Advanced Computer Networking (ACN)

IN2097 – WiSe 2026–2027

Prof. Dr.-Ing. Georg Carle, Dr. Sebastian Gallenmüller

Christian Dietze, Marcel Kempf, Lorenz Lehle

Chair of Network Architectures and Services
School of Computation, Information and Technology
Technical University of Munich

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Internet Protocol Version 6

Summary

How to Run Out of Addresses

Bibliography

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

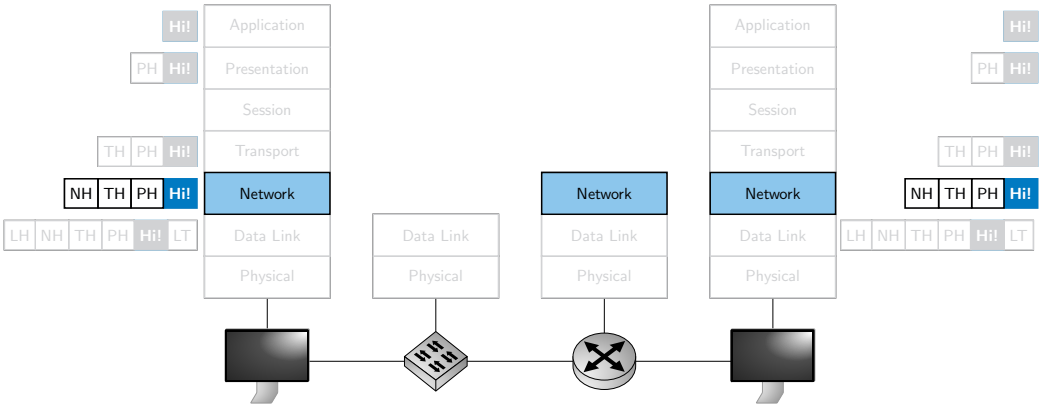
Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary


Hi!
Application
PH Hi!
Presentation
Session
TH PH Hi!
Transport
NH TH PH Hi!
Network
LH NH TH PH Hi! LT
Data Link
Physical
Data Link
Physical
Network
Data Link
Physical
Application
Presentation
Session
Transport
Network
Data Link
Physical
Hi!
PH Hi!
TH PH Hi!
NH TH PH Hi!
LH NH TH PH Hi! LT

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

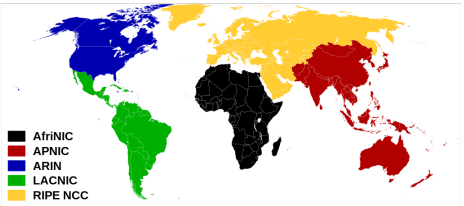
Motivation
IPv4 Address Space Exhaustion

IPv4 address space is too small

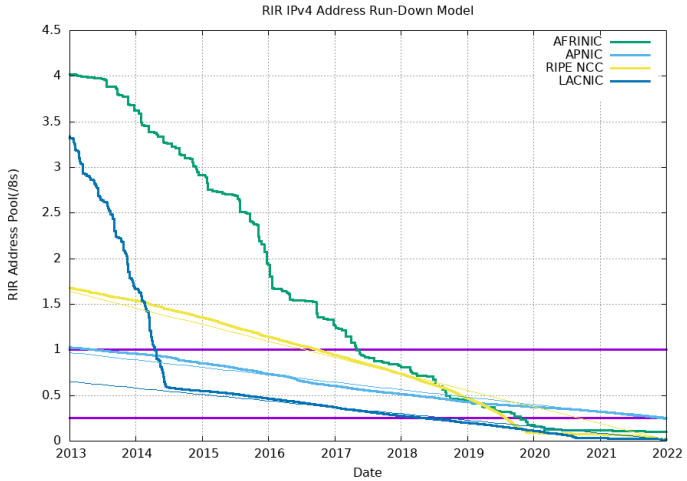
- Usage of NAT devices (home routers)
- Deployment of Carrier-Grade-NAT (CGN)
- IPv4 addresses are no longer end-to-end reachable

Timeline of exhaustion

- 3 February 2011: IANA assigns last networks to RIRs
- 14 April 2011: APNIC depletion
- 14 September 2012: RIPE depletion
- 10 June 2014: LACNIC depletion
- 24 September 2015: ARIN depletion
- All RIRs have transition reserves
- AFRINIC is not yet depleted (1,046,784 IPv4 addresses available as of April 2025)



Motivation
Overview of Address Space Exhaustion[1]



1 Source: http://www.potaroo.net/tools/ipv4/

Overview of Address Space Exhaustion: RIPE NCC

RIPE NCC ran out of addresses on 2019-11-25[2]

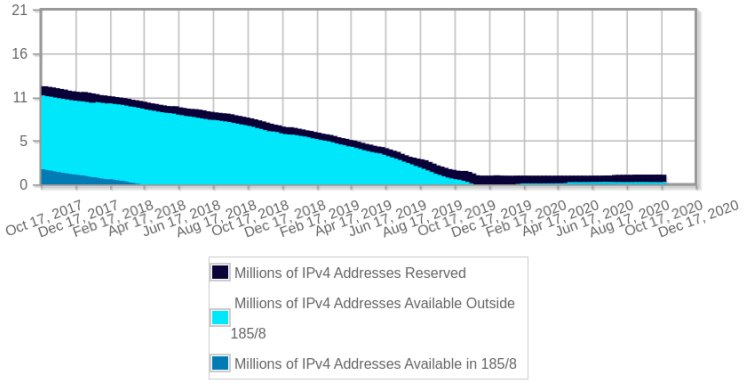


Figure 1: RIPE NCC IPv4 Pool — Last 36 Months (2020-10-27)[3]

