IIIT Hyderabad
Advanced Computer Networks
Lab 6 - Basic IPv6 configuration

Contents

1 Basic IPv6 configuration

1.1 Lab equipment

Three PCs would be required to do routing related exercises which are part of this lab. Hence each team / group should have access to three PCs with IPv6 compatible OS installed for lab work

1.2 IPv6 resources

IPv6 should be learned using video tutorial hosted at 6deploy website before performing lab, so that proper understanding of IPv6 is present before one attempts to use it.

Additional very good resources related to IPv6 are available at:

- APNIC website
- 6choice website

1.3 Minimum understanding

One should understand at least following things about IPv6 before proceeding with lab:

- IPv6 address structure and :: shorthand notation
- IPv6 header format and significance of various fields present it.
- Different types of extension headers defined so far and their purpose.
- Recommended order of extension headers within a IPv6 packet
- Following option header formats should be understood properly:
  - Hop-by-hop options header
  - Destination options header
  - Structure of small extension header options that are used to specify individual options in hop-by-hop and destination options header. Also understand significance of higher three bits of option type field.

  - Routing header. Understand how multiple address can be specified in routing header and how destination addresses get changed based on addresses mentioned in IPv6 routing header. This header complements IPv6 loose source routing header, but does not suffers from same 9 IP address limit as IPv4 header.
  - Fragment header. Notice how fields that were present in IPv4 header and helped with fragmentation are moved as it is to this header.
  - No next header.

- One should understand various IPv6 address classes. Given that IPv6 is rapidly evolving one should prefer information available on websites mentioned above over the information mentioned in this pdf as those websites will have more recent information.

- How 48-bit MAC address is converted to IEEE EUI-64 address

- Neighbour discovery protocol

- Duplicate address detection

- Stateless auto-configuration process.

Motivated students should try to understand following headers also to improve their understanding of IPv6:

- Generic ICMPv6 message format

- ICMPv6 error messages reporting
  - Destination unreachable
  - Packet too big
  - Time exceeded
  - Parameter problem

- ICMPv6 information messages related to
  - Echo request
  - Echo reply
  - Neighbour solicitation
  - Neighbour advertizement

  - Redirect
  - Router solicitation
  - Router advertizement
  - Multicast listener query
  - Multicast Listener report
  - Multicast listener done

1.4 Lab work

As part of lab you must capture packets using wireshark for every small step of the lab and verify you understanding of IPv6 protocols and header formats by seeing the description and fields of packets captured.

Do not keep wireshark running continuously in single run to capture all packets sent / received during lab as that much memory would not be available. You would have to restart packet capture after every few steps so that older packets get discarded. You can optionally save interesting packet captures in pcap format for future reference.

1.5 Submission

There is no submission required as part of this lab. But thorough understanding of IPv6 is very important for anyone who wants to have career in network field. So ensure that you do all the tasks properly and go through all reading material.

2 IPv6 notes

Notes on IPv6 are provided as part of handout to help with quick revision of concepts and packet formats related to IPv6. This notes are not meant for beginners to learn IPv6, this are meant for revision / reference.

This is first draft of notes and can have errors. Students are required not to trust everything in this document blindly, it is hoped that they would verify things they are not sure about and inform instructors if they find any errors in the document.

- IPv4 lifetime has been extended using techniques like:
  - NAT
  - DHCP

– CIDR

- Although IPv4 address space is the most important reason for development or migration to IPv6, IPv6 has many other advantages / benefits when compared to IPv4 apart from address size.

2.1 IPv6 advantages

- **128-bit address space :** Allows globally unique address for each device. NAT breaks end-to-end IPsec security and hence interferes with security. With IPv6 NAT is not required as all devices can have globally reachable IP address. Thus IPSec security based secure connection between any two devices is possible.

  Having unique IP address will also affect number and type of applications that can be run on devices like PDAs, Mobile phones etc. With unique IPs mobile phones can have simple servers like FTP and other devices / mobile phones can download the information using FTP protocol. We already have this on Wi-fi based networks using smart phones, but IPv6 would allow one to do the same over Internet using phones.

  It is believed that even if iPv6 address space is used sparingly it would provide more than 1000 IPs per person.

