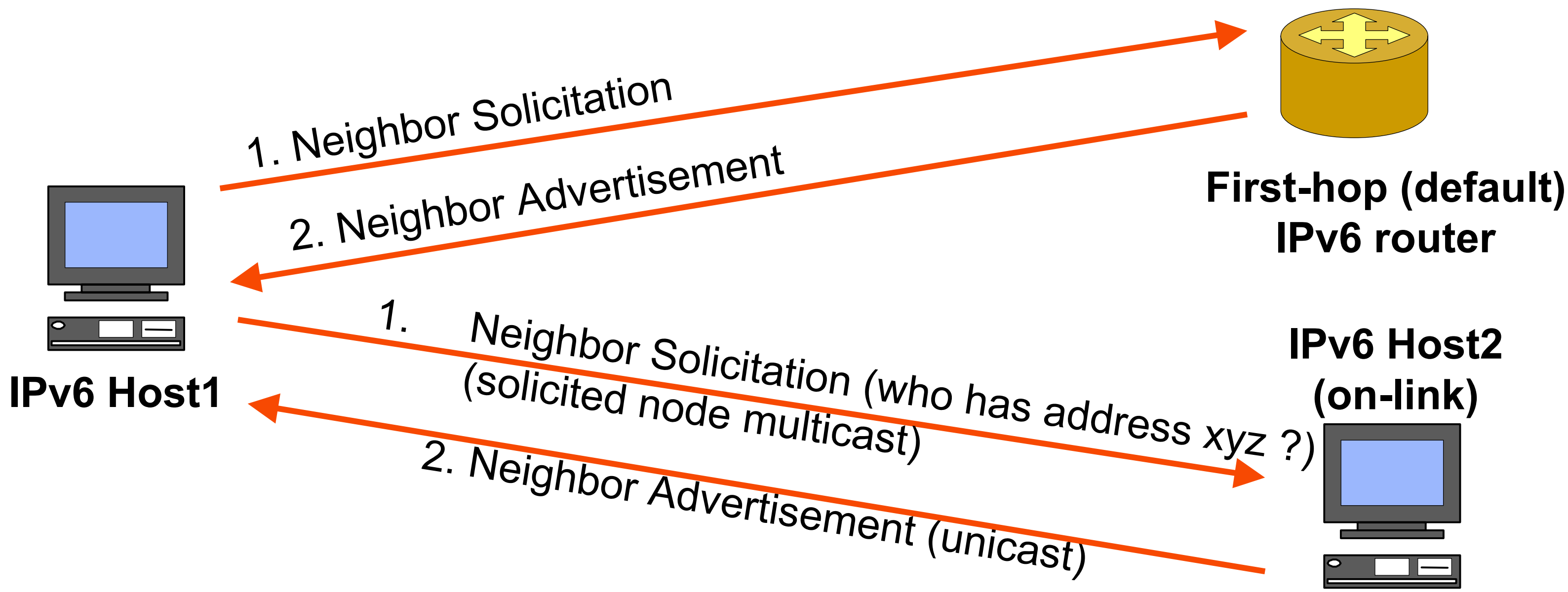
**First-hop (default) IPv6 router**

4. Host forms Global address (128 bits) by combining



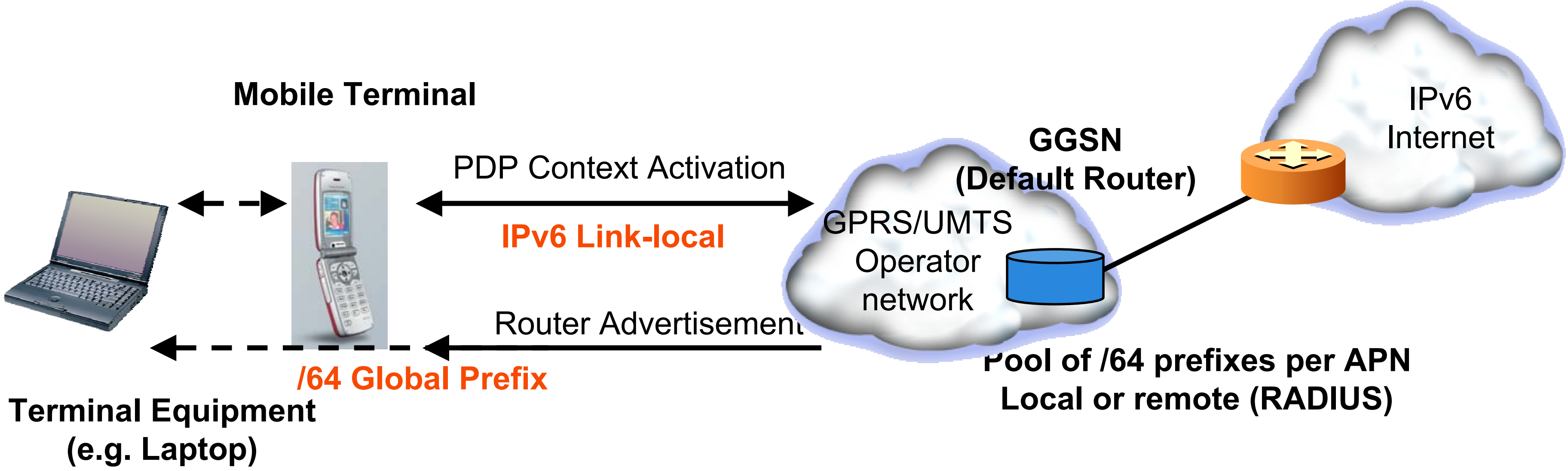

- Neighbor Discovery (ND) uses ICMPv6
- Router Advertisement (RA) contains Global Prefix, Lifetimes
- Interface Identified (IID) may be based on MAC address or random number

# IPv6 Neighbor Discovery (continued)




- ND replaces ARP (IPv4) for link-layer address resolution of nodes on the same IPv6 link – link-local addresses are used for these messages
- Neighbor Unreachability Detection (NUD) to determine when a neighbor is no longer reachable (e.g. default router)

# How are IPv6 addresses assigned to Mobiles?

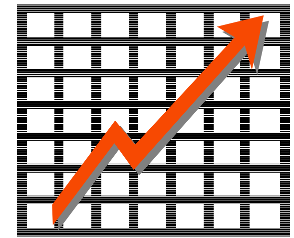



- IETF/3GPP successful collaboration produced future-proof IPv6 standard for Mobile Networks
- Mobile host's Link-local address provided by GGSN to avoid duplication
- Each Mobile Terminal is assigned a unique /64 IPv6 prefix which can be used to create multiple addresses (privacy), Personal Area Networks etc.
- Unreliable DAD may be avoided (reduces messages over air)

# A few IPv6 Specifications

- IPv6 Protocol Specification (RFC2460)
