# IPv6 Essentials

**Thesis** · April 2018

DOI: 10.13140/RG.2.2.27781.04322



**1 author:**

Zainab Abdullah Jasim
University of Babylon
**2** PUBLICATIONS **0** CITATIONS

SEE PROFILE

# *IPv6 Essentials*

## 3.1 Introduction to IPv6

Internet Protocol version 6 (IPv6) is a new network layer protocol. It is an enhancement to Internet Protocol version 4 (IPv4), IPv4 was developed in the early 1970s for use in government and academic communities in the United States to facilitate communication and information sharing.

IPv6 is already gaining momentum globally, and is defined in the Request for Comments (RFC2460)[2]"Internet Protocol, Version 6 (IPv6) Specification" by Internet Engineering Task Force (IETF). The objective was to solve the address space limitations as well as provide additional functionality. IPv6 has more capabilities built into its foundation than IPv4.The IETF started the Internet Protocol Next Generation (IPng) work in 1993 to investigate different proposals and to make recommendations for further actions. The IETF recommended IPv6 in 1994. (The name IPv5 had previously been allocated to the experimental stream protocol.) Their recommendation is specified in [RFC 1752] , *"The Recommendation for IP Next Generation Protocol*". the basic function of the Internet Protocol is to move information across networks [41].

IPv6 was the major interest and activity in Europe and Asia, and there also is some traction in the United States. For example, in 2005 the U.S. Government Accountability Office (GAO) recommended that all agencies become proactive in planning a coherent transition to IPv6. All agency infrastructures had to be using IPv6 by June 30, 2008 (meaning that the network backbone was either operating a dual stack network core or it was operating in a pure IPv6 mode) [3]. IPv6 is expected to be the next step in

the industry evolution of the past 50 years from analog to digital to packet to broadband. IPv6 offers the potential of achieving increased scalability, reachability, end-to-end interworking, Quality of Service (QoS), and commercial-grade robustness for data communication ,mobile connectivity, and for Voice Over IP(VoIP)/networks. IPv6 has been enabled on many computing platforms, and many operating systems come with IPv6 enabled by default; IPv6-ready Operating Systems (OS) include but are not limited to: Mac OSX, OpenBSD, NetBSD, FreeBSD, Linux, Windows Vista, Windows XP (Service Pack 2), Windows 2003 Server, and Windows 2008 Server. Java began supporting IPv6 with J2SE 1.4 (in 2002) on Solaris and Linux. Other languages, such as C and C++ also support IPv6. Table 3.1 shows the core protocols that compose IPv6.

**Table 3.1 Key IPv6 protocols[43]**

| **protocol** | Description |
|---|---|
| Internet Protocol version 6 (IPv6): RFC 2460 | IPv6 is a connectionless datagram protocol used for routing packets between hosts. |
| Internet Control Message Protocol for IPv6 (ICMPv6): RFC 2463 | A mechanism that enables hosts and routers that use IPv6 communication to report errors and send status messages. |
| Multicast Listener Discovery (MLD): RFC 2710, RFC 3590, RFC 3810 | A mechanism that enables one to manage subnet multicast membership for IPv6. MLD uses a series of three ICMPv6 messages. MLD replaces the Internet Group Management Protocol (IGMP) v3 that is employed for IPv4. |
| Neighbor Discovery (ND): RFC 2461 | A mechanism that is used to manage node-to-node communication on a link. ND uses a series of five ICMPv6 messages. ND replaces Address Resolution Protocol (ARP), ICMPv4 Router Discovery, and the ICMPv4 Redirect message. ND is implemented using the Neighbor Discovery Protocol (NDP). |

In IPv6, interior/core routers do not perform packet fragmentation, but the fragmentation is performed end to end. That is, source and destination nodes perform, by means of the IPv6 stack, the fragmentation of a packet and the reassembly, respectively. The fragmentation process consists of dividing the source packet into smaller packets or fragments [3].

IPv6 is both simpler and more flexible than its IPv4 predecessor in number of enhancements and features in IPv6. Most significant is the vast amount of address space, along with support for orderly address assignment. Efficient network address aggregations on the Internet illustrated in Table 3.2 are some of the major differences between IPv4 and IPv6 followed by basic IPv6 terminology used later in this search. These differences can have implications for IPv6 security and are discussed throughout this.

Table 3.2 Differences between IPv4 and IPv6[41]

| Property | IPv4 | IPv6 |
|---|---|---|
| Address size and network size | 32 bits,<br>network size 8-30 bits | 128 bits,<br>network size 64 bits |
| Packet header size | 20-60 bytes | 40 byte |
| Header-level extension | limited number of small IP options | unlimited number of IPv6 extension headers |
| Fragmentation | sender or any intermediate router allowed to fragment | only sender may fragment |
| Control protocols | mixture of non-IP (ARP), ICMP, and other protocols | all control protocols based on ICMPv6 |
| Minimum allowed MTU | 68 bytes | 1280 bytes |
| Path MTU discovery | optional, not widely used | strongly recommended |
| Address assignment | usually one address per host | usually multiple addresses per interface |
| Address types | use of unicast, multicast, and broadcast address types | broadcast addressing no longer used, use of unicast, multicast and anycast address types |
| Address configuration | devices configured manually or with host configuration protocols like DHCP | devices configure themselves independently using stateless autoconfiguration or use DHCP |

**Basic Terminology**

The following basic IPv6 definitions are important for any IPv6 discussion.

- **Address**. An IPv6-layer identifier for an interface or a set of interfaces.
- **Node**. A device on the network that sends and receives IPv6 packets .
- **Deprecated address.** An address, assigned to an interface, whose use is discouraged, but not forbidden. A deprecated address should no longer be used as a source address in new communications, but packets sent from or to deprecated addresses are delivered as expected.
- **Router.** A node that sends and receives packets, and also accepts packets and forwards them on behalf of other nodes .
- **Host.** A node that may send and receive packets but does not forward packets for other nodes.
- **Link**. A communication facility or medium over which nodes can communicate at the link layer, i.e., the layer immediately below IPv6. Examples are Ethernets (simple or bridged); Point-to-Point Protocol (PPP); X.25, Frame Relay, or Asynchronous Transfer Mode (ATM) networks; and layer three (or higher) tunnels, such as tunnels over IPv4 or IPv6 itself.
- **Link MTU**. The maximum transmission unit (MTU), i.e., maximum packet size in octets, which can be conveyed over a link.
- **Path MTU**. The minimum link MTU of all the links in a path between a source node and a destination node. IPv4 mandates a minimum MTU of 68 octets. But; In IPv6, the minimum MTU is 1280 octets (see Table 3.3). 68 octets is very small, since most current link layer technologies have a minimum MTU of 1500.Since routers do not fragment IPv6 datagrams, if a datagram is larger than the size of one link in the path to the destination, the router connected to that link and receiving the datagram sends an ICMP error message to the

**Table 3.3 MTU for IPv4 and IPv6**

| | IPv4 | IPv6 |
|---|---|---|
| Minimum MTU | 68 | 1280 |
| Most efficient MTU | 576 | 1500 |

source and the datagram is dropped. IPv6 nodes use **P**ath **MTU** (PMTU) discovery to discover the right MTU to use for this destination. Path MTU discovery is not new for IPv6 [**RFC 1981**]. It is defined for IPv4 [**RFC 1191**], but has been rarely used for IPv4.

**Upper Layer.** A protocol layer is immediately above IPv6. As a new version of IP, IPv6 strictly changes the IP but it should not change the upper layer protocol. Examples are transport protocols such as **T**ransmission **C**ontrol **P**rotocol (TCP) and **U**ser **D**atagram **P**rotocol (UDP), control protocols such as **I**nternet **M**essage **C**ontrol **P**rotocol (ICMP), routing protocols such as **O**pen **S**hortest **P**ath **F**irst (OSPF), and internet or lower-layer protocols being *tunneled* over (i.e., encapsulated in) IPv6 such as **I**nternetwork **P**acket **E**xchange (IPX), AppleTalk, or IPv6 itself.

**Interface**. The point at which a node connects to a link. Unicast IPv6 addresses are always associated with interfaces.

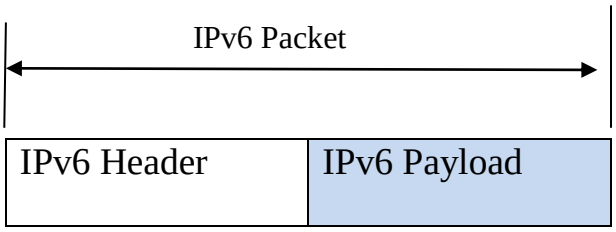


**Figure 3.1 IPv6 packet**

**Packet**. The basic unit of encapsulation, is passed across the interface between the network layer and the data link layer. An IPv6 packet contains header plus payload, as shown in Figure 3.1.

**Neighbors**. Nodes attached to the same link.

## 3.1.1 IPv4 Limitations .

IPv4 was designed over 25 years ago for a relatively small number of users. As a result of growing Internet use and growth of personal computing technologies" including IP networking", IPv4's IP address capacity could not meet the demand. In practice, the supply of available IPv4 addresses has been limited since the early 1990s.