- Simpler and much more efficient header. Major improvements in IPv6 header include reduction of number of fields, fixed header length, removal of header checksum that had to be re-computed at every hop. Instead of options which had restricted space (40 bytes) available in IPv4 header, IPv6 has extension header chain. This is more easy to process and also more flexible in terms of space.

- Better decisions are taken for address allocation and subnet masking. All links in IPv6 have /64 subnet prefix which is fixed. Each site gets /48 based prefix from ISP and can divide organization into 2^{16} /64 subnets.

  Private addresses are not allocated in IPv6. Hence routing tables are very small and extremely efficient. CIDR / Prefix based route aggregation is used to combine various small routes into one single bigger parent route, same as in IPv4 with CIDR.

  In IPv4 ISP migration problems were solved using private addresses. Since private addresses are not present, IPv6 supports mechanisms like

preferred address, address lifetime, multiple address per interface etc. which allow migration from one ISP to other wihtout much difficutly.

- **Mandatory IPsec :** In IPv6 all nodes / end-hosts must support IPsec. Thus it is possible for administrator to enable IPsec on all nodes in network to make network more secure, without worrying about IPSec support.

  IPSec has been part of IPv6 design from start and is not hacked / patched into IPv6 design. Thus IPv6 has been created with security in mind from start as opposed to most protocols where security is added later on.

- IPv6 used multicast instead of broadcast. This can help in improving performance.

- Anycast has been introduced in IPv6, which can allow one to communicate with the nearest node among group of nodes without requiring application / transport level support to locate neartest node.

- ICMPv6 is much improved over ICMPv4. Notable improvements are:

  - Support for auto-configuration
  - Neighbour discovery and duplicate address detection. Thus ARP is no longer required and one does not needs to worry about IP conflict.
  - Support for multicasting making support for separate protocol like IGMP for multicasting unnecessary.

- Built in support for mobile nodes so that mobility is not affected due to network layer limitations.

2.2 IPv6 addressing

- IPv6 addresses are represented with 8 blocks of set of 4 hexadecimal characters. For example: aa01:fe06:0000:0000:0205:0023:a935:bcde

- The above address can be written as aa01:fe06::205:23:a935:bcde

- IPv6 addresses are case-insensitive, that is, hexadecimal part can be expressed in both small or capital letters

- Prefix notation (same as was used in CIDR) is used in IPv6. Hence address can be aa01:fe06::205:23:a935:bcde/64. The same syntax is also used to represent contiguous addresses or networks.

- In IPv6 addresses are assigned to interfaces, that is, multiple addresses can be assigned to single interface. Hence there is no need of sudo interfaces like eth0:0, eth0:1, etc.

  While doing IPv6 labs ensure that you do not have interfaces like eth0:0, eth0:1, etc.

- There are three major types of IPv6 addresses:
  - Unicast
  - Multicast
  - Anycast

- Unicast addresses can be further of many different types:

  Global unicast addresses : These can be treated similar to public IPs in IPv4. These addresses are in range 2000::/3.

  Unique local addresses : These addresses are used for private addressing within a organization. These can be assigned to machines which we do not want accessible from outside organization. These addresses can help in robust communication within site when it is getting renumbered. Even in this case the local IP assigned to machine is unique world-wide.

  This is not similar to many organizations using addresses in range 192.168.0.0/16, 172.16.0.0/12 or 10.0.0.0/8 internally. As in IPv4 two different organizations can use same address in any of these ranges by design, whereas in IPv6 even private addresses used can be unique.

  In early drafts of IPv6 addresses in fec0::/10 were supposed to be used similar to private IP address ranges in IPv4. But later this range / design was deprecated. As per latest drafts organizations can choose some random prefix in FD00::/8 and assign all private IPs with that prefix. Given that the range is so large, it should be very rare that many organizations choose same random prefix

  The newer draft also has provision for assigning private addresses in range FC00::/8 using some world-wide authority to ensure that addresses in FC00::/8 are uniquely assigned.

Link local unicast addresses: Addresses in range fe80::/10 are used as link local address. Hosts generate their 64-bit Interface Identifier (IID) from their 48-bit MAC address. The process of generation of IID from MAC address is explained later in the notes.