- IPv6 Neighbor Discovery (RFC2461) – being updated
- IPv6 Stateless Address Autoconfiguration (RFC2462) – being updated
- IPv6 Internet Control Message Protocol (RFC2463) – being updated
- IPv6 Addressing Architecture (RFC3513)
- IPv6 over Ethernet (RFC2464)
- Support for IPv6 in Session Description Protocol (SDP) (RFC 3266)
- Mobile IPv6 (RFC 3775)
- Dynamic Host Configuration Protocol for IPv6 (RFC 3315)
- Recommendations for IPv6 in 3GPP Standards (RFC 3314)
- Internet Protocol Version 6 (IPv6) for Some Second and Third Generation Cellular Hosts (RFC 3316)
- Transition Scenarios for 3GPP Networks (RFC 3574)

# Market Trends & Demands



- World Mobile Subscriptions forecast to pass the 2 Billion mark in 2007*
- Restrictive IPv4 Assignment Policies
- IPv4 Unassigned addresses expected to run out in 2018
- Not enough IPv4 public addresses for mobile users
  Need a long-term solution for continued market growth
- Traditionally when more circuit switched “phone numbers” are needed this is solved by modifying/adding area codes
  - We don’t share phone numbers, why share IP addresses?
- Increased network security: working e2e security model
- Increased Quality of Service demands