2 https://www.ripe.net/manage-ips-and-asns/ipv4/getting-ready-for-ipv4-run-out

3 https://www.ripe.net/manage-ips-and-asns/ipv4/ipv4-available-pool

Overview of Address Space Exhaustion: RIPE NCC

RIPE set up a waiting list for LIRs to request IPv4 addresses

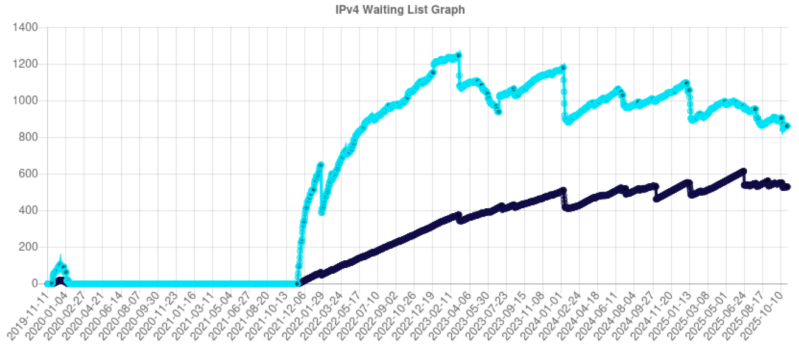


Figure 2: RIPE Waiting List (2026-11-02)[4]

4 https://www.ripe.net/manage-ips-and-asns/ipv4/ipv4-waiting-list

Motivation
RIPE 93



Figure 3: RIPE 93: Discussion of policies on assigning IPv4 pools to IXPs[5]

5 https://ripe93.ripe.net/on-site/photo-gallery/1147/

Motivation

- IPv6 designed as successor to IPv4
- IPv4 and IPv6 are not compatible by design
- First proposed in 1995 [RFC 1883] [1]
- First standard dates back to 1998 [RFC 2460] [2]
- Current standard was finalized in 2017 [RFC 8200] [3]

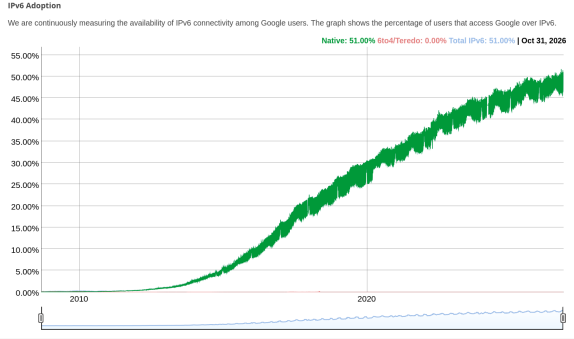


Figure 4: IPv6 adoption over time (2026-10-28)[6]

6 https://www.google.com/intl/en/ipv6/statistics.html

Motivation
IPv6 Capable[7]

Region	2026	2025	2024	2023	2020
World	44.07%	41.84%	39.94%	35.38%	27.33%
Americas	51.86%	49.28%	43.82%	42.57%	33.34%
Asia	50.34%	47.78%	47.20%	40.13%	31.34%
Oceania	43.43%	39.71%	39.90%	35.05%	25.78%
Europe	38.08%	33.31%	33.27%	31.80%	22.03%
Africa	6.72%	3.92%	3.29%	1.99%	1.41%

Country	2026	2025
France	86.05%	77.87%
India	76.35%	78.53%
Saudi Arabia	74.80%	69.58%
Germany	73.01%	65.36%
Malaysia	72.51%	70.12%
Belgium	72.02%	69.22%
Åland Islands	70.17%	63.12%
Israel	65.06%	62.38%

7 Based on: https://stats.labs.apnic.net/ipv6 (2026-11-03, 2025-10-28, 2024-11-08, 2023-10-02, 2020-10-27)

Motivation
IPv6 Adoption Worldwide

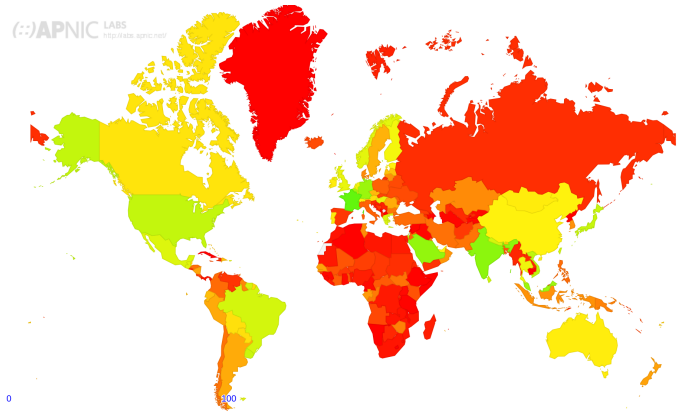


Figure 5: Percentage of IPv6 capability by country (2026-11-02)[8]

8 Source: https://stats.labs.apnic.net/ipv6

Motivation
IPv6 Adoption Worldwide – Comparison 2025

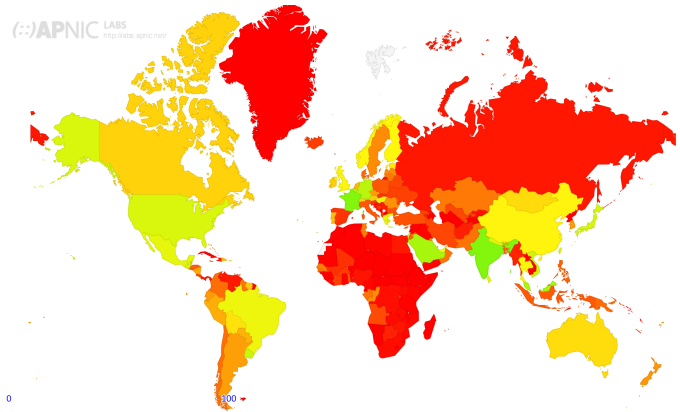


Figure 6: Percentage of IPv6 capability by country (2025-11-02)[9]