Technologies widely adopted in response to the constrained supply of IPv4 addresses are **N**etwork **A**ddress **T**ranslation (NAT) as defined in [ RFC 3022]. NAT allows multiple computers that use private IPv4 address on a private network to share a single public IPv4 address and communicate with the Internet. NAT is typically implemented in routers.

A number of protocols cannot travel through NAT device, and hence the use of NAT implies that many applications (e.g., VoIP) cannot be used effectively in all instances. As a consequence, these applications can only be used in intranets**.**

***Examples include the following***:

**1-** Multimedia applications such as videoconferencing, VoIP, or video-on-demand/ IPTV do not work smoothly through NAT devices. Multimedia applications make use of **R**eal-Time **T**ransport **P**rotocol (RTP) and **R**eal-**T**ime **C**ontrol **P**rotocol (RTCP). These in turn use UDP with dynamic allocation of ports, and NAT does not directly support this environment.

**2**-Kerberos authentication needs the source address, and the source address in the IP header is often modified by NAT devices.

**3**-IPSec is used extensively for data authentication, integrity, and confidentiality. However, when NAT is used, there is an impact on IPSec because NAT changes the address in the IP header.

**4-**Multicast, although possible in theory, requires complex configuration in a NAT environment and hence in practice is not utilized as often as could the case be [43].

The need for obligatory use of NAT disappears with IPv6.

### 3.1.2 IPv6 Benefits

IPv6 adds improvements in areas such as routing and network autoconfiguration. Specifically, new devices that connect to Internet will be plug-and-play devices. With IPv6, one is not required to configure dynamic no published local IP addresses, the gateway address, the subnetwork mask, or any other parameters. The equipment, when plugged into the network, automatically obtains all requisite configuration data .

*The advantages of IPv6 can be summarized as follows:*

**1-Scalability:** IPv6 has 128-bit addresses versus 32-bit IPv4 addresses. with IPv4, IP address is $2^{32}$or ($3.4 \times 10^{10}$). IPv6 offers a $2^{128}$ or $4.3 \times 10^{39}$, unique node addressees.

**2-Security:** IPv6 includes security features, such as payload encryption and authentication of the source of the communication in its specifications.

**3-Real-time applications**: To provide better support for real-time traffic (e.g., VoIP), IPv6 includes "labeled flows" in its specifications. By means of this mechanism, routers can recognize the end-to-end flow to which transmitted packets belong. This is similar to the service offered by **M**ulti-**P**rotocol **L**abel **S**witching (MPLS), but it is intrinsic with the IP Mechanism.

**4-Plug-and-play:** IPv6 includes a plug-and-play mechanism that facilitates the connection of equipment to the network. The required configuration is automatic.

**5-Mobility :** IPv6 includes more efficient and enhanced mobility mechanisms, particularly important for mobile networks.

**6-Optimized protocol**: IPv6 embodies IPv4 best practices but remove unused or obsolete IPv4 characteristics. This results in a better optimized Internet Protocol.

**7-Addressing and routing:** simplified header and hierarchal addressing structure of IPv6 improved the addressing and routing hierarchy, routing information from a source to destination.

**8-Extensibility:** IPv6 has been designed to be extensible and offers support for new options and extensions.

**9- Merging** two IPv4 networks with overlapping addresses (say, if two organizations merge) is complex; it will be much easier to merge networks with IPv6 [43] [41].

## 3.2 IPv6 Addressing

The RFC 4291,"IPv6 Addressing Architecture specification", defines the address scope that can be used in an IPv6 implementation and the various configuration architecture for network designers of the IPv6 address space. Two advantages of IPv6 are that support for multicast is essential, and nodes can create link-local addresses during initialization.

IPv6 addressing differs from IPv4 in several ways aside from the address size. First, addresses specifically belong to interfaces, not to nodes. Interfaces often have multiple address. Second, IPv6 addresses consist of a network prefix in the higher order bits and an interface identifier in the lower order bits. Moreover, the *prefix indicates a subnet or link within a site*, and a link can be assigned multiple subnet IDs.

**Addressing**: refers to how end hosts become assigned IP addresses and how subnet works of IP host addresses are divided and grouped together.

### 3.2.1 IPv6 Address Types

IPv6 uses the notion of address types for different situations. These different address types are defined below:

**Unicast Addresses**. Addresses that identify one interface on a single node; a packet with unicast destination address is delivered to that interface. Figure 3.2 depicts a Unicast Address.

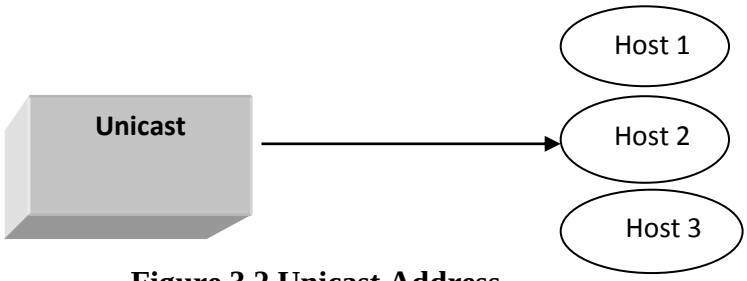


**Figure 3.2 Unicast Address**

**Anycast Addresses**. They are addresses that can identify several interfaces on one or more nodes; a packet with an anycast destination address is delivered to one of the interfaces bearing the address, usually the closest one as determined by routing protocols. Anycast addressing was introduced as an add-on for IPv4, but it was designed as a basic component of IPv6.The format of anycast addresses is identical from unicast addresses.

| *n bites* | *128-n bits* |
|---|---|
| Subnet Prefix | 00000000000000 |

The *subnet prefix* in an anycast address is the prefix that identifies a specific link. Anycast addresses may not be used as source addresses and may only be assigned to routers. It should be noted that there are no defined mechanisms for security or registration for anycast, nor is there a way to verify that a response to a packet sent to an anycast address was sent by an interface authorized to do so. Figure 3.3 depicts the Anycast Address.

**Multicast Addresses**. RFC 4291 defines a multicast address as an identifier for a set of interfaces (typically belonging to different nodes). A packet sent to a multicast address is delivered to all interfaces identified by that address. Although multicast addresses are common in both IPv4 and

IPv6, IPv6 multicasting has new applications.

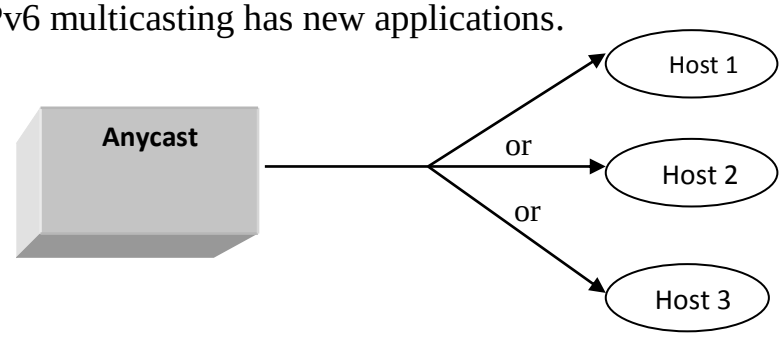


Figure 3.3 Anycast Address

The single most important aspect of multicast addressing under IPv6 is that it enables fundamental IPv6 functionality, including Neighbor Discovery (ND) and router discovery. Multicast addresses begin with **FF00::/8**. They are planned for efficient one-to-many and many-to-many communication, as shown in Figure 3.4.

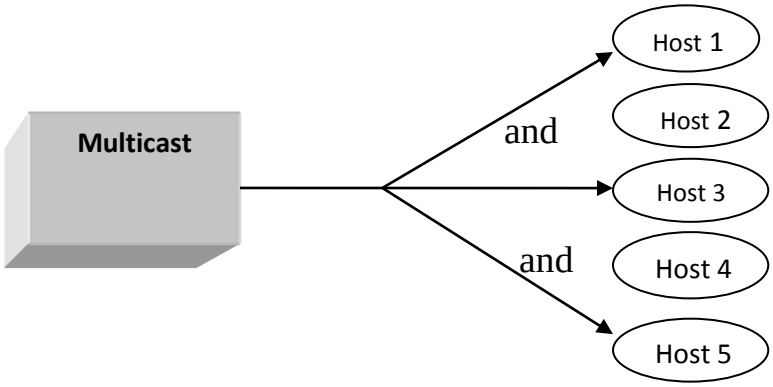


Figure 3.4 Multicast Address

**Broadcast Addresses**. Broadcast addressing is a common attribute of IPv4, but is not implemented in IPv6. Multicast addressing in IPv6 meets the requirements that broadcast addressing formerly fulfills [41].

