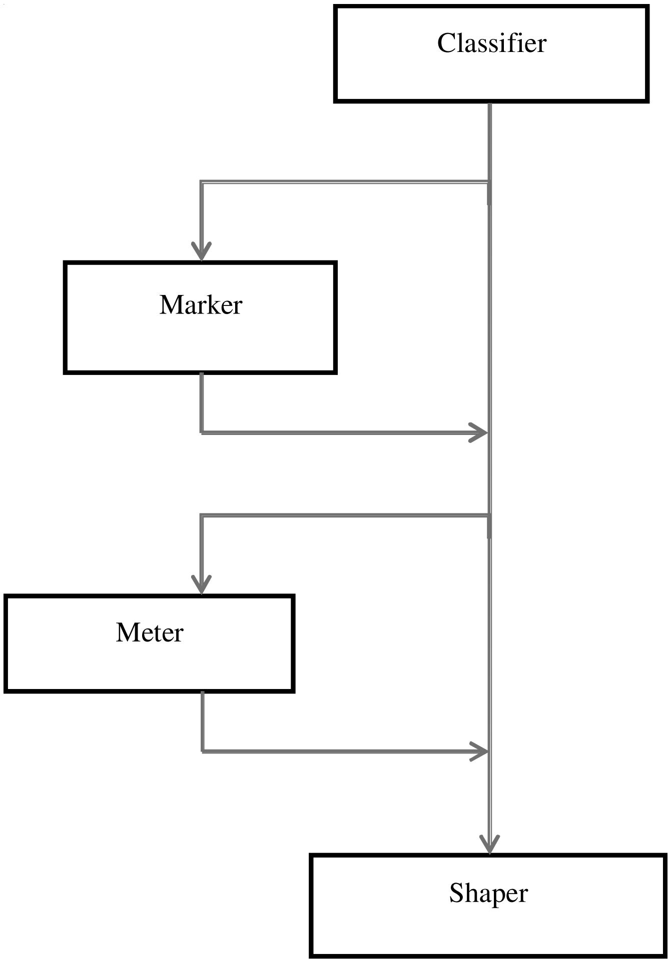

Figure: - **The Differentiated Services (DiffServ) Mode**

address lookup from routing table. The solution is to use a per packet stateless information. For the differentiated service approach several requirements were identified and addressed.

The key requirements are:

1. Independence of applications, services and policing.
2. Deployable incremental (only some part(s) of path), interoperability with quality of service other technologies,
3. No customer or micro flow information or state in core network nodes - no nop-by-hop signaling. Core nodes utilize only small set of simple aggregated classification policies.

4) IPv4 Services

The IPv4 header packet consists of 14 fields, of which 13 are required and 14th field is optional. The IPv4 datagram is conceptually divided into two pieces: the header and the payload. The header contains addressing and control fields, while the payload carries the actual data to be sent over the internetwork. Unlike some message formats, IP datagrams do not have a footer following the payload.

Ver 4 Bits	IHL 4 bits	ToS (8 Bits)	Total Length (16 bits)	
Identification (16 bits)			Flags (3 bits)	Fragmen Offset (1 bits)
Time to Live (TTL) (8 bits)		Protocol (8 bits)	Header Checksum (16 bit	
Source Address (32 bits)				
Destination Address (32 bits)				
Options				

Figure : IPv4 Services

i) Version

Length of version field is 4 bits and it identifies the version of IP used to generate the datagram. For IPv4, this is of course the number 4. The purpose of this field is to ensure compatibility between devices that may be running different versions of IP. In general, a device running an older version of IP will reject datagrams created by newer implementations, under the assumption that the older version may not be able to interpret the newer datagram correctly.

ii) Internet Header Length (IHL)

Internet heard length is 4 bits long and it specifies the length of the IP header, in 32-bit words. This includes the length of any options fields and padding. The normal value of this field when no options are used is 5 (5 32-bit words = 5*4 = 20 bytes). Contrast to the longer Total Length field below.

iii) Type of Service (TOS)

Type of services is 8 bit long field in Ipv4 header designed to carry information to provide quality of service features, such as prioritized delivery, for IP datagrams. It was never widely used as originally defined, and its meaning has been subsequently redefined for use by a technique called Differentiated Services (DS). See below for more information.

iv) Total Length (TL)