* Ovum Sept 2004

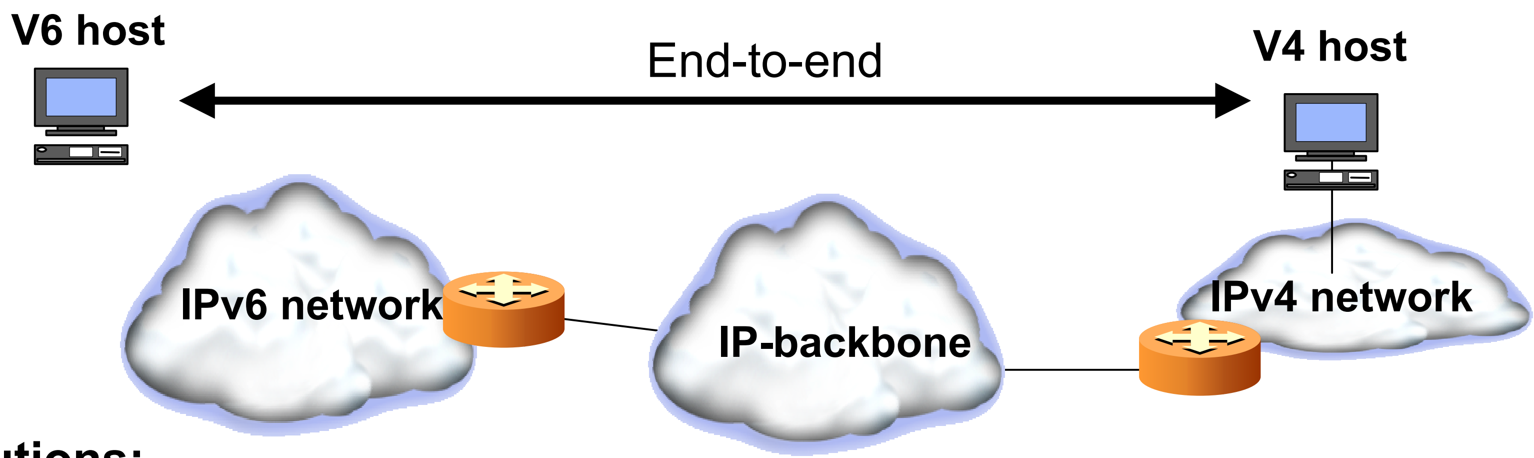
# Why can’t we just use IPv4 NATs?

- NATs translate IP addresses (including IP addresses used inside applications when the NAT is combined with an ALG)
- Security is problematic with NATs
- NATs need to process all packets => can add delay to delay-sensitive sensitive packets (e.g. voice, video)
- NATs keep per-connection state => scalability is an issue
- What if a NAT fails?
  => Reliability
- Need to support ALL applications which use IP addresses at the application level (FTP, Netmeeting etc.)
  => What happens to new or proprietary/secret applications?
- NATs do not provide security on their own – need a Firewall
- IPv6 allows you to do away with these problems

# IPv6 – Other advantages

- Built-in security support (IPsec)
- Efficient routing
- Features for simplified Network operations
  - Address autoconfiguration
  - Automated network (router) renumbering
  - Automated Server discovery (e.g. DNS)
- Same level of Quality of Service (QoS) support as in IPv4 with potential improvement using Flow Label
- Built in mobility support (MIPv6)