### 3.2 .2 IPv6 Addressing Representation.

IPv6 addresses are 128 bits long and are written in colon-delimited hexadecimal address comprising of eight distinct numbers representing 16 bits each and written in base-16 (*hex*) notation. The valid hex digits are 0 - 9 and A - F and together with the colon separator are the only characters that can be used for writing an IPv6 address. An example of an IPv6

address is:

**7f87:**43e3**:9095:**02e5**:0216:**cbff**:feb2:**7474

**Note** that the address contains eight distinct four-place hex values, separated by colons. Each of these values represents 16 bits. IPv6 addresses are divided among three portions of the address.

-The ***network prefix*** is the high-order bits of an IP address, used to identify a specific network and, in some cases, a specific type of address.

-The ***subnet identifier***, which identifies a link within a site, is assigned by the local administrator of the site; a single site can have multiple subnet IDs. This is used as a designator for the network .

-The ***host identifier*** (host ID) of the address is a unique identifier for the node within the network upon which it resides. It is identified with a specific interface of the host. Figure 3-5 depicts the IPv6 address format.

[RFC 4291] also describes the notation for prefixes. There is no subnet mask in IPv6, although the slash notation used to identify the network address bits is similar to IPv4's subnet mask notation, but not equivalent.

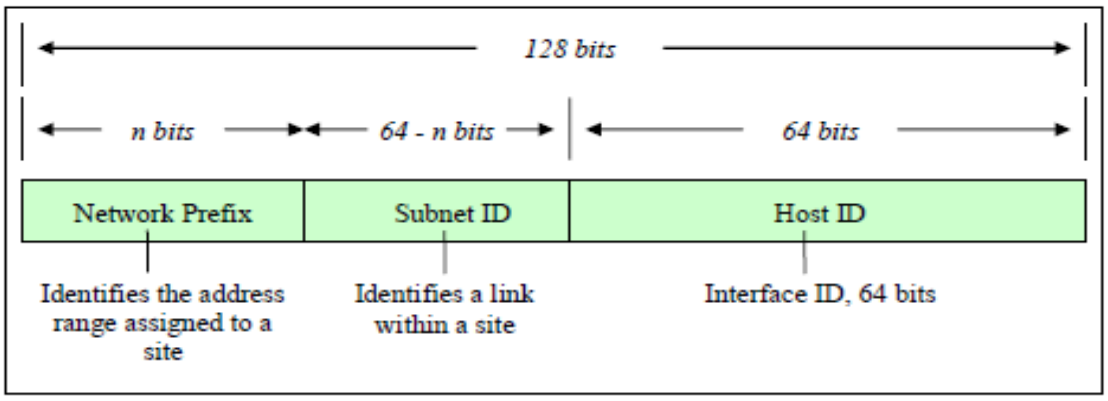


Figure 3.5 IPv6 Address Format

The IPv6 notation appends the prefix length and is written as a number of bits with a slash, which leads to the following format :

IPv6address/prefix length

The prefix length specifies how many of the address's left-most bits comprise the network prefix. An example address with a 32-bit network

prefix is: **7f87:43e3**:9095:02e5:0216:cbff:feb2:7474/**32 bits**

the first two groupings of hex values comprise the network prefix for the assignee of the addresses. The remaining 96 bits administrator is primarily for reallocation of the subnet ID and the host ID as shown in figure 3.6.

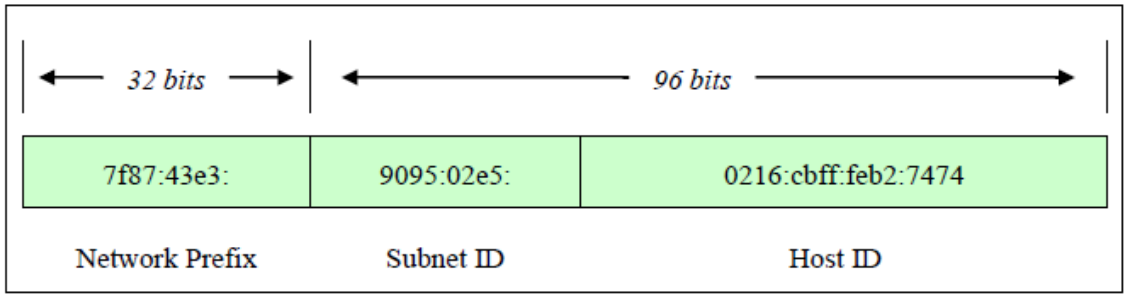


Figure 3.6 32-Bit Network Prefix

Figure 3-7 depicts address with Government, educational, commercial, and other networks with a network prefix of 48 bits (**/48**).

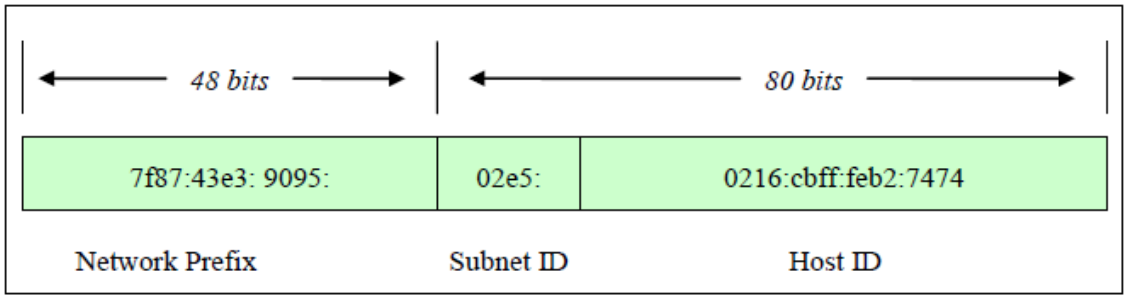


Figure 3.7 48-Bit Network Prefix

Subnets within an organization often have network prefixes of 64 bits (/64), leaving 64 bits for allocation to hosts' interfaces. The host ID should use a 64-bit interface identifier. as shown in Figure 3.8.

### 3.2.3 Shorthand for Writing IPv6 Addresses .

The notation for IPv6 addresses may be compressed and simplified under specific circumstances. This simplifies the address and makes it easier to

read , write and to remember [43] .

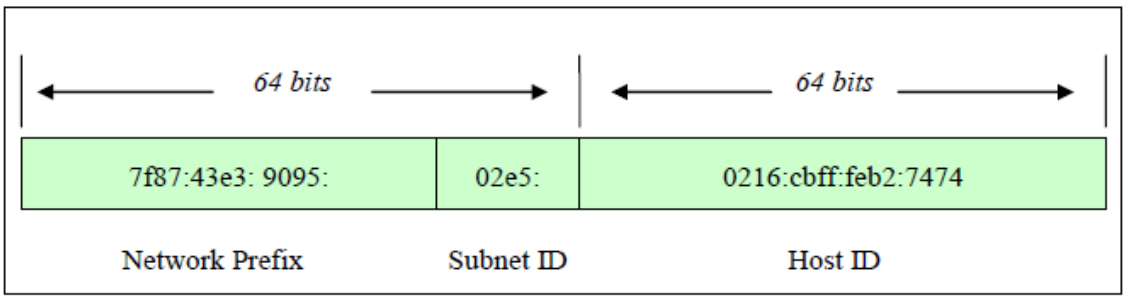


Figure 3.8 64-Bit Network Prefix

Example 1:

**7f22:065f:0aba**:02e5:**0000**:0ee9:**0000**:0444/**48** bits becomes

7f22:65f:aba:2e5**:0:**ee9**:0:**444/**48**

It is important to note that trailing zeroes may *not* be dropped, because they have essential place value in the address format.

**Example 2:**

**7f22:065f:0055**:**0000**:cd23:**0000**:**0000**:**0**205/**48** bits becomes

7f22:65f:55**:0:**cd23**::**205/**48**

The example address could be written:

7f22:65f:55**::**cd23**:0:0:**205/**48**, **but** this is not as efficient as

7f22:65f:55**:0:**cd23**::**205/**48**.

It is important to note that both of the formatted, but the latter address are shorter.

**Note:** Compression is just a convention for writing addresses, it does not affect how an address is used.

**Example 3**: 3223:0ba0::1234 that mean in fact

3223:0ba0:0000:0000:0000:0000:0000:1234

## 3.3 IPv6 Packet Format

The IPv6 packet is composed of three main parts: fixed header, optional extension headers and the payload.

### 3.3.1 IPv6 Fixed Header

Two primary design goals for the new header are efficiency and extendibility . The IPv6 header is 40 bytes long and contains only 8 fields, whereas IPv4 headers may be as short as 20 bytes or as long as 60 bytes and contain at least 14 different fields (some of which may be unused) Figure3.9 depicts the IPv4 fields.When comparing these attributes, it becomes apparent that the IPv6 header is simpler and more efficient to process. *Three examples are:*

 The checksum has been removed, because error checking is usually performed in link layer and transport layer protocols.

 Fragmentation has been relegated to an extension header, the minimum MTU has been increased to 1280 bytes, and fragmentation and reassembly are only performed by endpoints.

 Routers have to examine more than the 40-byte header only when the Next Header (NH) field is zero.