Once hosts generate IID they choose fe80::/64 as prefix and do duplicate address detection for fe80::<IID> on the link connected to interface for which IID has been generated. If no other host on same link already has taken the address then node assigns itself address in fe80::/10 as link local address and later on uses this address for future communication with routers on link for further auto-configuration.

IPv4 mapped IPv6 addresses : IPv4 mapped IPv6 addresses are for very specific inter-communication between IPv4 and IPv6. These addresses are used when an IPv4 only client sends a request to IPv6 only application running on dual stacked (IPv4 + IPv6) host.

In this case OS of destination host converts IPv4 address of incoming request to IPv4 mapped IPv6 address before it is passed to application as application supports only IPv6. Replies for application are again converted back to IPv4 and sent to original host which had sent request via IPv4.

IPv4 mapped IPv6 addresses are in range ::FFFF/96. Last 32 bits are taken from 32-bit IP address as it is without any change.

- Two special unicast addresses are:

  Unspecified address (::) This address is used as source when sending machine does not has a address that destination would recognize / reply to. This can never be destination of any IPv6 packet. Routers must never forward packets with source as :: from one link to another.

  Loopback address (::1) This is used as loop-back address. This is similar to 127.0.0.1 address in IPv4. This can be used to check whether current system supports IPv6 or not by using '`ping6 ::1`' command.

- IPv6 introduced concept of anycast. With anycast an IPv6 end host can contact one among group of IPv6 end host, where group of nodes are all identified by single anycast address. In this case routing protocols help in deciding which among group of nodes is nearest to sender and then anycast packet is sent only to nearest nodes. Anycast can help in

finding nearest among group of routers connected to link with specified prefix or in finding nearest service of certain types, like DNS.

Anycast addresses can only be assigned to routers and they cannot be source address of an IPv6 packet.

- IPv6 has provision of very big multicast range. All addresses in FF00::/8 are multicast addresses. IPv6 multicast is very similar to IPv4 multicast as nodes needs to join multicast groups, send periodic membership messages, etc. In IPv6 instead of separate protocol like IGMP, ICMPv6 itself has provision for multicast related messages.

  Example multicast addresses are:

  - **FF02::1** can be used to communicate with all nodes on link. This can be treated as replacement of broadcast in IPv6.
  - **FF02::2** can be used to communicate with all routers on link. This can be used to find all routers on link so that we can later communicate with them about prefixes etc. or to just check if there is some IPv6 router on link.s

- 3^{rd} and 4^{th} hexadecimal characters of ulticast addresses (ff00::/8) are used for flags and scope respectively and have special significance. Meaning of flags is complex and I am not very sure about its usage yet. Hence flags is not described in this document.

  Scope is used to denote the scope/range of multicast address, that is, how far administratively can this multicast group members be from one another. Various popular values of scope field are:

  0 : Reserved

  1 : Node local

  2 : Link local

  3 : Admin local

  5 : Site local

  8 : Organizational local

  E : Global

  F : Reserved

  With above description in mind the following addresses must make more sense:

ff02::1 : All nodes on the link

ff02::2 : All routers on the link

ff05::2 : All routers on the site

- 3ffe::/16 was assigned to 6-bone for testing. It should not be routed over Internet.
- 6-to-4 tunnels use protocol 41 on IPv4 side and addresses in range 2002::/16 on IPv6 side. These can be blocked on firewall to prevent 6-to-4 tunneling.
- 2001:db8::/32 address range is reserved for authors to for examples within books. This is same as domains example.com, example.net, etc. being reserved and not assigned to anyone. These addresses should also never get routed on Internet.
- 16 multicast addresses - FF0[0-F]:: - are reserved. We should not use these 16 multicast addresses for normal applications.
- **Solicited node multicast address** : Solicited node multicast address can be formed by taking lower 24 bits of IPv6 address and appending it to prefix

  ff02:0:0:0:0:1:ff

  A node is required to compute and support a solicited node multicast address for every unicast and anycast address it is assigned