Total length is 16 bit long field in IPv4 header that specifies the total length of the IP datagram, in bytes. Since this field is 16 bits wide, the maximum length of an IP datagram is 65,535 bytes, though most are much smaller.

v) Identification

Identification field is 16 bit long in Ipv4 header and this field contains a 16-bit value that is common to each of the fragments belonging to a particular message; for datagrams originally sent un-fragmented it is still filled in, so it can be used if the datagram must be fragmented by a router during delivery. This field is used by the recipient to reassemble messages without accidentally mixing fragments from different messages. This is needed because fragments may arrive from multiple messages mixed together, since IP datagrams can be received out of order from any device.

vi) Fragment Offset

Fragmentation offset field is 13 bits long in Ipv4 header. When fragmentation of a message occurs, this field specifies the offset, or position, in the overall message where the data in this fragment goes. It is specified in units of 8 bytes (64 bits). The first fragment has an offset of 0.

vii) Time to Live (TTL)

This field is 8 bit long specifies how long the datagram is allowed to "live" on the network, in terms of router hops. Each router decrements the value of the TTL field (reduces it by one) prior to transmitting it. If the TTL field drops to zero, the datagram is assumed to have taken too long a route and is discarded.

viii) Header Checksum

Header checksum is 16 bit long field in IPv4 header. A checksum computed over the header to provide basic protection against corruption in transmission. This is not the more complex CRC code typically used by data link layer technologies such as Ethernet; it's just a 16-bit checksum. It is calculated by dividing the header bytes into words (a word is two bytes) and then adding them together. The data is not checksummed, only the header. At each hop the device receiving the datagram does the same checksum calculation and on a mismatch, discards the datagram as damaged.

ix) Source Address

Source address is 32 bits long field in Ipv4 header and it is 32-bit IP address of the originator of the datagram. Note that even though intermediate devices such as routers may handle the datagram, they do not normally put their address into this field—it is always the device that originally sent the datagram.

x) Destination Address

Destination address is 32 bits long in IPv4 header and this is 32-bit IP address of the intended recipient of the datagram. Again, even though devices such as routers may be the intermediate targets of the datagram, this field is always for the ultimate destination.

xi) Options

One or more of several types of options may be included after the standard headers in certain IP datagrams.

5) Services of IPv6

Today's need is moving frequently in internet. By using the IPv4 header, it is very difficult to manage the communication during mobility. IPv4 header does not support direct bigger network because of its header length. It always needs subnet masking. Another important reason is security & authentication during mobility. Now the Pv6 had removed the problems of IPv4 version. IPv6 support 128 bit addressing scheme, it means network can grow with unique identity. It uses different type of header scheme for different purposes. There are Mobility header, Security header, Authentication header and others. All necessary headers attached with IPv6 header and sent. At the receiver take necessary steps.

A host can use the flow label and the traffic fields in the IPv6 header. A host uses these fields to identify those packets for which the host requests special handling by IPv6 routers. For example, the host can request non-default quality of service or real-time service. This important capability enables the support of applications that require some degree of consistent throughput, delay, or jitter. These types of applications are known as multi media or real-time applications.

The nodes that originate a packet must identify different classes or different priorities of IPv6 packets. The nodes use the Traffic Class field in the IPv6 header to make this identification. The routers that forward the packets also use the Traffic Class field for the same purpose.

The following general requirements apply to the Traffic Class field:

a) The service interface to the IPv6 service within a node must supply the value of the Traffic Class bits for an upper-layer protocol. The Traffic Class bits must be in packets that are originated by that upper-layer protocol. The default value must be zero for all of the 8 bits.

b) Nodes that support some of the Traffic Class its or all of the Traffic Class bits can change the value of those bits. The nodes can change only the values in packets