The fixed header makes up the first 40 octets (320 bits) of an IPv6 data packet. The format of the fixed header is presented in Figure 3.10. The fixed length of the IPv6 header does not preclude flexibility function. Options are handled with extension headers. The following are the 8 fields in the fixed IPv6 header.

**Ver/IP version (4-bits)**: The 4-bit version field contains the number 6. It indicates the version of the IPv6 protocol. This field is the same size as the IPv4 version field that contains number 4. This field has a limited use .

**Traffic class/ Packet priority(8 bits):** The 8-bit Priority field can assume different values to enable the source node to differentiate between the packets generated by it by associating different delivery priorities to them.

**Flow Label/QoS management (20 bits):** The 20-bit is used by a source to label a set of packets belonging to the same flow. A flow is uniquely

identified by the mixture of the source address and of a non-zero Flow label. This new capability is added to enable the labeling of packets belonging to particular traffic "flows" for which the sender requests special handling, such as nondefault quality of service or "real-time" service.

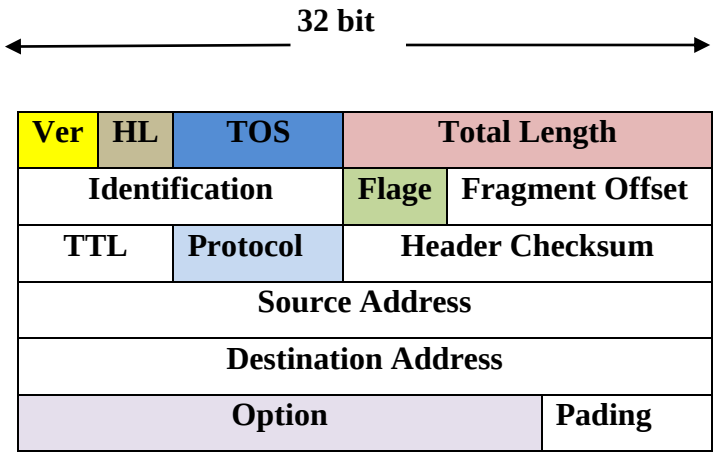


**figure 3. 9 IPv4 Header**

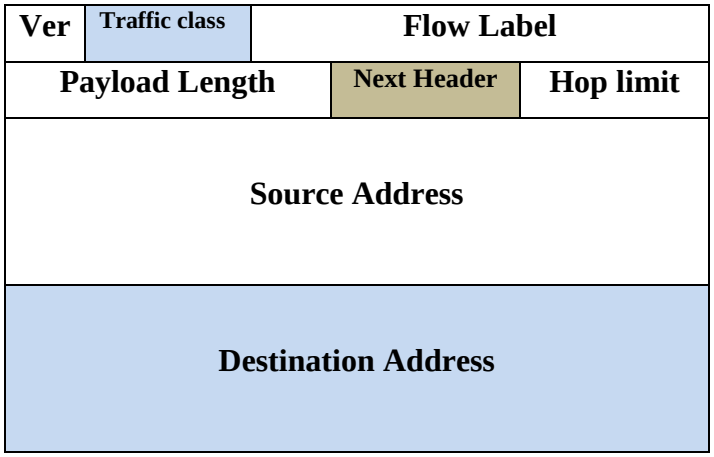


**Figure 3.10 IPv6 Header(packet header format)**

**Payload length in bytes(16 bits)** The 16-bit contains the length of the data field in octets/bits following the IPv6 packet header. This field puts an upper limit on the maximum packet payload to 64 kilobytes.

**Next Header (8 bits)** Identifies the type of header immediately following the IPv6 header in distinct order and is located at the beginning of the data field (payload) of the IPv6 packet. This field usually specifies the transport

layer protocol used by a packet's payload. The two most common kinds of Next Headers are TCP (6) and UDP (17).

**Hop Limit/**Time To Live (TTL). The main function of this field is to identify and to discard packets that are stuck in an unclear loop due to any routing information errors. The 8-bit field also puts an upper limit on the maximum number of links between two IPv6 nodes.

**Source address (128 bits)** The 128-bit source address field contains the IPv6 address of the originating node of the packet. It is the address of the originator of the IPv6 packet.

**Destination address (128 bits)** The 128-bit contains the destination address of the recipient node of the IPv6 packet. It is the address of the intended recipient of the IPv6 packet.

## 3.3.2 Extension Header.

In IPv6, optional Internet-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper-layer header in a packet. There is a small number of such extension headers, each identified by a distinct Next Header value [3], as illustrated in the examples of Figure 3.11 .

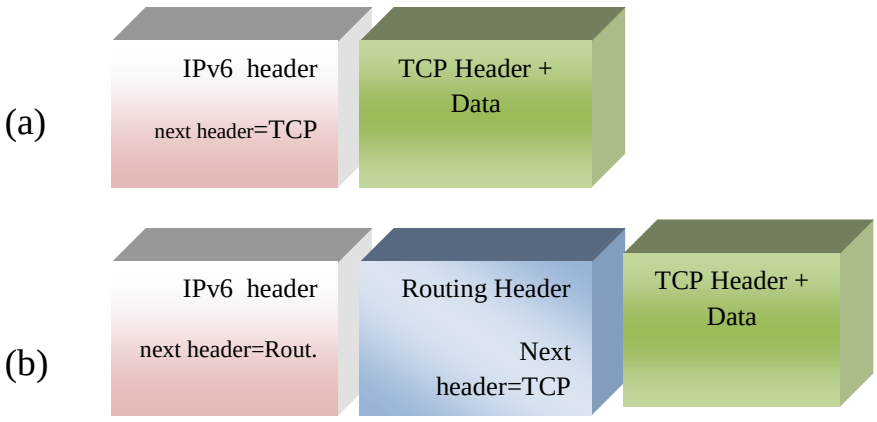


figure 3.11 (a) packet without extension header

(b) with extension header(Routing)

An IPv6 packet may carry zero, one, or more extension headers, each identified by the Next Header field of the preceding header.

**Note**: If the value "59" appears in the field" next header", that means this header is the last one.

### 3.3.3 Payload

The payload can have a size of up to 64 KB in standard mode, or larger with a "jumbo payload" option in a *Hop-By-Hop Options* extension header , and hosts are expected to use Path MTU discovery.

## 3.4 IPv6 Security

Computers, networks, operating systems, applications, users, policies and protocols are components forming a complex system, where they interact with each other. Security should be applied to each component separately, as well as to all components as a system. Perfect security does not exist: security is about managing risks.

Figure 3.12 identifies simply where some of the security solutions fit in the networking stack(TCP/IP).

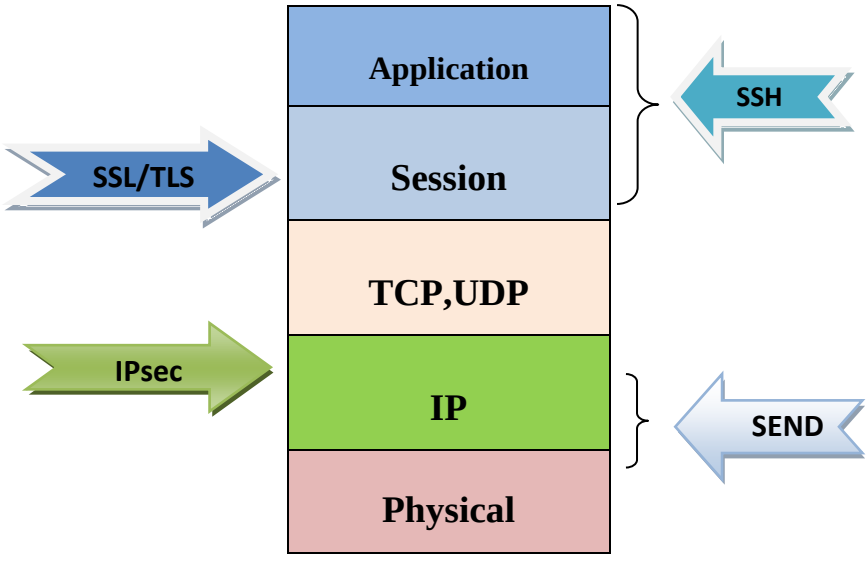


Figure 3.12 TCP/IP layers and security

IP security (IPsec) works at the IP layer, SSL/TLS at the session layer, SSH at the session/application layer and Newer one is Secure Neighbor Discovery (ND) , or sometimes called SEND, at the frontier between IP and the link-layer (Physical) [45].

### 3.4.1 Security Mechanisms

This section identifies security mechanisms that can be employed in Internet Protocol version 6 (IPv6) environments. Many of these mechanisms are imported from the IPv4 world to identify some of the commonly available techniques, methods, and protocols that can be used to secure IPv6 networks. *We view security in four realms*:

(1) confidentiality and integrity of information while in transit (in a network);

(2) perimeter/access security in reference to a defined set of private information technology (IT) assets;

(3) system security/integrity (including both client and server systems); and,

(4) security of data at rest (namely, security of database/storage systems).