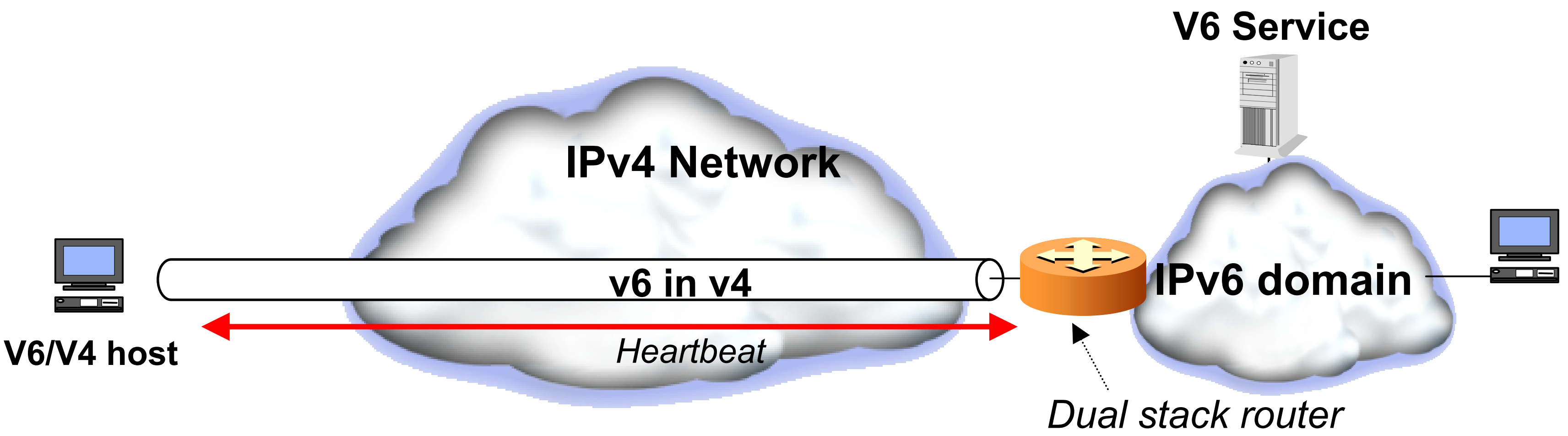
# End-to-end incompatibility
## IPv6 host communicating with an IPv4 peer




**Solutions:**

- Dual stack networks (but then every network on the left should have both an IPv6 and IPv4 address, not always possible if the host is a mobile)
- Translators:
  - Stateful Translators (NA(P)T-PT), with ALGs
  - Stateless Translators (SIIT) – not recommended
- Dual-stack Application Proxies (e.g. HTTP, FTP, E-mail, WAP)

# Automatic tunnelling, using heartbit




- V6/V4 host change IPv4 address, need to communicate host IPv4 address.
- Procedure password protected
- Heartbeat every 60 seconds, after 300 seconds tunnel tear down

# IPv6 Support in OS

- Virtually every OS that runs on PC and server supports IPv6.
  - Linux
  - FreeBSD, NetBSD, OpenBSD
  - 

  But also
  - Solaris
  - HP-UX
  - Windows

# Open Source IPv6 Applications

- Many of today open source applications are dual stacks. Just to list a few:
  - Apache www server
  - Firefox www client
  - Ircd irc server
  - Xchat2 irc client
  - ftp, telnet, openssh
  - ….