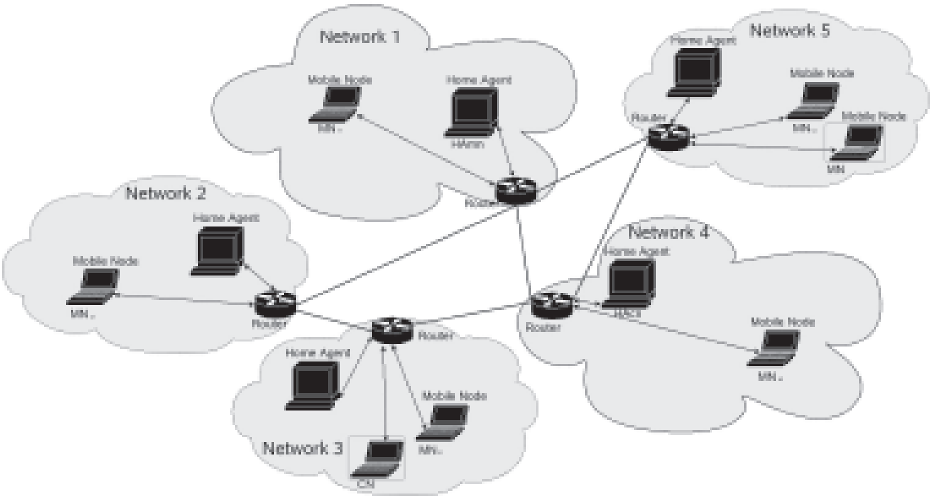

Figure:- Mobile IPv6 Example

that the nodes originate, forward, or receive, as required for that specific use. Nodes should ignore and leave unchanged any bits of the Traffic Class field for which the nodes do not support a specific use.

c) The Traffic Class bits in a received packet might not be the same value that is sent by the packet's source. Therefore, the upper-layer protocol must not assume that the values are the same.

i) IPv6 Security Improvements

The current Internet has a number of security problems. The Internet lacks effective privacy and effective authentication mechanisms beneath the application layer. IPv6 remedies these shortcomings by having two integrated options that provide security services. You can use these two options either individually or together to provide differing levels of security to different users. Different user communities have different security needs.

The first option, an extension header that is called the IPv6 Authentication Header (AH), provides authentication and integrity, without confidentiality, to IPv6 datagrams. The extension is algorithm independent. The extension supports many different authentication techniques. The use of AH is proposed to help ensure interoperability within the worldwide Internet. The use of AH eliminates a significant class of network attacks, including host masquerading attacks. When using source routing with IPv6, the IPv6 authentication header becomes important because of the known risks in IP source routing. Upper-layer protocols and upper-layer services currently lack meaningful protections. However, the placement of the header at the Internet layer helps provide host origin authentication.

The second option, an extension header that is called the IPv6 Encapsulating Security Payload (ESP), provides integrity and confidentiality to IPv6 datagrams. Though simpler than some similar security protocols, ESP remains flexible and is algorithm independent. Similar security protocols include SP3D and ISO NLSP. IPv6 Authentication Header and IPv6 Encapsulating Security Payload are features of the new Internet Protocol Security (IPsec).

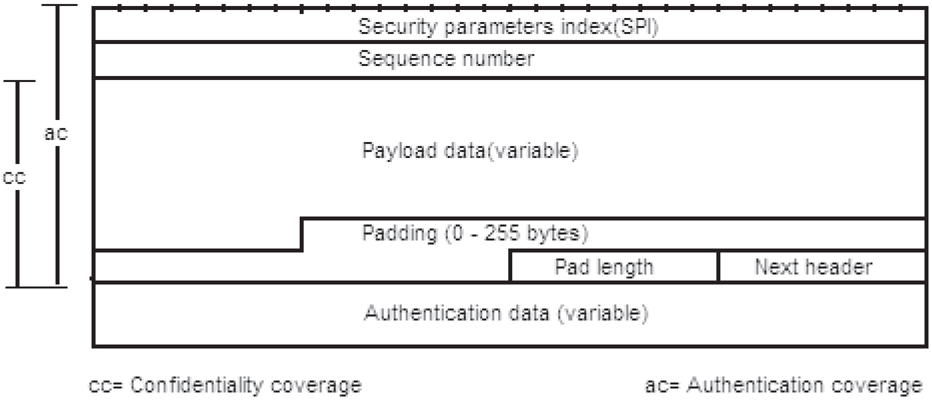
Security parameters index(SPI)
Sequence number
Payload data(variable)
Padding (0 - 255 bytes)
Pad length
Next header
Authentication data (variable)
ac
cc
cc= Confidentiality coverage
ac= Authentication coverage

Figure:**- IPv6 Security Header**

ii) Security Parameters Index

The SPI is an arbitrary 32-bit value that, in combination with the destination IP address and security protocol (ESP), uniquely identifies the Security Association for this datagram. The set of SPI values in the range 1 through 255 are reserved by the IANA for future use; a reserved SPI value will not normally be assigned by IANA unless the use of the assigned SPI value is specified in an RFC. It is ordinarily selected by the destination system upon establishment of an SA. The SPI field is mandatory.