2.3 Converting MAC address to IID

- 7^{th} and 8^{th} bit of MAC address have special significance. 7^{th} bit indicates whether MAC address is universal, that is assigned by some manufacturer to be globally unique. 8^{th} indicates whether MAC address belongs to individual machine or to a group of nodes.
- First 24-bits of MAC address are called OUI portion. These bits can be used to determine which manufacturer produced the NIC. The first 24-bit ranges are assigned to manufacturers by some global numbering authority.
- To convert 48-bit MAC address to 64-bit IID use following steps:

1. Seperate 48-bit MAC address into two parts each having 24 bits. First 24-bits in one group and last 24-bits in another. For example for address 00:24:56:ab:ed:2b we will get 00:24:56 and ab:ed:2b
2. Now insert sequence ff:fe in middle of two groups and create 64-bit sequence. In the previous example case it would mean 00:24:56 + ff:fe + ab:ed:2b making it 0024:56ff:feab:ed2b.
3. Now we have to flip the 7^{th} or Universal bit. So in case of above MAC address it would become 0224:56ff:feab:ed2b. This 64-bit sequence can be used as IID.

2.4 IPv6 Header

- Fixed length and less number of fields. This enables very fast routing and forwarding.
- Header format is

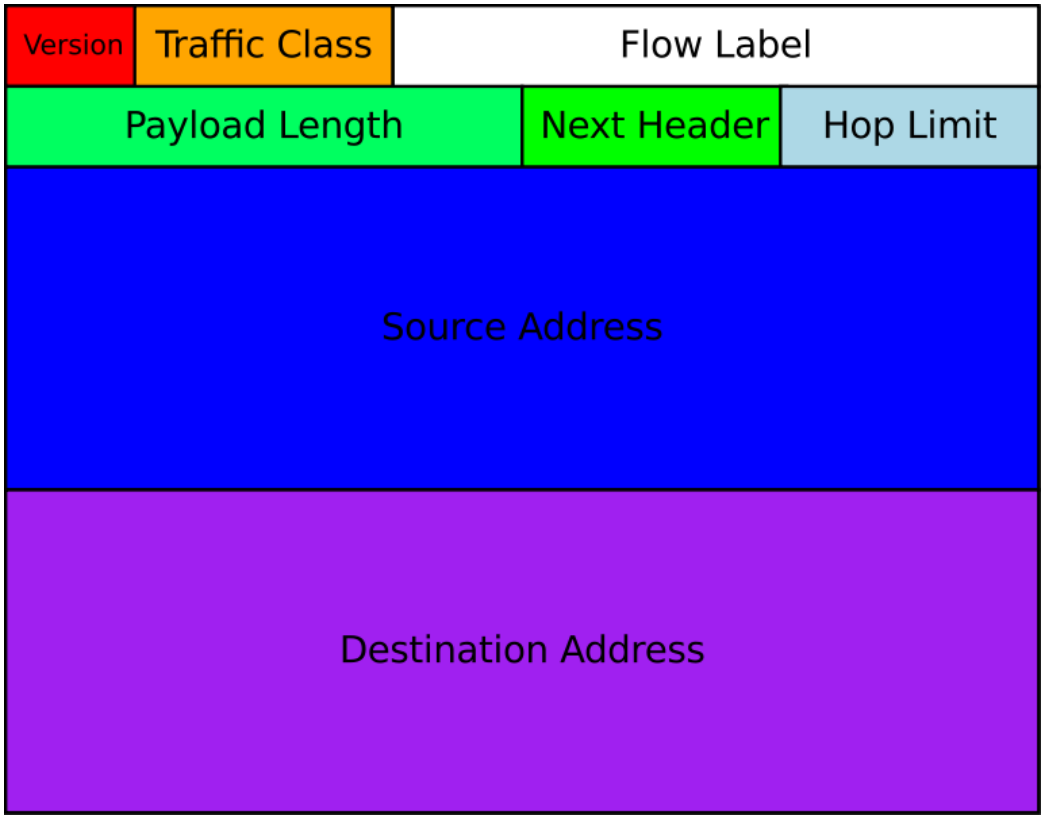

Figure 1: IPv6 header format

Here:

Version : It is a 4-bit field containing value 6 for IPv6.

Source address : It is 128-bit

Destination address : It is 128-bit

Flow label : It is used to recognize flows. Study about MPLS routing and MPLS labels to understand significance of flow label and how it can be used to make routing very fast.