9 Source: https://stats.labs.apnic.net/ipv6

Motivation
Goals

- Better scalability
  - Larger address space
- Easier deployment
  - Simplified header, easier implementation
  - Better auto-configuration (SLAAC)
  - Updated stateful configuration (DHCPv6)
- Easier analysis
  - Better flow associativity (see flow label)

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

IPv6 Addressing

As in IPv4, an IPv6 unicast address identifies an interface connected to an IP subnet

IPv6 routinely allows each interface to be identified by several addresses

- $\rightarrow$ Usually only one IPv4 address per interface

IPv6 addresses use 128 bits vs. 32 bits for IPv4

- $2^{128} = 3.4 \times 10^{38}$ IPv6 addresses
- Estimated 10^{22} stars in the universe
- 3×10^{31} unique IPv6 addresses would be assigned after 1 trillion years if a new IPv6 address were assigned every picosecond

IPv6 Addressing
IPv6 Address Representation

An IPv6 address is represented as 8 groups of 4 hexadecimal digits (16 bits), separated by colons (`:`)

- Example: `2001:0db8:85a3:0000:0000:8a2e:0370:7334`

Leading zeroes in a group may be omitted, but each group must retain at least one hexadecimal digit.

- `2001:0db8:85a3:0000:0000:8a2e:0370:7334` → `2001:db8:85a3:0:0:8a2e:370:7334`

Two or more consecutive groups of zero value may be replaced with a single empty group using two consecutive colons (`::`), but the substitution may only be applied once in the address

- `2001:db8:85a3:0:0:8a2e:370:7334` → `2001:db8:85a3::8a2e:370:7334`
- The localhost (loopback) address `0:0:0:0:0:0:0:1` is written as `::1`
- Representations are shortened as much as possible. Only the longest sequence of consecutive all-zero fields – if there are two equally long sequences, the first – is replaced by a double-colon.

Lowercase is recommended over uppercase:

- Example: `2001:db8::1` is preferred over `2001:DB8::1`

Representing IP addresses and port numbers using brackets (`[ ]`)

- IPv4: `192.168.1.1:80`
- IPv6: `[2001:db8::1]:80`

IPv6 Addressing
IPv6 Address Representation – Examples

IPv6 Address		Correct representation
`2001:0db8::0001`	→	`2001:db8::1`
`2001:db8:0:0:0:0:2:1`	→	`2001:db8::2:1`
`2001:db8:0:0:1:0:0:1`	→	`2001:db8::1:0:0:1`
`2001:db8:0:1:1:1:1:1`	→	`2001:db8:0:1:1:1:1:1`

RFC 5952: A Recommendation for IPv6 Address Text Representation

Note: The `2001:db8::/32` subnet is used only in documentation.

Notable IPv6 addresses

Facebook	`2a03:2880:fffe:c:face:b00c::35`
Sprint.net	`2600::`
BBC.com	`2001:41c1:400c::bbc:1`

IPv6 Addressing
IPv6 Address Subnets

The IPv4 CIDR notation is also used for IPv6:

- `address/prefix` → `2001:db8::1/64`

The design of the IPv6 address space follows a very different design philosophy compared to IPv4.

In IPv6, the address space is considered large enough for the foreseeable future, and a local area subnet always uses 64 bits for the host portion of the address, designated as the interface identifier, while the most-significant 64 bits are used as the routing prefix.

- First n bits: Global routing prefix
- Next $64 - n$ bits: Subnet ID
- Last 64 bits: Interface ID

IPv6 Addressing
Address Scopes

- Unicast: From one sender to exactly one receiver
- Multicast: From one sender to all members in a group (one or more)
- Anycast: From one sender to one member of a group
- Broadcast: Not used in IPv6

IPv6 Addressing
Address Types

Global-Unicast IPv6 Addresses

- Globally unique and addressable for unicast communication

Site-Local IPv6 Addresses [RFC 3513][RFC 3879]

- Developed at the beginning of IPv6
- Intended as equivalent to private IPv4 addresses
- Early deprecation in 2004

Unique-Local IPv6 Addresses [RFC 4193]

- Locally addressable
- High likelihood of global uniqueness
- Part of the `fc00::/7` subnet

Link-Local IPv6 Addresses

- Addressable on the link
- Can be globally unique but only have to be unique on the link
- Part of the `fe80::/10` subnet

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

Header Format
IPv4 Header

Just as a reminder:

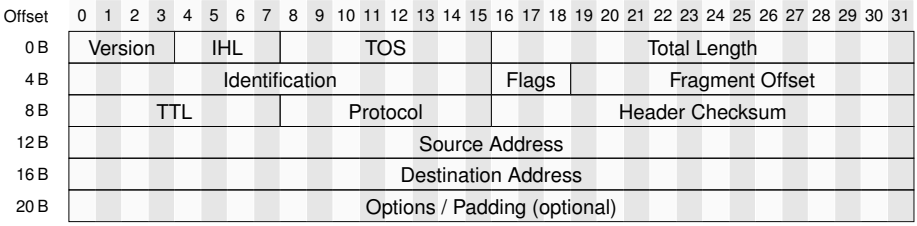


Figure 7: IPv4 Header

- 12 Fields + Options
- 20 Bytes + Options

Header Format
IPv6 Header

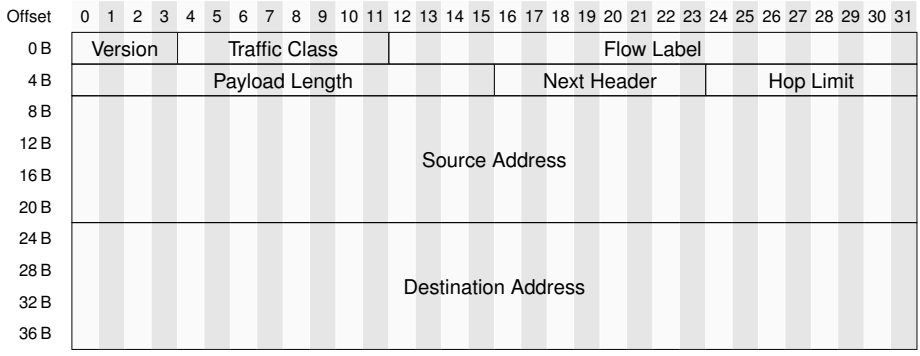


Figure 8: IPv6 Header

Version	Always 0x6
Traffic Class	QoS indicator (including DSCP and ECN bits)
Flow Label	Same for each flow
Payload Length	Length of payload in bytes
Next Header	Type of payload (UDP, TCP, ...)
Hop Limit	No. of hops left until discarded
Source Address	IPv6 address of sender
Destination Address	IPv6 address of receiver

Header Format
IPv6 vs. IPv4 Header

Changes between IPv4 and IPv6:

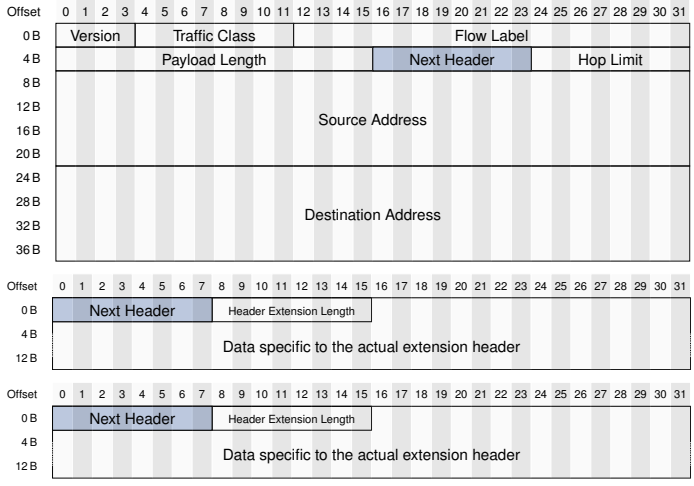
IPv4		**IPv6**
Header Length	→	Removed
Header Checksum	→	Removed
TOS	→	Traffic Class
Total Length	→	Payload Length
Fragmentation	→	Extension Header
TTL	→	Hop Limit
Protocol	→	Next Header
Options	→	Extension Headers

Header Format
IPv6 Flow Label

- 20 bit Flow Label field to identify specific flows needing special QoS
- Traditional IPv4 way of specifying flows:
  - 5-tuple: source and destination IP addresses, source and destination port numbers, and protocol type
- Some of these fields may be unavailable due to fragmentation, encryption, or because they are located after extension headers
- With flow labels, each source chooses its own flow label value. Routers use IPv6 source address + flow label to identify distinct flows

Header Format
Extensions

The headers form a chain, using the Next Header fields.



Header Format
Extensions

Administered by IANA in a central registry[10]:

Number	Description	Reference
0	IPv6 Hop-by-Hop Option	[RFC 8200]
43	Routing Header for IPv6	[RFC 8200][RFC 5095]
44	Fragment Header for IPv6	[RFC 8200]
50	Encapsulating Security Payload	[RFC 4303]
51	Authentication Header	[RFC 4302]
60	Destination Options for IPv6	[RFC 8200]
135	Mobility Header	[RFC 6275]
139	Host Identity Protocol	[RFC 7401]
140	Shim6 Protocol	[RFC 5533]
253	Use for experimentation and testing	[RFC 3692][RFC 4727]
254	Use for experimentation and testing	[RFC 3692][RFC 4727]

10 https://www.iana.org/assignments/ipv6-parameters

Header Format
Extensions

Hop-by-Hop Options Header

- TLV coded options processed by every hop along the path
- Jumbo payload option for packets larger than 65 535 B [RFC 2675]
- Router alert option [RFC 2711]

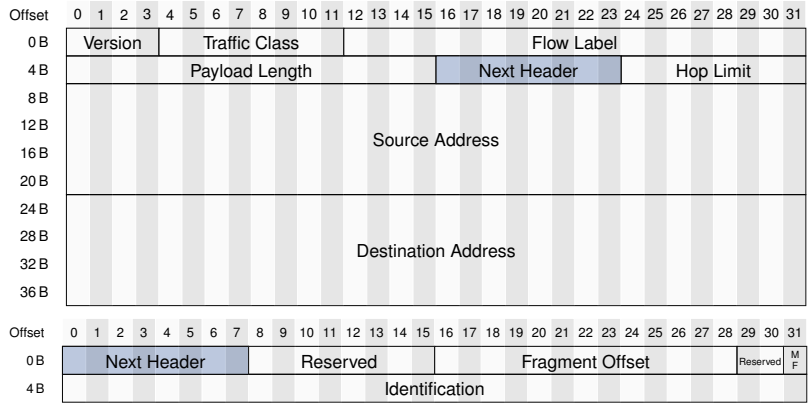
Routing Header

- Strict or loose source routing
- Similar to the IPv4 Source route and Record route options