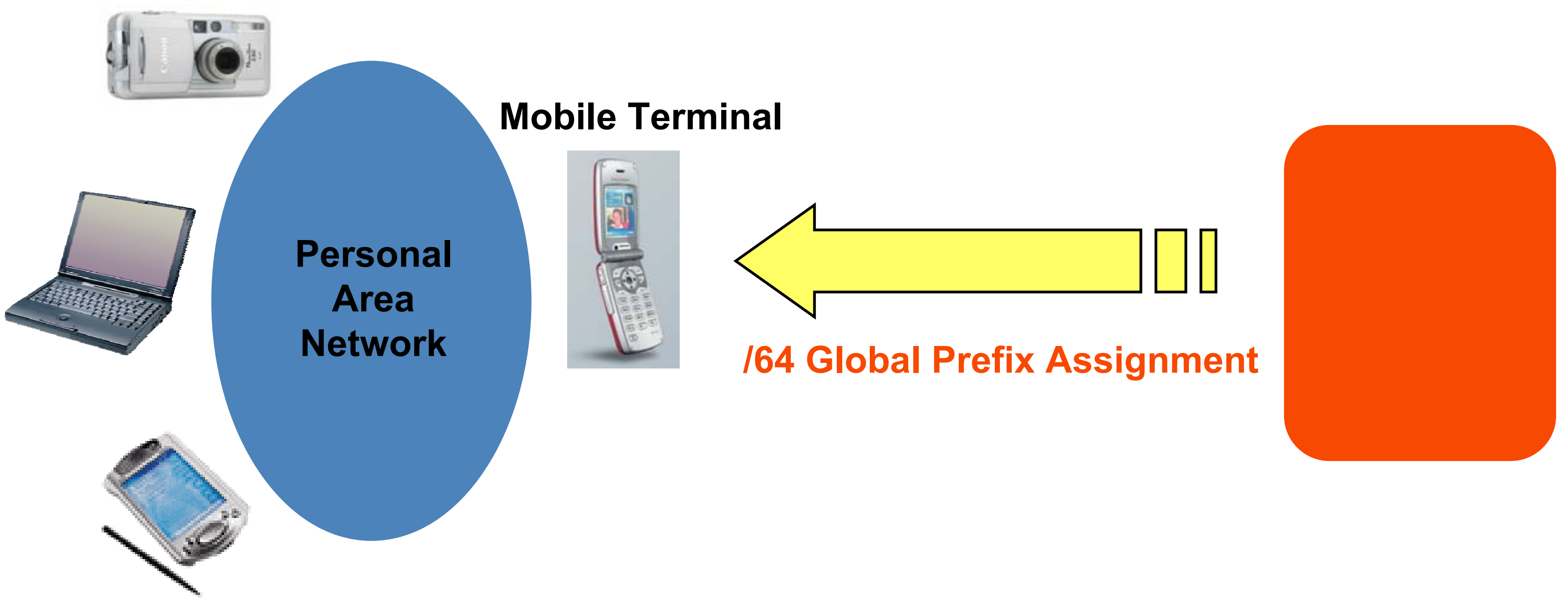
# Open Source supporting IPv6

- Kame www.kame.net. IPv6 open source stack ported to BSD main tree
- USAGI www.linux-ipv6.org. Patch to Linux IPv6 kernel.
- TAHI http://www.tahi.org/. Set of test to assure conformance and interoperability. Collaborate in the IPv6 ready logo program
- Nautilus6 http://www.nautilus6.org/ IPv6 mobility related technologies
- Ferrara LUG has a web page on IPv6