- In IPv6 extension header chain is used for efficiency.
- Compared to IPv4 removed fields are:
  1. **Header checksum :** Rely on lower / higer layers for checksum, error detection / correction.
  2. **Header length :** No longer required, fixed header length

3. **Identification, Flags, Fragment offset :** Can now be done with optional fragmentation extension header.

- Renamed fields:

  1. Type of server → Traffic class
  2. Protocol type → Next header
  3. Total length → Payload length
  4. Time to live → Hop limit

- Added fields:

  1. **Flow label :** Can be optionally used to identify flows. This can allow IPv6 to support real time traffic like audio / video etc. more efficiently then IPv4 can. If particular router does not supports flows it can ignore flow value. If sender does not supports flow it can fill all zeros as flow, which indicates flow is not being used.

- Fragmentation is done and assembled only at end hosts. In IPv4 routers could fragment if MTU changed, but in IPv6 one must use path MTU sized packets or fragment information at source. This reduces load on routers as they do not have to spend time fragmenting packets.

- Traffic class is similar to type of service and can be used to provide differential services. We do not use this in most practical networks / Internet at present, but most switches / routers do support type of service based switching / routing.

- Next header can be TCP / UDP / ICMPv6 payload. It can also indicate optional next IPv6 header. Thus we have options (same as in IPv4) but without variable length of packet.

- IPv6 payload length is length of TCP / UDP etc. payload + length of all extension headers. Hence payload length includes everything except constant IPv6 header size.

- Hop limit allows removing of packet which otherwise would remain in network forever due to routing loops.

- Extension headers are 64-bit aligned, that is size of extension headers is multiple of 64-bit. There is no fixed number of extension headers.

- Sender should arrange headers in following order while sending:

1. Hop-by-hop options header
2. Destination options header – Destination options header placed here is meant for al destinations. This location is used when we use routing header.
3. Routing header (RH)
4. Fragment header
5. Authentication header (AH)
6. Destination options header – When destinations options header is placed after AH or ESP then it is meant only for final destination or end host and is not processed at intermediate nodes.
7. Encapsulating Security Payload (ESP) header
8. Upper layer header

- AH and ESP are security related headers. They must be supported by all hosts as part of mandatory IPSec in IPv6.
- Source node must arrange headers in recommended order. Destination nodes must be prepared to accept headers in any order.
- There is also mobility header for mobile IPv6.

2.5 Extension header details

2.5.1 Hop-by-hop options header

- NH = 0
- Since all nodes in network must process this header, it should be used only when necessary as it will have performance implications on routing.
- Two types of hop-by-hop options have been defined:
  - Jumbo payload
  - Router alert
- Jumbo payload can be used to send very large packets of size ¿ 65535 bytes. When jumbo payload is used payload length in IPv6 header is set to zero.
- Since hop-by-hop option header and destination option headers can contain multiple options they use:

TLV : Type Length Value

format. The same format is used by SNMP.

Type	Length	Value

- Two highest order bits of option type specify what should be done if options are unrecognizable at processing IPv6 node:

  00 : Skip over option and continue processing the packet

  01 : Discard the packet

  10 : Discard the packet and send an ICMP parameter problem message (Unrecognized option type) to the source

  11 : Discard the packet and send an ICMP parameter problem message (Unrecognized option type) to the source, only if destination is not multicast.

- Third highest bit of type sepcifies whether option data of that option can change en router to the packets final destination or not:

  0 : Option data does not change en route

  1 : Option data can change en route

- **Pad1 Option:** Used to occupy 1 byte. Can be used for 64-bit alignment.

0

- **PadN option:** Can occupy two or more bytes:

1	$n-2$	1 2 3 ... $n-2$

n byte padding

- Hop-by-hop option header packet format:

Next header	Header Extension Length
Options	

Here:

NH : Identifies next header in chain

Header Extenstion Length : Specifies size of hop-by-hop options header in multiple of 8 bytes. First 8 bytes are not included in this count. Hence a 64-bit hop-by-hop options header would have header extension length as zero.

- Jumbo payload option format:

194	4	Jumbo payload length

  Hence when using Jumbo payload a single packet can be of 2^{16} (65536) to $2^{32} - 1$ (4,294,967,295) bytes.