However, this section only deals with the first item. This kind of security can be seen as achievable using tunnels that carry encrypted information.

Confidentiality and integrity of information while in Transit:- the extent of possibilities related to confidentiality and integrity of information while in transit, the encryption functions within the protocol stack of a communicating node.

1- military applications often use link-level encryption at the physical layer (channel).
2- commercial applications use encryption (tunneling) at the network layer (via IPsec tunneling discussed below); Or,

3- at the transport layer (via Transport Layer Security (TLS)/Secure Sockets Layer (SSL) tunneling).

As noted, the confidentiality and integrity of information while in transit (in an IPv6 network) is achieved by using encryption (tunneling) at the network layer via IPsec tunneling or at the transport layer via TLS/SSL tunneling.

The tunnel is generally established between end-system nodes (e.g., between a client accessing a remote host) or between network-edge nodes (e.g., wide area network router in the intranet to wide area network router at the far end of the intranet).

The resulting encrypted/tunnel arrangement is called a virtual private network (VPN), particularly when the service provider handles the encryption or when the Internet is used (and the end-user organization does the router-to-router encryption).

**VPN** (Virtual Private Network) is a generic term referring to the use of public or private networks to create groups of users separated from other network users and may communicate among them as if they were on a private network [RFC2764], by Encryption Mechanisms.

Encryption is the cryptographic transformation of data to produce ciphertext and prevent it from being known. If the transformation is reversible, the corresponding reversal process is called *decryption*, which is a transformation that restores encrypted data to its original state. Usually, the plaintext input to an encryption operation is cleartext (in some cases, the plaintext may be ciphertext that was output from another encryption operation). Encryption and decryption involve a mathematical algorithm for transforming data. In addition to the data to be transformed, the algorithm has one or more inputs that are control parameters: a key value that varies with the transformation and, in some cases [43], an

Initialization value that establishes the starting state of the algorithm [RFC2828].

### 3.4.2 IPsec in IPv6

In the early 1990s, the IETF began to view the lack of IP-level security as a serious drawback, and it started developing a collection of network layer security protocols known as IPsec to be used specifically to secure IP communications. The IETF has published three versions of IPsec, which now provide strong, up-to-date confidentiality and integrity protection, access control, replay detection, key management, and strong peer-entity authentication for IPv4. IPsec has proved difficult to deploy with IPv4, and its widespread use has been limited to protecting certain (VPNs) and for secure remote access to enterprise networks when strong security is a requirement. One of the reasons for this has been the constrained availability of IPv4 addresses[3].

The use of Network Address Translation (NAT) at many edge or customer premise routers breaks the end-to-end model by using *non-routable addresses* at end systems. IPsec was designed long after IPv4. Today, most operating systems, routers, and security appliances bundle or integrate IPsec with their IPv4 protocol stacks, but historically, IPsec was implemented separately from IPv4. This is in contrast to IPv6, for which IPsec is an integral part of the specification. IPv6 does not have the addressing limitations that inhibit end-to-end use of IPsec with IPv4. Also, IPsec has been recommended as the way to secure important features of IPv6 such as OSPFv3 routing, mobility, and even neighbor discovery.

IPsec offers three primary models for protection, as follows:

 **Gateway-to-gateway.** This model protects communications between two specific networks, such as an organization's main office network and a branch office network, or two business partners' networks.

 **Host-to-gateway.** This model protects communications between one or more individual hosts and a specific network belonging to an organization. The host-to-gateway model is most often used to allow hosts on unsecured

networks, such as traveling employees and telecommuters, to gain secure remote access to internal organizational services, such as the organization's email, Web servers, and custom applications.

 **Host-to-host.** A host-to-host architecture protects communication between two specific computers. It is most often used when a small number of users need to use or administer a remote system that requires network layer security for some or all of its higher layer protocols. OSPFv3 is an example. IPv6 offers increased opportunities to use this mode, because IPsec is a mandatory component for every IPv6 implementation, and end-to-end connectivity without Network Address Translation (NAT) makes using IPsec easier[46].

### 3.4.3 IPsec Mechanisms

IPsec as defined in [RFC2401] is a fundamental element of IPv6 security in the context of confidentiality and integrity of information while in transit IPsec is a protocol that provides for the support of encrypted payloads in IP networks; it includes encryption and authentication technologies. It is a broadly deployed mechanism (in IPv4 environments) used to support VPNs defined over the Internet.

The IPsec architecture as shown in Figure 3.13 .

(1) security protocols (Authentication Header (AH) and Encapsulating Security Payload (ESP));

(2) security associations SA(what they are, how they work, how they are managed, and associated processing),see Figure 3.14, that means; SA is a logical connection between two devices transferring data. An SA provides

data protection for unidirectional traffic by using the defined IPSec protocols. An IPSec tunnel typically consists of two unidirectional SAs, which together provide a protected, full-duplex data channel. The SAs allow an enterprise to control exactly what resources may communicate securely, according to security policy.

-One way relationship between a sender and a receiver.

-Determine IPsec processing, security services are not fixed generated and customized per traffic flows, that are provided to a user .

(3) key management (IPsec key exchange (IKE)); IPSec uses the Internet Key Exchange (IKE) protocol to assist and automate the SA setup and the exchange of keys between parties transferring data. Using keys ensures that only the sender and receiver of a message can access it. IPSec requires that keys be recreated, or refreshed, frequently so that the parties can communicate securely with each other. *IKE manages the process of refreshing keys;* however, a user can control the key strength and the refresh frequency. Refreshing keys on a regular basis ensures data confidentiality between sender and receiver; and,

(4) algorithms for authentication and encryption[44].

The set of security services includes access control service, connectionless data integrity service, data origin authentication service, protection against replays, data confidentiality service, and limited traffic flow confidentiality [RFC2828].

**AH** is designed to provide connectionless data integrity service and data origin authentication service for IP datagrams and (optionally) to provide protection against replay attacks. AH provides for integrity but without confidentiality. AH may be used alone, in combination with the IPsec ESP protocol, or in a nested fashion with tunneling. Security services can be provided between a pair of communicating hosts, between a pair of

communicating security gateways, or between a host and a gateway.

**A gateway** is a device that monitors and manages incoming and outgoing network traffic and routes the traffic accordingly. **A host** is a device that sends and receives network traffic.

**ESP** can provide the same security services as AH, and ESP can also provide data confidentiality service. ESP provides for confidentiality with optional integrity and authentication. The main difference between authentication services provided by ESP and AH is the extent of the coverage; ESP does not protect IP header fields unless they are encapsulated by AH [RFC2828]. IPsec treats everything in an IP datagram. Usually, an IP datagram has three consecutive parts: the IP header (for routing purpose only), the upper-layer protocol headers (e.g., the Transmission Control Protocol (TCP) header), and the user data (e.g., TCP data). In transport mode, an IPsec header (AH or ESP) is inserted after the IP header and before the upper-layer protocol header to protect the upper-layer protocols and user data. In tunnel mode, the entire IP datagram is encapsulated in a new IPsec packet (a new IP header followed by an AH or ESP header). In either mode, the upper-layer protocol headers and data in an IP datagram are protected as one indivisible unit. The keys used in IPsec encryption and authentication are shared only by the sender-side and receiver-side security gateways. All other nodes in the public Internet, whether they are legal routers or malicious eavesdroppers, see only the IP header and will not be able to decrypt the content or tamper with it without detection. Traditionally, the intermediate routers do only one thing forward packets based on the IP header (mainly the destination address field); IPsec's "end-to-end" protection model is well suited to this layering paradigm [42].

As implied above, a VPN is a restricted use, logical (i.e., artificial or

simulated) computer network that is constructed from the system resources of a relatively public, physical (i.e., real) network (such as the Internet), often by using encryption (located at hosts or gateways), and often by tunneling links of the virtual network across the real network [RFC2764]. For example, if a corporation has local area networks (LANs) at several different sites, each connected to the Internet by a firewall, the corporation could create a VPN by (1) using encrypted tunnels to connect from firewall to firewall across the Internet and (2) not allowing any other traffic through the firewalls. A VPN is generally less expensive to build and operate than a dedicated real network, because the virtual network shares the cost of system resources with other users of the real network [RFC2828]. Having noted that IPsec is fairly important in the context of network security (at least in terms of confidentiality and integrity).

### 3.4.4 Ipsec Protocols

There are several types of IP tunneling mechanisms and, depending on their form, they can provide some level of essential data security. IP tunneling mechanisms include IP/IP, generic routing encapsulation (GRE) tunnels, layer 2 tunneling protocol (L2TP), IPsec, and multiprotocol label switching (MPLS). Some of these protocols are not often thought of as tunneling protocols, but they are and they do provide some type of protection. IPsec is considered the best tunneling protocol for IP networks because it provides strong security services such as encryption, authentication, and key management.

IPsec provides security services at the IP layer by enabling a system to select required security protocols, determine the algorithm(s) to use for the service(s), and put in place any cryptographic keys required to provide the requested services [44].