# IPv6 socket API

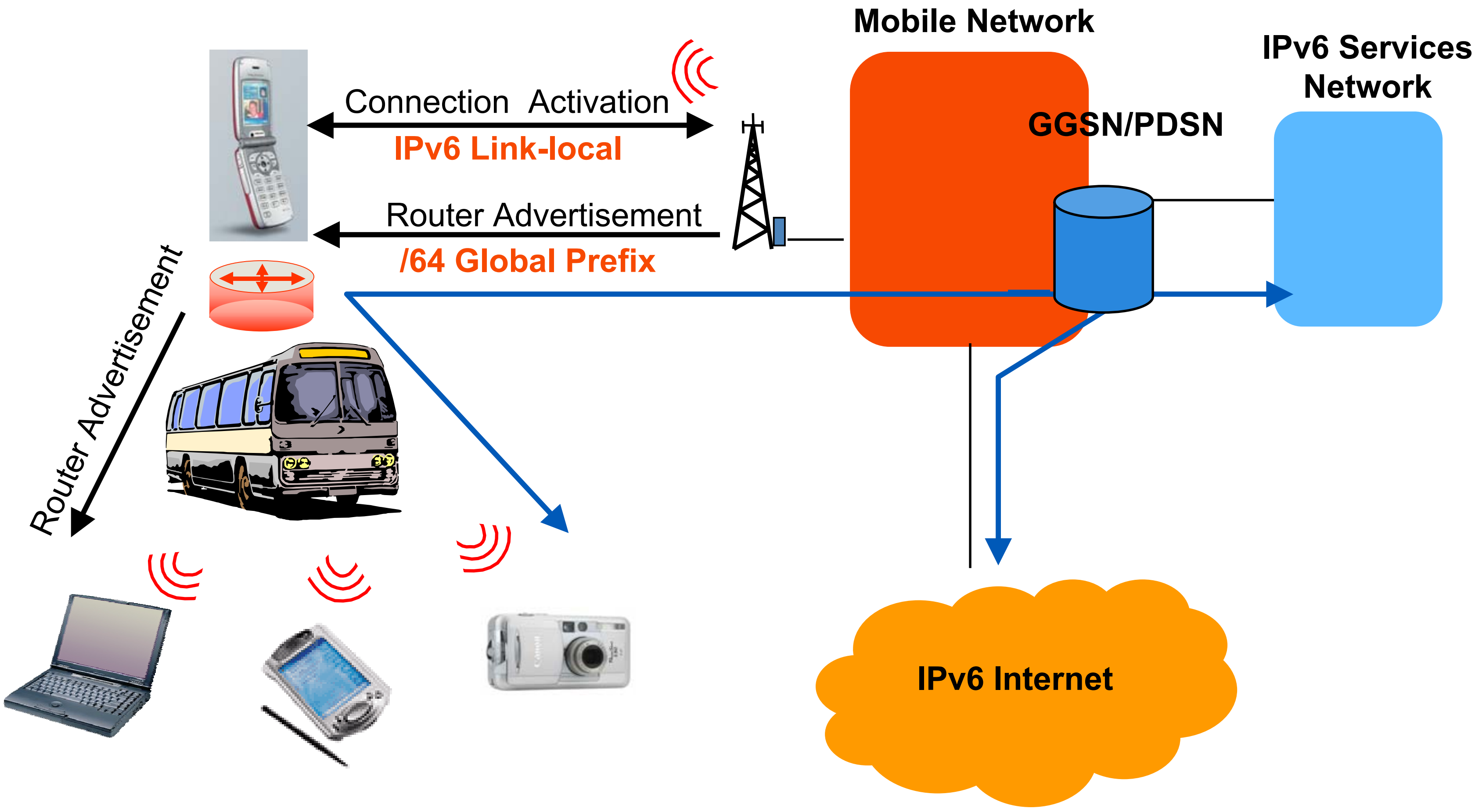
- Application developers are required to support both IPv4 and IPv6
- Basic Socket Interface Extensions for IPv6 (RFC 3493)
- Advanced Sockets Application Program Interface (API) for IPv6 (RFC 3542)
- Porting of application from IPv4 to IPv6 require changes in
  - Structures used (e.g. **sockaddr_in6** instead of **sockaddr_in**)
  - **INADDR_ANY** and **INADDR_LOOPBACK** are not IPv6 compatible and need to be replaced
  - Protocol family AF_INET and PF_INET are replaced by AF_INET6 and PF_INET6
  - Some function like gethostbyname() that only support IPv4 should be replaced by functions such as getaddrinfo() that supports both IPv4 and IPv6

# IPv6 future usage: Prefix Assignment




- IETF/3GPP successful collaboration produced future-proof IPv6 standard for Mobile Networks
- Each Mobile Terminal is assigned a unique /64 IPv6 prefix
  - Personal Area Networks and Moving Networks
  - Privacy addressing

# Future usage of IPv6 in Moving Networks



# Summary

- IPv6 is not backward compatible with IPv4
- The area interested by the transition is anything has to with the internet today and tomorrow from applications to networking.
- The community for open source has been very active in developing applications and stacks for IPv6
- Now is the time to get it spread and start to use the new Internet and its new possibilities
- IPv6 will enable the direct communication between hosts in the Internet and reachability