  Please note that like payload length this does not include IPv6 header length, but includes length of all optional headers including hop-by-hop option header.

2.5.2 Destination options header

- NH=60
- Can be after hop-by-hop options header or after any security header (AH or ESP). When destination options header is used after hop-by-hop options header and before routing header, it is processed at each destination in list of destinations in routing header.

  When it is present after security headers (AH or ESP) it is processed only by final destination node even if routing header is used.
- Mobile IP makes use of this header.
- Pad1 and PadN are defined for destination options same as they are defined for specifying options in hop-by-hop options header.
- Packet format is:

Next header	Header Extension Length
Options	

  Meaning of NH and Head. Extn. Length are same as before.

2.5.3 Routing header

- NH=43
- Routing header has field called Routing type and header format depends on this field.
- Packet format with routing type=0 is:

NH	Head. Extn. Length	Routing Type	Segments Left
Reserved			
Address [1]			
Address [2]			
⋮			
Address [N]			

- With use of extension headers who length can be upto 8 * 256 bytes, one can store upto 128 addresses of 16 bytes each. This is considerable improvement over IPv4 where in source based routing, loose source based routing, record route etc. only 9 IPv4 address (each 4 byte) could be stored.
- Consider example case where packet needs to be sent from source S, to destination D, via intermediate nodes IN1, IN2 and IN3 in same order. In this case values of various fields in IPv6 and Routing header are shown in below table:

Path	IPv6 header	Routing Header
From S to IN1	SA=S DA=IN1	HEL=6 SL=3 A[1]=IN2 A[2]=IN3 A[3]=D
From IN1 to IN2	SA=S DA=IN2	HEL=6 SL=2 A[1]=IN1 A[2]=IN3 A[3]=D
From IN2 to IN3	SA=S DA=IN3	HEL=6 SL=1 A[1]=IN1 A[2]=IN2 A[3]=D
From IN3 to D	SA=S DA=d	HEL=6 SL=0 A[1]=IN1 A[2]=IN2 A[3]=IN3

2.5.4 Fragment header

- NH=44
- Fragment header format:

NH	Reserved	Fragment offset	Res	m
Identification				

  Here:

  - Fragment offset : 13-bit value. Specifies where the current payload starts in unfragmented packet. It is specified in multiple of 8 bytes, same as in IPv4.
  - m : When 1 indicates more fragments are left. Is set to 0 on last fragment.

- Fragmentation in IPv6 must be done at source node. Fragments get assembled at destination same as in IPv4.

2.5.5 Authentication header

- NH=51
- Part of IPSec. AH provides authenticity and integrity with help of cryptograhically signed hash/checksums (Digital signatures).
- Details of AH and ESP are omitted as they require very good understanding of security related parameters like Security Parameter Index(SPI), secure sockets etc.

2.5.6 ESP header

- NH=50
- Encrypts all the following headers and data to provide confidentiality. This header is also part of IPSec

2.5.7 Upper layer headers

- TCP : NH=43
- UDP : NH=17
- ICMPv6 : NH=58
- These are followed by respective payloads. Note that TCP and UDP are same as in case of IPv4. They do not change because of change in network layer.

2.5.8 None or No next header

- NH=59
- In case IPv6 packet does not has upper layer header / payload then it can use this header to indicate end of header chain

2.5.9 Mobility header

- NH=135
- Used by mobile node, correspondent node and home agents.
- Details could be covered in theory lecture by prof. Shatrunjay Rawat

3 Lab work

As part of lab three different things are covered:

1. Adding and removing IPv6 addresses to/from interfaces. Seeing current addresses assigned, their lifetime, etc.
2. Static routing using IPv6. Looking and analyzing IPv6 routing table.
3. Creating IPv6 router to provide prefixes for nodes which want to configure themselves using auto-configuration.

The steps are mentioned in three different supporting pdf documents. Please follow following three pdf in given order as part of lab work:

1. `General_IPv6_node_configuration_and_commands.pdf`
2. `Basic_routing_using_IPv6.pdf`
3. `Configuring_Linux_machine_as_IPv6_router.pdf`