IPsec is a suite of protocols that assist in protecting communications

over IP networks RFC4301," Security Architecture for the Internet Protocol" . IPsec protocols work together in various combinations to provide protection for communications. the three primary components.

the Encapsulating Security Payload (ESP) RFC4303, Authentication Header (AH) RFC4302, and Internet Key Exchange (IKE) protocols RFC 4306 [42] .

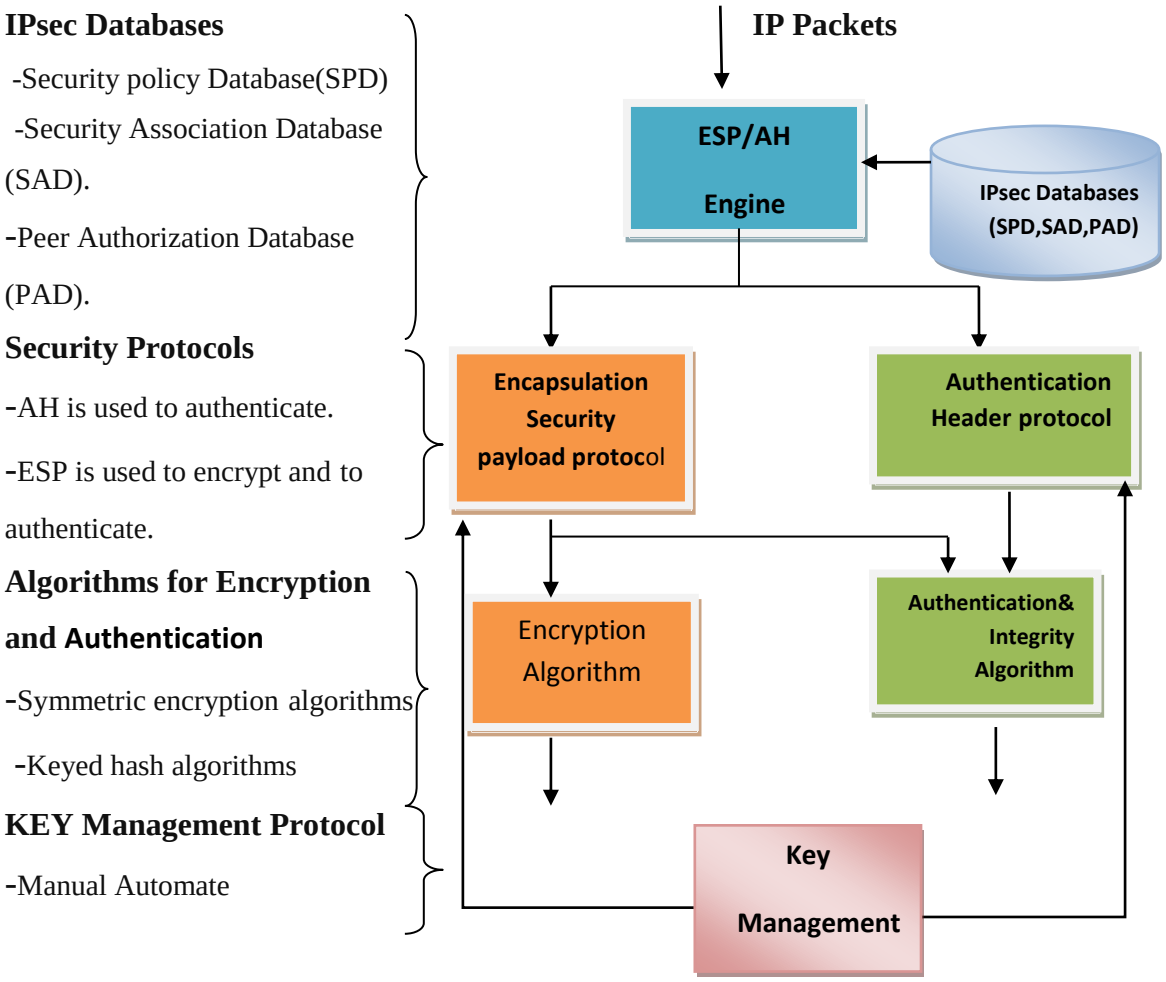


**figure 3.13 IPsec architecture**

Figure 3.14 shows the relationship of the IPsec protocols. Implementation of IPsec protocols is optional for IPv4, but it is mandatory for IPv6. **I**nternet **K**ey **E**xchange (IKE) and **I**nternet **S**ecurity **A**ssociation and **K**ey **M**anagement **P**rotocol (ISAKMP) RFC 2408,are application layer protocols that are used in combination with IPsec.

IPsec provides network security ,for IP Traffic only, specifically confidentiality and authentication. IPsec is an interoperable (open), reasonably high quality, cryptographically based security mechanism. It provides data origin authentication, connectionless integrity, confidentiality (encryption), replay detection (a form of partial sequence integrity), partial traffic flow confidentiality, and access control (via packet filtering). These capabilities are provided at the IP layer, offering protection for IP or upper layer protocol [3].

To protect data as it travels across a public or a closed IP network, IPsec supports a combination of the following *network security functions*:

(i) Data confidentiality: it encrypts packets before transmission;

(ii) Data integrity: it authenticates packets to help ensure that the data has not been altered during transmission;

(iii) Data origin authentication: it authenticates the source of received packets, in conjunction with data integrity service;

(iv) Anti-replay: it detects aged or duplicate packets, rejecting them to avoid replay attacks.

## 3.4.4 IPsec Modes

IPsec can be used in transport mode or in tunnel mode [RFC2828].

***Transport mode****:* The protection applies to (i.e., the IPsec protocol encapsulates ) the *packets of upper-layer protocols*, the ones that are carried above IP. A transport **mode** is an IPsec mode as defined in [RFC2401], (Security Architecture for the Internet Protocol) [RFC3884].

Transport mode secures portions of the existing IP header and the payload data of the packet and inserts an IPsec header between the IP header and the payload. In transport mode, IPsec inserts a security protocol header into outgoing IP packets between the original IP header and the packet payload.

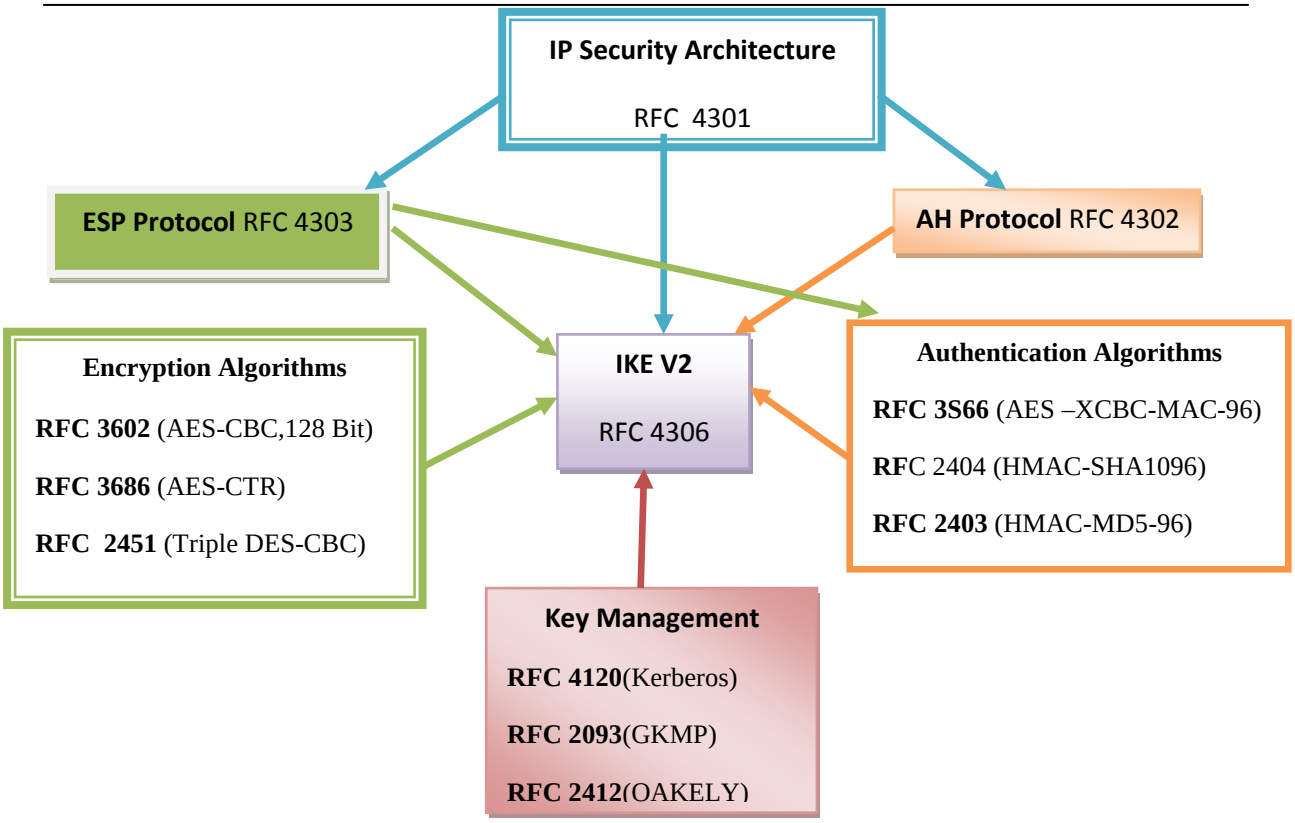


Figure 3.14 IPsec protocol[44]

The contents of the IPsec header are based on the result of a security association (SA) look up that uses the contents of the original packet header as well as its payload (especially transport layer headers) to locate an SA in the security association database (SAD).