Fragment Header

- Only the source can fragment packets in IPv6
- The source must use Path MTU Discovery (RFC 8201) or send at most 1240 B payload
- Fragmentation information
  - Offset shifted 3 bits (i. e. a multiple of 8 B as in IPv4)
  - Identifier is 32 bits (16 bits in IPv4)

Header Format
Fragment Extension Header



Header Format
Extensions

Authentication Header (IPsec)

- To validate the message sender and ensure data integrity

Encapsulating Security Payload Header (IPsec)

- To provide confidentiality and guard against eavesdropping

Destination Options Header

- TLV coded options processed by destination only

Mobility Header

- Parameters used with Mobile IPv6

Host Identity Protocol (HIP)

- To enable continuity of communications across IP address changes

Shim6 Protocol

- Multihoming Shim Protocol for IPv6 with failover and load-sharing properties

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

ICMPv6

- ICMP has been revised along with the IPv6 development, with a new version: ICMPv6 [RFC 4443]
- Several protocols associated with IPv6 use the ICMPv6 message format to exchange messages

Associated protocols:

- Neighbor Discovery Protocol (NDP) is the IPv6 equivalent of ARP in IPv4 [RFC 4861]
- Secure Neighbor Discovery (SEND) is an extension of NDP with extra security [RFC 3971]
- Multicast Listener Discovery (MLD) is used by IPv6 routers for discovering multicast listeners on a directly attached link [RFC 2710] [RFC 9777]
- Multicast Router Discovery (MRD) allows discovery of multicast routers

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

Neighbor Discovery Protocol (NDP)

Goal

- ARP equivalent for IPv6: Resolve IPv6 addresses to MAC addresses
- Provide information about routers

Realization

- Part of ICMPv6 [4]
- Split into “Neighbor Solicitation” and “Neighbor Advertisement”

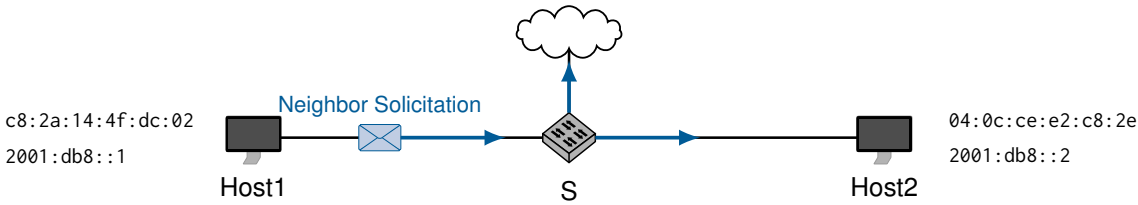
Neighbor Solicitation

- Ask for the MAC address of the interface configured for a given IPv6 address
- Destination IPv6 address (“Solicited Node Address”):
  - `ff02::1:ffXX:XXXX`
  - Insert the lowest 24 bits of the given IPv6 address at the end of the destination IPv6 address
- Destination MAC address:
  - `33:33:XX:XX:XX:XX`
  - Insert the lowest 32 bits of the “Solicited Node Address”

Neighbor Advertisement

- Answer a Neighbor Solicitation
- Use MAC and IPv6 addresses of the destination host

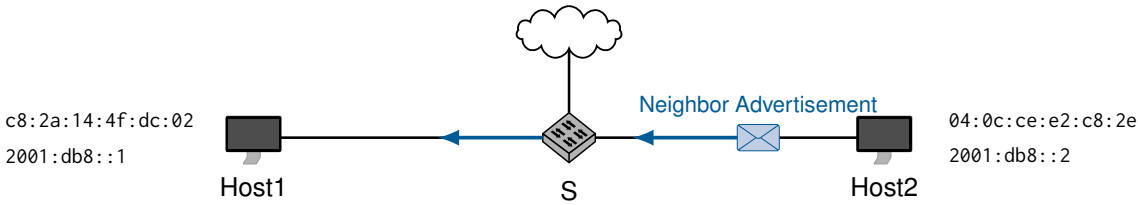
Neighbor Discovery Protocol (NDP)
Neighbor Solicitation / Advertisement – Example



Destination MAC Address: `33:33:ff:00:00:02`

Destination IPv6 Address: `ff02::1:ff00:2`

Neighbor Discovery Protocol (NDP)
Neighbor Solicitation / Advertisement – Example



Destination MAC Address: `c8:2a:14:4f:dc:02`

Destination IPv6 Address: `2001:db8::1`

Neighbor Discovery Protocol (NDP)
Router Solicitation / Advertisement [4]

Router Solicitation (RS)

- Prompt all routers on this segment to send a Router Advertisement
- Normally sent when an interface comes up

Router Advertisement (RA)

- Sent to the all-nodes multicast address at regular intervals by all routers
- Contains information about the network segment:
  - Autoconfiguration methods (SLAAC, DHCPv6)
  - Prefix information
  - Route information
  - Link MTU
  - Link-layer address of the router

Problem: Everybody can claim to be a router

- Use Secure Neighbor Discovery (SEND)
- Use RA Guard to filter unauthorized RAs [RFC 6105]

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

IP Multicast

Goal

- Efficient one-to-many delivery
- Sender transmits a single packet to a group address
- Routers replicate packets when paths diverge
- Hosts can join multicast groups to receive corresponding traffic

Special role in IPv6

- No broadcast in IPv6 $\rightarrow$ replaced by multicast [RFC 4291]
- Every IPv6 host is a multicast listener as soon as it configures an address
- NDP, SLAAC and router discovery rely on multicast [RFC 4861][RFC 4862]