The SPI value of zero (0) is reserved for local, implementation- specific use and MUST NOT be sent on the wire. For example, a key management implementation MAY use the zero SPI value to mean "No Security Association Exists" during the period when the IPv6 sec implementation has requested that its key management entity establish a new SA, but the SA has not yet been established.

iii) Sequence Number

This unsigned 32-bit field contains a monotonically increasing counter value (sequence number). It is mandatory and is always present even if the receiver does not elect to enable the anti-replay service for a specific SA. Processing of the Sequence Number field is at the discretion of the receiver, i.e., the sender MUST always transmit this field, but the receiver need not act upon it.

The sender's counter and the receiver's counter are initialized to 0 when an SA is established. (The first packet sent using a given SA will have a Sequence Number of 1) If anti-replay is enabled (the default), the transmitted Sequence Number must never be allowed to cycle. Thus, the sender's counter and the receiver's counter MUST be reset (by establishing a new SA and thus a new key) prior to the transmission of the 2^32nd packet on an SA.

iv) Payload Data

Payload Data is a variable-length field containing data described by the Next Header field. The Payload Data field is mandatory and is an integral number of bytes in length. If the algorithm used to encrypt the payload requires cryptographic synchronization data, e.g., an Initialization Vector, then this data MAY be carried explicitly in the Payload field. Any encryption algorithm that requires such explicit, per-packet synchronization data MUST indicate the length, any structure for such data, and the location of this data as part of an RFC specifying how the algorithm is used with ESP. If such synchronization data is implicit, the algorithm for deriving the data must be part of the RFC.

v) Padding (for Encryption)

Several factors require or motivate use of the Padding field. First if an encryption algorithm is employed that requires the plaintext to be a multiple of some number of bytes, e.g., the block size of a block cipher, the Padding field is used to fill the plaintext (consisting of the Payload Data, Pad Length and Next Header fields, as well as the Padding) to the size required by the algorithm.

Padding beyond that required for the algorithm or alignment reasons cited above may be used to conceal the actual length of the payload, in support of (partial) traffic flow confidentiality. However, inclusion of such additional padding has adverse bandwidth implications and thus its use should be undertaken with care.

The sender MAY add 0-255 bytes of padding. Inclusion of the Padding field in an ESP packet is optional, but all implementations MUST support generation and consumption of padding.

For the purpose of ensuring that the bits to be encrypted are a multiple of the algorithms block size (first bullet above), the padding computation applies to the Payload Data. And for the purposes of ensuring that the Authentication Data is aligned on a 4-byte boundary (second bullet above), the padding computation applies to the Payload Data inclusive of the IV, the Pad Length, and Next Header fields.

If Padding bytes are needed but the encryption algorithm does not specify the padding contents, then the following default processing MUST be used.

Any encryption algorithm that requires Padding other than the default described above, MUST define the Padding contents (e.g., zeros or random data) and any required receiver processing of these Padding bytes in an RFC specifying how the algorithm is used with ESP. In such circumstances, the content of the Padding field will be determined by the encryption algorithm and mode selected and defined in the corresponding algorithm RFC. The relevant algorithm RFC MAY specify that a receiver MUST inspect the Padding field or that a receiver MUST inform senders of how the receiver will handle the Padding field.

vi) Pad Length

The Pad Length field indicates the number of pad bytes immediately preceding it. The range of valid values is 0-255, where a value of zero indicates that no Padding bytes are present. The Pad Length field is mandatory.

vii) Next Header

The Next Header is an 8-bit field that identifies the type of data contained in the Payload Data field, e.g., an extension header in IPv6 or an upper layer protocol identifier. The value of this field is chosen from the set of IP Protocol Numbers defined by IANA. The Next Header field is mandatory.

viii) Authentication Data

The Authentication Data is a variable-length field containing an Integrity Check Value (ICV) computed over the ESP packet minus the Authentication Data. The length of the field is specified by the authentication function selected. The Authentication Data field is optional, and is included only if the authentication service has been selected for the SA in question. The authentication algorithm specification MUST specify the length of the ICV and the comparison rules and processing steps for validation. Note that although both confidentiality and authentication are optional, at least one of these services MUST be selected hence both algorithms MUST NOT be simultaneously NULL

IPv4 vs. IPv6

The biggest problem in IPv4 is the lack of a big enough address field, 32 bits, and its capability was not used very efficiently.