When receiving packets secured with IPsec transport mode, a similar SA lookup occurs based on the IP and IPsec headers, followed by a verification step after IPsec processing that checks the contents of the packet and its payload against the relevant SA.

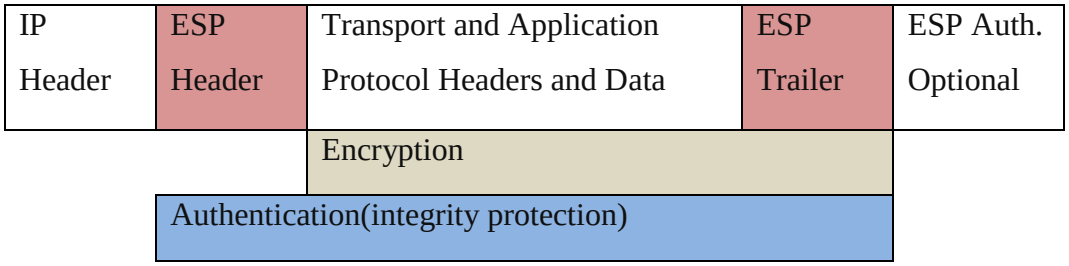


figure 3.15 ESP Transport mode packet

The verification step is similar to firewall processing [RFC3884]. Transport mode Security Association (SA) is always between two hosts only, see figure 3.16.

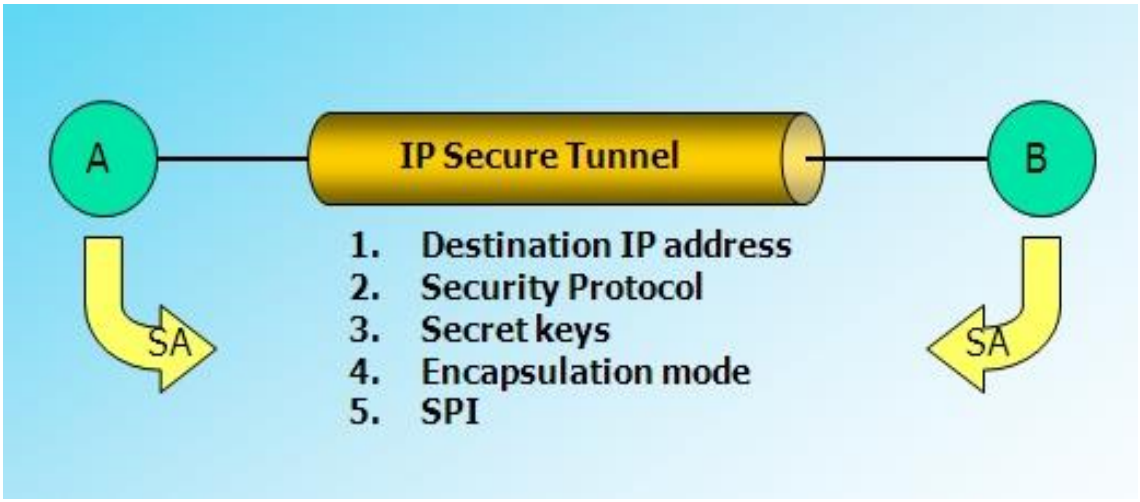


figure 3.16 Security Association parameter

***Tunnel mode:*** The protection applies to (i.e., the IPsec protocol encapsulates) *IP packets*. In a tunnel mode security association, each end may be either a host or a gateway. Whenever either end of an IPsec security association is a security gateway, the SA is required to be in tunnel mode. An additional set of origination/destination IP addresses is required in this mode and is an IPsec mode as defined in [RFC2401] Security Architecture for the Internet Protocol. Tunnel Mode is required when at least one of the endpoints is a security gateway (intermediate system that implements IPsec functionality, e.g., a router).

By contrast, transport mode is allowed between two end hosts only

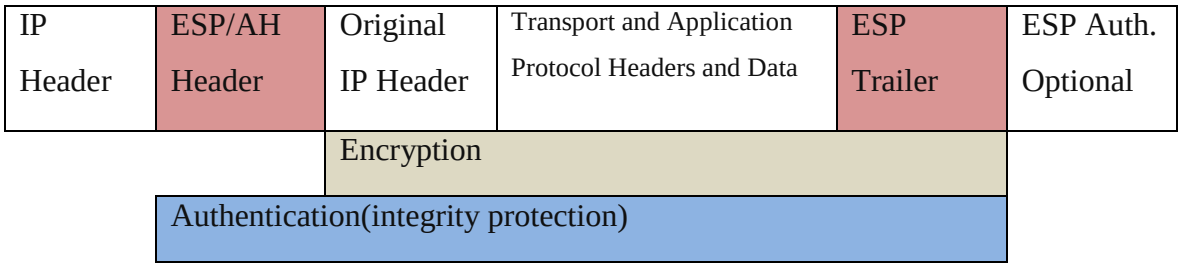


figure 3.17 ESP Tunnel mode packet

[RFC3884]. Tunnel mode adds an additional IP header before performing similar operations. When using tunnel mode, IPsec prepends an IPsec header and an additional IP header to the outgoing IP packet. In essence, the original packet becomes the payload of another IP packet, which IPsec then secures.

This has been described as "a tunnel mode, SA is essentially a (transport mode) SA is applied to an IP tunnel." In IPsec tunnel mode, the IP header of the original outbound packet together with its payload (especially transport headers) determines the IPsec SA, as in transport mode. However, a tunnel mode SA also contains encapsulation information, including the source and destination, IP addresses for the outer tunnel IP header, which is also based on the original outbound packet header and its payload [RFC3884].

The IPsec architecture has a security policy database that specifies which traffic is protected and how [42].

### 3.5 Transition Approaches and Mechanisms for IPv6.

Deployment of IPv6 occurs gradually. to begin with, IPv6 is to be deployed within isolated islands with interconnectivity among the islands achieved by the existing IPv4 infrastructure; a number of transition mechanisms have been defined to interconnect such islands.

There is an additional need for support for IPv6 hosts and routers that need to interoperate with inheritance IPv4 hosts. That [RFC2893] defines the following types of nodes with respect to the transition to IPv6:

**IPv4-only node**: Is a host or router that implements only IPv4. An IPv4-only node does not understand IPv6. The installed base of IPv4 hosts and routers is an example of IPv4-only nodes.

**IPv4/IPv6 node**: Is a host or router that implements both the IPv4 and IPv6 protocols.

**IPv6-only node**: Is a host or router that implements IPv6 and does not implement IPv4.

The [RFC 2893] also defines the IPv4-compatible IPv6 address e.g., ::156.55.23.5 which are used to implement a simple automatic tunneling mechanism[41].

The Internet Engineering Task Force (IETF) has defined a number of specific mechanisms to assist in transitioning to IPv6.

*Transition mechanisms fall into three categories.*

**-Dual Stack**: The principal building block for transitioning is the dual-stack approach. Dual-stack nodes maintain two protocol stacks that operate in parallel, as shown in Figure 3.18 and thus allow the end system or router to operate via either protocol. In end systems, they enable both IPv4-and IPv6-capable applications to operate on the same node.

The purpose of a dual stack method is to minimize the number of tunnels used in a transition for rapid deployment.

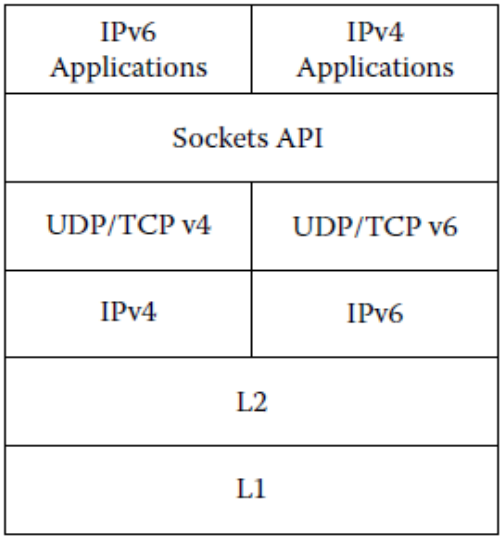


Figure 3.18 Dual-stack

Dual-stack capabilities in routers allow handling of both IPv4 and IPv6 packet types for more details [RFC 4213] . Dual Stack environment uses more resources than a single protocol environment, for example filtering must be implemented for both protocols, forward packets for both

protocols, that means extra address computation and run routing, and hosts must allocate resources to both protocol stacks [43][41].