IP Multicast
IPv6 Multicast Address Format

Multicast addresses have the following convention:

- First 8 bits set to `11111111`
- Next 4 bits as flags (permanent or transient address)
- Next 4 bits as scope:
  - `0000` Reserved
  - `0001` Interface-local
  - `0010` Link-local
  - `1000` Organization-local
  - `1110` Global
  - `1111` Reserved
- Last 112 bits: Group ID

Fixed (well-known) addresses

- All nodes: `ff02::1` / `224.0.0.1`
- All routers: `ff02::2` / `224.0.0.2`
- Solicited-node: `ff02::1:ffXX:XXXX` (lowest 24 bits of the unicast address)
- mDNS: `ff02::fb` / `224.0.0.251`
- Other protocol-specific groups assigned by IANA

IP Multicast
Group Membership (Local Network)

- Hosts register for groups with their local router
  - IPv4: Internet Group Management Protocol (IGMP) [RFC 1112]
  - IPv6: Multicast Listener Discovery (MLD), part of ICMPv6 [RFC 2710]
- Routers forward group traffic only onto links with at least one listener

Message types

- Membership/Listener Query: multicast routers query the listening state of neighboring interfaces (`224.0.0.1` / `ff02::1`)
- Membership/Listener Report: host reports interest in a group or set of groups

IGMPv3 [RFC 9776] / MLDv2 [RFC 9777]

- Any-Source Multicast (ASM): receive traffic from any sender to the group
- Source-Specific Multicast (SSM): receive traffic only from selected sources

IP Multicast
Multicast Routing (Internet)

Inside an AS: PIM Sparse Mode (PIM-SM) [RFC 7761]

- Rendezvous Point (RP) serves as the distribution hub
- Sources unicast their traffic to the RP
- Routers with receivers in the local network join using PIM Join and receive traffic
  $\rightarrow$ shared tree rooted at RP
- Once established, routers may switch to a source-specific shortest-path tree

Across ASs

- Multiprotocol BGP (MBGP) [RFC 4760]
- Multicast Source Discovery Protocol (MSDP) [RFC 3618]
- In practice: organizational differences and lack of support

IP Multicast
Multicast on Layer 2

IP to MAC address mapping

- IPv4: `01:00:5e` + low-order 23 bits [RFC 1112]
  Example: `224.0.0.251` → `01:00:5e:00:00:fb`
- IPv6: `33:33` + low-order 32 bits [RFC 2464]
  Example: `ff02::fb` → `33:33:00:00:00:fb`
- Not injective → hosts discard frames for groups they have not joined

Layer 2 switches

- Handle multicast frames like broadcast frames
- Frames are flooded to all ports

IGMP / MLD snooping [RFC 4541]

- Switch inspects (“snoops” on) IGMP/MLD messages between hosts and routers
- Maintains a forwarding table: group address → set of ports
- Not standardized → varies across vendors

IP Multicast
Applications and Challenges

Applications

- ICMPv6 (NDP, DAD), DHCPv6
- Synchronous IPTV: each channel is a group, set-top box joins the group of the selected channel; now largely discontinued
- Multicast DNS (mDNS) [RFC 6762]: resolves domains under `.local.` via `224.0.0.251` / `ff02::fb`, port 5353

Challenges of wide-area multicast

- Routers maintain per-group state
  → scales poorly with many groups
- Inter-AS multicast not deployed on the Internet
  → multicast IPTV only within the provider's own network

IP Multicast
IPv4 vs. IPv6 Multicast

	IPv4	**IPv6**
Address range	`224.0.0.0/4`	`ff00::/8`
Broadcast	Yes (e. g. ARP)	No, replaced by multicast
Multicast support	Optional	Mandatory
Membership	IGMP	MLD (ICMPv6)
MAC mapping	`01:00:5e` + 23 bits	`33:33` + 32 bits

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

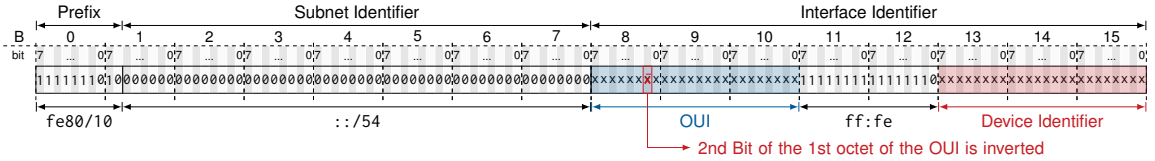
IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

Stateless Address Autoconfiguration (SLAAC)



Use case

- Automatic configuration of link-local address on system startup

Address generation

1. Create modified EUI-64 address
   - Always in subnet `fe80::/64`
   - First 24 bits of Interface Identifier: First 24 bits of MAC address
     Flip second least-significant bit of the first octet (universal/local bit)
   - “Middle” 16 bits: Always `ff:fe`
   - Last 24 bits: Last 24 bits of MAC address
2. Perform Duplicate Address Detection (DAD)
3. Assign address to interface

Stateless Address Autoconfiguration (SLAAC)

- Use link-local address and interface ID
- Hosts join all-nodes multicast address (`ff02::1`)
- Hosts perform DAD with the solicited-node multicast address
- Hosts communicate with routers using all-routers multicast address (`ff02::2`)
- ICMPv6 Router Solicitation sent by host to request additional information
- ICMPv6 Router Advertisement sent by router to inform host about prefixes for site and global addresses

Privacy and Security Considerations

SLAAC uses a modified MAC address, which makes it possible to trace a device

- Privacy Extensions prevent tracking [RFC 4941]
- Use random 64 bit number for the host part
- Change the number regularly

DHCPv6

SLAAC and Router Advertisements can configure an IPv6 host and provide connectivity.