Address class	Bits	Number of nets	Bits	Addresses
A	7	128	24	16 777 216
B	14	16 384	16	65 536
C	22	4 194 304	8	256

One B class net can be replaced by three C class nets if class B is going to be "empty", but when one address is replaced by three, the router's memory.

IPv6 in the contrary can support at least 10^12 nodes and 10^9 networks.

The routing algorithm have no knowledge how the network has been made and can support all IPv4's routing algorithms, and also support much larger number of hops then IPv4 (limit of 256).

IPv6 can handle different speed of networks; from Extra Low Frequency networks to very high speed of 500Gbits/s. IPv6 provide a security layer that places "options" in separate extension headers while IPv4 does not. The extension headers can be of arbitrary length and has no limit to the amount of options that can be carried. IPv6 has an anycast address that allows nodes to control the path which their traffic flows, IPv4 does not.

IPv6 headers are extensible, the option in IPv4 is not efficient to decode. IPv6 connects to global internet using a combination of its global prefixes (see details in IPv6 Addressing), while IPv4 manually renumbers to connect to the internet. IPv6 renumbers automatically.

Transition from IPv4 to IPv6

The challenge fro an IPv6 is for its transition to be complete before IPv4 routing and addressing break. The 2 transition requirements are the flexibility of deployment and the ability for IPv4 hosts to communicate with IPv6 hosts. IPv6 hosts should be able to communicate with IPv4-only hosts globally.

The first objective to the deployment strategy is to have great flexibility and to implement and maintain the interoperability between the two. Features designed into an IPv6 have to be backwards compatible with IPv4.

The second objective is to allow IPv6 to be deployed in a highly diffuse and incremental fashion, with few interdependencies.

The third objective is that the transition should be easy for users, system administrators, and network operators to understand and carry out. Basic features include:

a) **Incremental upgrade and deployment:** individual IPv4 hosts and routers may be upgraded to IPv6 one at a time without any other hosts or routers to be upgraded at the same time.

b) **Minimal upgrade dependencies:** the only prerequisite is to upgrade DNS server to handle IPv6 address records.

c) **Easy addressing:** When existing installed IPv4 hosts or routers are upgraded to IPv6, they may continue to use their existing address, they do not need to be assigned new addresses.

d) **Low startup costs:** there is no or little work to prepare for the upgrade of IPv4 to IPv6. The mechanisms employed by the IPv6 transition mechanisms include:

i) An IPv6 addressing structure that embeds IPv4 addresses within IPv6 addresses, and encodes other information used by the transition mechanisms. IPv6 nodes are assigned IPv6 unicast addresses that carry an IPv4 address in the low order 32-bits. This type of address has the format:

```
| 80 bits                          | 16  | 32 bits           |
+----------------------------------+---------------------+
|0000....................................0000|0000| IPV4 ADDRESS |
+--------------------------------+------+------------------------+
```

ii) A model of deployment where all hosts and routers upgraded to IPv6 in the early transition phase are "dual" capable (i.e. implement complete IPv4 and IPv6 protocol stacks).

iii) The technique of encapsulating IPv6 packets within IPv4 headers to carry them over segments of the end-to-end path where the routers have not yet been upgraded to IPv6.

iv) The header translation technique to allow the eventual introduction of routing topologies that route only IPv6 traffic, and the deployment of hosts that support only IPv6. Use of this technique is optional, and would be used in the later phase of transition if it is used at all.

The gradual upgrade features of IPv6 transition mechanisms allow the host and router vendors to integrate IPv6 into their product lines at their own pace, and allow the end users and network operators to deploy IPv6 on their own schedules.

References

1. P.C.Saxena, Sanjay Jasola , " Intelligent Model for Mobility of correspondent node in mobile IPv6" computer Standards & interface 28(2006) 737-751
2. P.C.Saxena, Sanjay Jasola " A new model for Mobility of Correspondent Node in mobile IPv6". Vol. 36 No.1, January - March 2006.
3. D. Johnson, C.Perkins Mobility support IPv6 RFC, vol 3775,IETF,2004.
4. S.Glass, et al., mobile IP Authentication ,Authorization and Accounting Requirements , RFC,vol,2977,IETF,2000.
5. S.Deering ,R. Hidden ,Internet Protocol ,Version 6(ipv6) Specification RFC,vol,2460,IETF,1998.
6. http://www.omnetpp.org.