**-Translation**: Translation refers to the direct conversion of protocols (e.g., between IPv4 and IPv6) and may include transformation of both the protocol header and the protocol payload, that mean transforming IPv4 or IPv6 packets into other protocol. Translation can occur at several layers in the protocol stack, including IP as shown in Figure 3.19, transport, and application layers.

Translation mechanisms (IPv4 to IPv6 and IPv6 to IPv4) introduce new methods to construct networks and systems and thus enlarge the set of possible attacks against those networks and systems.

**Note** that protocol translation can result in feature loss when there is no clear mapping between the features provided by translated protocols.

For instance, translation of an IPv6 header into an IPv4 header will lead to

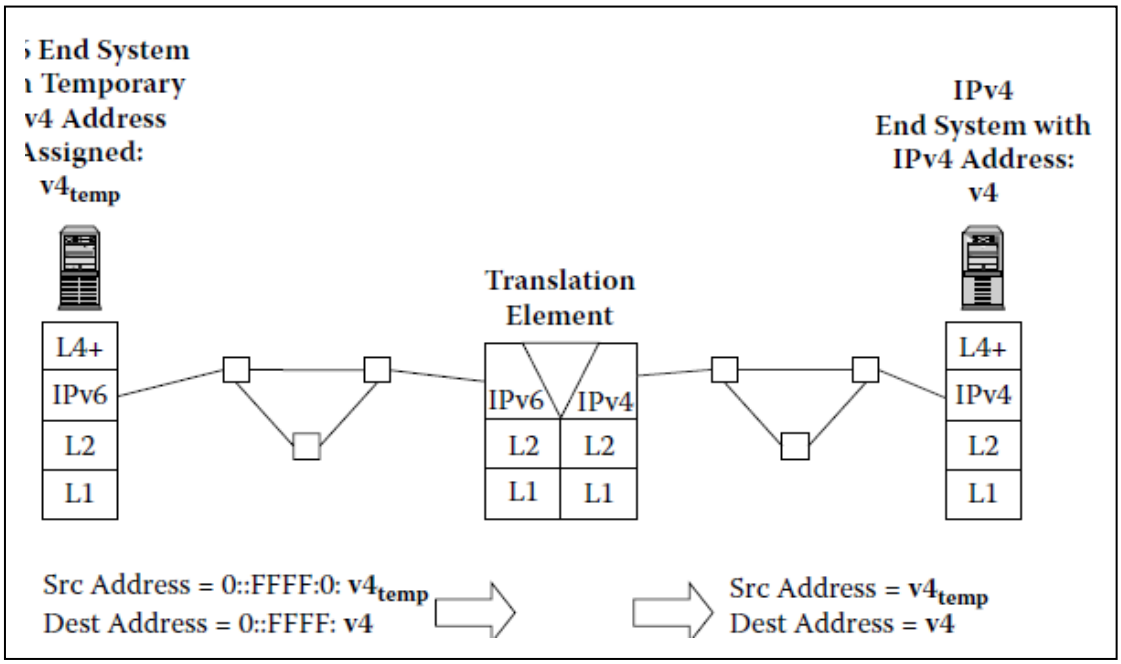


Figure 3.19 IP address translation, IPv6 to IPv4.

The loss of the IPv6 flow label and its accompanying functionality.

**Note** :Translation and dual stack mechanisms allow IPv6 hosts to use IPv4

resources.

**-Tunneling** (or encapsulation):Tunneling is used to interconnect compatible networking nodes or domains across incompatible networks as shown in Figure 3.21 . It can be viewed technically as the transfer of a payload protocol data unit by an encapsulating carrier protocol. For IPv6 transition, the IPv6 protocol data unit is generally carried as the payload of an IPv4 packet. Encapsulation of the payload protocol data unit is performed at the tunnel entrance, and decapsulation is performed at the tunnel exit point.

In other words tunneling is the encapsulation of one protocol inside another. Tunneling is a flexible transition mechanism and supports several scenarios including : as part of a dual stack IPv4/IPv6 transition strategy, as a separate transition method or used together with protocol translation.

**Note** that a transition mechanism may employ techniques from more than one of these categories. For example, when an end system or router creates an IPv6-in-IPv4 tunnel, this could be classified as both dual stack (having both an IPv4 and IPv6 address) and tunneling[41][34].

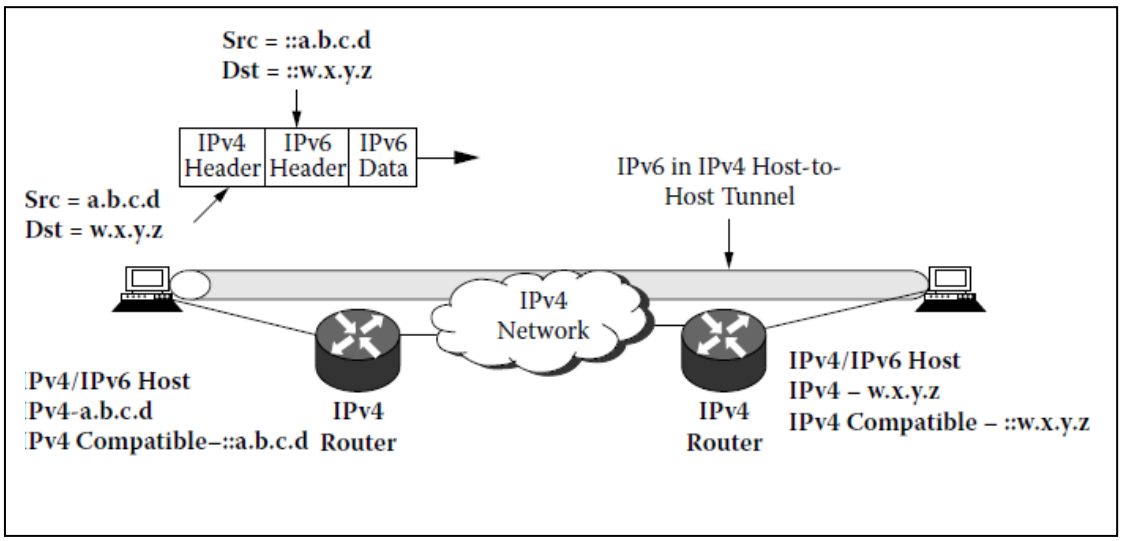


Figure 3.20 Automatic tunneling with IPv4-compatible IPv6 addresses

As shown in Figure 3.21, these routers are also set up to serve as tunnel endpoints. As tunnel endpoints, they encapsulate the IPv6 packet inside an

IPv4 packet and forward it to the other tunnel endpoint. Tunnel endpoints are always a focal point for security; attackers frequently target them in attacks. Even if the traffic going through the tunnel is protected with end-to-end IPsec, additional security controls such as authorization should be applied at the tunnel endpoints[41].

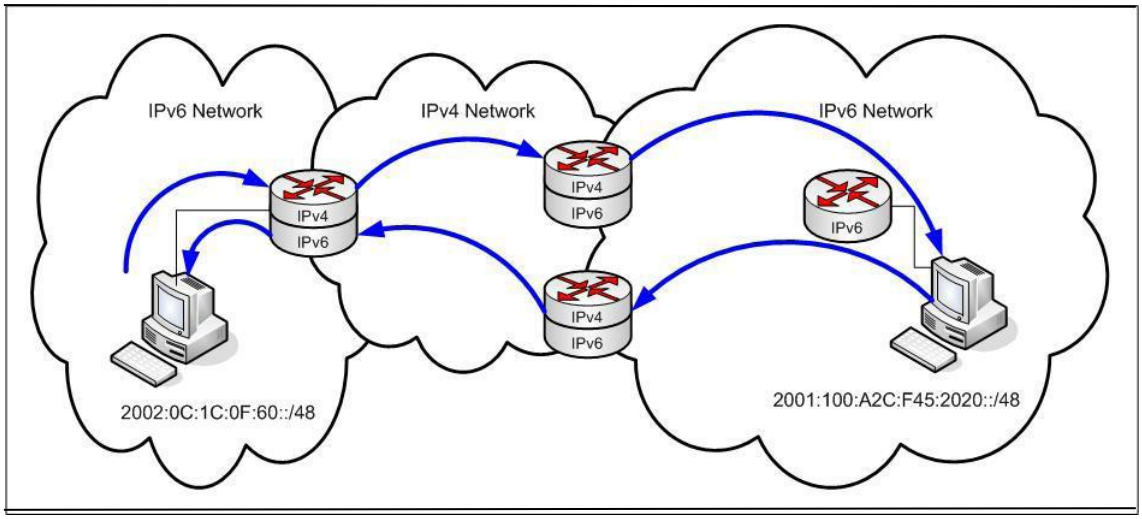


Figure 3.21 Example of Tunneling IPv6 over IPv4 Networks