Why do we need DHCPv6?

- Can be used to provide prefix information
- Can be used to assign fixed addresses based on device identifiers
- Can be used to provide DNS information
- Has to be used to provide boot information for Netboot

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

IPv4 to IPv6 Transition
Going from IPv4 to IPv6

Because of the large number of systems and actors on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.

Three transition strategies have been devised by the IETF:

- Dual Stack
  - All hosts run both the IPv4 and IPv6 protocol stacks until all of the Internet runs IPv6
  - Avoids the complexities of tunneling, such as security issues, increased latency and management overhead

IPv4 to IPv6 Transition
Dual Stack

Approach

- Assign each end user one public IPv4 address and one public IPv6 subnet
- End device chooses to use IPv4 or IPv6
- Both networks are directly accessible

Problems

- IPv4 addresses might already be exhausted
- Supporting IPv4 and IPv6 inside the ISP network at the same time is expensive and complex

Solutions

- No solution for address exhaustion in full Dual Stack (see DS Lite)
- ISPs can transparently tunnel IPv4 in IPv6 (or the other way round), and only set up one network, which provides IPv4 and IPv6

IPv4 to IPv6 Transition
Dual Stack Lite

Approach

- Same as full Dual Stack, but private instead of public IPv4 address
- Deployment of a Carrier-Grade-NAT

Problems

- Peer-to-peer communication almost impossible
- Mostly affects VoIP and gaming

No real solution for this problem is available

IPv4 to IPv6 Transition
Going from IPv4 to IPv6

Because of the large number of systems and actors on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.

Three transition strategies have been devised by the IETF:

- Dual Stack
  - All hosts run both the IPv4 and IPv6 protocol stacks until all of the Internet runs IPv6
  - Avoids the complexities of tunneling, such as security issues, increased latency and management overhead

IPv4 to IPv6 Transition
Going from IPv4 to IPv6

Because of the large number of systems and actors on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.

Three transition strategies have been devised by the IETF:

- Dual Stack
  - All hosts run both the IPv4 and IPv6 protocol stacks until all of the Internet runs IPv6
  - Avoids the complexities of tunneling, such as security issues, increased latency and management overhead
- Tunneling
  - Encapsulation of IPv6 packets in IPv4
  - Some automatic tunneling techniques: 6to4, Teredo, ISATAP

IPv4 to IPv6 Transition
Tunneling

Approach

- Encapsulate IPv6 packets in IPv4 packets
- Implementable in a variety of forms

Implementations

- Any general-purpose VPN service, with an IPv6 overlay and IPv4 underlay network (IPsec, OpenVPN, Cisco AnyConnect, ...)
- Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs) [5]
  - Developed by Microsoft
  - Components: Client, Server, Relay
  - Client: Has only IPv4 and is possibly behind a NAT
  - Server: Helps client to set itself up and establish the tunnel
  - Relay: Forwards traffic from and to client
  - IPv4 client and server addresses are embedded in IPv6 address
  - Teredo prefix: `2001::/32`
  - Based on NAT UDP hole punching

IPv4 to IPv6 Transition
6to4 [6]
Approach

- Local router encapsulates IPv6 in IPv4
- IPv4 packet gets transmitted to the “nearest” relay
- Source IPv4 address is embedded in the IPv6 address
- 6to4 prefix: `2002::/16`, append public IPv4 address to get /48 IPv6 subnet
- Address IPv4 packets to 192.88.99.1 (IPv4 Anycast)
- Deploy dual-stack 6to4 relays, and 6to4 routers as NAT router

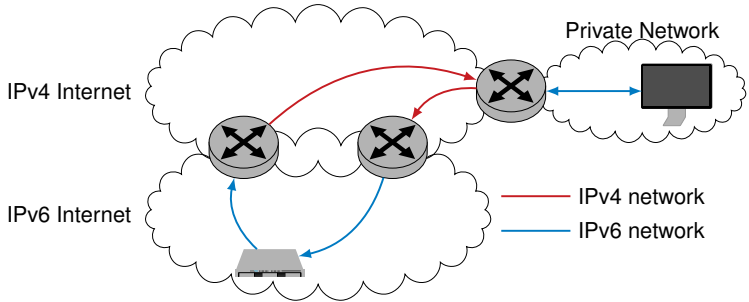


Figure 9: 6to4 Mechanism

IPv4 to IPv6 Transition
6to4 Address Example

Building an IPv6 address from an IPv4 address

- IPv4 address (dotted decimal): `131.159.255.1` (Gateway/Router)
- IPv4 address (hexadecimal): `0x83 0x9f 0xff 0x01`
- IPv6 6to4 prefix: `2002::/16`
- Resulting gateway IPv6 prefix: `2002:839f:ff01::/48`

Benefit

- Devices behind the gateway can configure an address inside the `2002:839f:ff01::/48` subnet
- Gateway encapsulates all IPv6 packets inside IPv4 packets
- All IPv6 packets destined to `2002:839f:ff01::/48` reach the gateway in an encapsulating IPv4 packet
- Completely transparent to clients

IPv4 to IPv6 Transition
Going from IPv4 to IPv6

Because of the large number of systems and actors on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.

Three transition strategies have been devised by the IETF:

- Dual Stack
  - All hosts run both the IPv4 and IPv6 protocol stacks until all of the Internet runs IPv6
  - Avoids the complexities of tunneling, such as security issues, increased latency and management overhead
- Tunneling
  - Encapsulation of IPv6 packets in IPv4
  - Some automatic tunneling techniques: 6to4, Teredo, ISATAP

IPv4 to IPv6 Transition
Going from IPv4 to IPv6

Because of the large number of systems and actors on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.

Three transition strategies have been devised by the IETF:

- Dual Stack
  - All hosts run both the IPv4 and IPv6 protocol stacks until all of the Internet runs IPv6
  - Avoids the complexities of tunneling, such as security issues, increased latency and management overhead
- Tunneling
  - Encapsulation of IPv6 packets in IPv4
  - Some automatic tunneling techniques: 6to4, Teredo, ISATAP
- Header translation
  - Stateless IP/ICMP Translation (SIIT)
  - Defines a class of IPv6 addresses called IPv4-translated addresses
  - Uses the prefix `::ffff:0:0:0/96`; addresses may be written as `::ffff:0:a.b.c.d`
  - Allows IPv6-only hosts to communicate with IPv4-only hosts

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

Summary

IPv6 offers:

- 128 bit address space → 340 undecillion addresses
- Revised and simplified header format
- New options and extensions
- Increased security measures

IPv6 uses hexadecimal colon notation with abbreviation methods

IPv6 has three address types: unicast, anycast, and multicast

ICMPv4, ARP, RARP, and IGMP are replaced with ICMPv6

IPv4 to IPv6 transition strategies are based on Dual Stack and tunneling

IPv6 adoption:

- IPv6 is supported by all major operating systems and all major router vendors
- Still some work to be done for complete adoption

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

How to Run Out of Addresses

Based on a talk at RIPE 77 by Benedikt Stockebrand [7]

Host-Density Ratio [8]: $HD = \frac{log_2(k)}{log_2(n)}$

- Address assignment efficiency
- Measured logarithmically
- k is the number of encoded addresses
- $log_2(k)$ is the number of required bits
- e. g. for 5 addresses, $log_2(5) \approx 2.3$ bits
- $log_2(n)$ is the number of overall address bits available

$\rightarrow$ Waste bits, not addresses

How to Run Out of Addresses
Infinite Amount of Addresses

Total IPv6 Addresses:

- 2^{128} = 340282366920938463463374607431768211456 $\approx 3.4 \times 10^{38}$

How to Run Out of Addresses
Infinite Amount of Addresses

Total IPv6 Addresses:

- 2^{128} = 340282366920938463463374607431768211456 $\approx 3.4 \times 10^{38}$

64 bits used for local area subnets:

- 2^{64} = 18446744073709551616 $\approx 1.8 \times 10^{19}$
- Only $\approx 5.4^{-20}$% remaining

How to Run Out of Addresses
Multi-Layer Delegations

Each delegation requires 1 bit :

- IANA
- RIR
- LIR
- Company
- Branches
- Teams

$\rightarrow$ Waste 6 bits

How to Run Out of Addresses
Multi-Layer Delegations

Each delegation requires 1 nibble:

- IANA
- RIR
- LIR
- Company
- Branches
- Teams

→ Waste 24 bits

How to Run Out of Addresses
Waste Bits

Digression: bit magic in programming

- We have fixed-size pointers of 32 bits
- We only reference 4 million items
- $\rightarrow$ $log_2(4000000) \approx 22$ bits
- $\rightarrow$ 10 bits can be used to encode information to save memory

Application: encode information into addresses

- Example: A company with several branches in Germany wants to encode the postal code of each branch into addresses.
- Germany has 5-digit postal codes (00000-99999)
- $log_2(99999) \approx 17$ bits

How to Run Out of Addresses
Missing Regulations/Knowledge

A lot of addresses are simply lost due to missing regulations and knowledge about actual requirements.

- There are enough addresses, therefore I want as many as possible.
- I have no idea how many addresses I need for my company.
- I want more addresses than other companies.

How to Run Out of Addresses
What now?

It is not that bad yet!

- Theoretically, there are enough IPv6 addresses
- Early IPv4 deployments had similar problems
  - → Transition to CIDR
- Prefix delegation and address assignments have to be done carefully

Internet Protocol Version 6

Introduction

Motivation

IPv6 Addressing

Header Format

ICMPv6

Neighbor Discovery Protocol (NDP)

IP Multicast

Autoconfiguration

IPv4 to IPv6 Transition

Summary

[1] S. Deering and R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, http://tools.ietf.org/html/rfc1883, 1995.

[2] S. Deering and R. Hinden, Internet Protocol, Version 6 (IPv6) Specification, http://tools.ietf.org/html/rfc2460, 1998.

[3] D. S. E. Deering and B. Hinden, Internet Protocol, Version 6 (IPv6) Specification, RFC 8200, Jul. 2017. DOI: 10.17487/RFC8200. [Online]. Available: https://rfc-editor.org/rfc/rfc8200.txt.

[4] E. Nordmark, W. Simpson, and H. Soliman, Neighbor Discovery for IP version 6 (IPv6), http://tools.ietf.org/html/rfc4861, 2007.

[5] C. Huitema, Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs), http://tools.ietf.org/html/rfc4380, 2006.

[6] B. Carpenter and K. Moore, Connection of ipv6 domains via ipv4 clouds, http://tools.ietf.org/html/rfc3056, 2007.

[7] B. Stockebrand, The Art of Running Out of IPv6 Addresses, 2018. [Online]. Available: https://ripe77.ripe.net/archives/video/2287/.

[8] A. Durand and C. Huitema, The Host-Density Ratio for Address Assignment Efficiency: An update on the H ratio, https://tools.ietf.org/html/rfc3194, 2001.