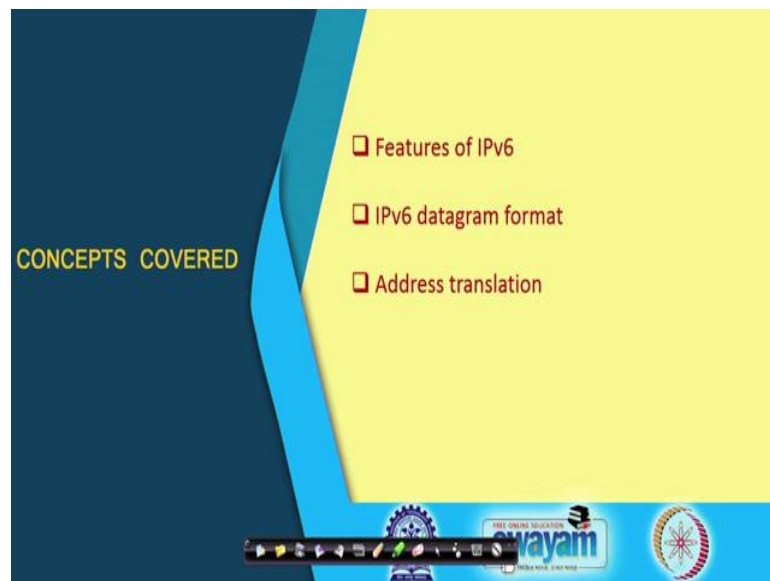


**Ethical Hacking**  
**Prof. Indranil Sengupta**  
**Department of Computer Science and Engineering**  
**Indian Institute of Technology, Kharagpur**

**Lecture - 14**  
**IP Version 6**

In this lecture we shall be talking about IP Version 6 which is the latest version of the IP protocol. Now earlier we have looked at this so called IP version 4 which is more conventional and traditional. Now IP version 4 has some drawbacks which has lead the designers of this protocols, internet protocols to come up with a newer IP version. So, in this lecture we shall particularly see what are the drawbacks of the older version, IP version 4 and what are the new features this IP version 6 incorporates.

(Refer Slide Time: 01:00)



So, in this lecture we shall be talking about some of the features of IP version 6 protocol, the datagram format in IP version 6 and how certain kinds of address translation are carried out ok.

(Refer Slide Time: 01:12)

**Introduction**

- The IP protocol forms the foundation of the Internet.
  - ✓ IP version 4 is used widely today.
    - IPv4 suffers from a number of drawbacks.
    - Need to enhance the capabilities of the protocol.
  - IP Next Generation
    - IPng / IPv6

swamyam

So, let us look at the overall scenario first. Now the first thing is that as we have repeatedly said, this TCP/IP forms the backbone of the internet that we see today and IP protocol is most crucial in the sense that all packets that flow through the internet, they are routed using IP protocol. So, IP ensures that packets reach the intended destination correctly. As I had mentioned, this IP version 4 is more widely used today, but there are a certain number of drawbacks in this protocol.

So, once we identify what this drawbacks are, then we can appreciate that what kind of enhancements were needed to enhance the capabilities and that is how this newer version IP version 6 has come. This is also sometimes known as IP next generation IPNG ok, this both names are used interchangeably.

(Refer Slide Time: 02:30)

The slide is titled "Problems with IPv4" and lists the following points:

- Limited address space.
  - 32-bit address is inadequate today.
- Applications demanding real-time response.
  - Real-time audio or video.
  - Must avoid changing routes frequently.
- Need for more complex addressing and routing capabilities.
  - Two-level structure of IPv4 may not serve the purpose.

A handwritten note in a blue oval on the right side of the slide says:  $2^{32} \approx 4 \text{ billion}$ .

The slide also features a Swayam logo at the bottom left and a small video inset of a man speaking at the bottom right.

Now, problems with IP version 4 are as follows. The first and most important problem is that we are using 32 bit IP addresses,  $2^{32}$  is how much? Close to 4 billion, but you see with the huge proliferation of devices that are getting connected to the internet today, we are now in an age where we are talking about internet of things, there will be a large number of very simple devices, they will also want to get connected to the internet.

So, how can we give unique IP addresses to everybody, this 4 billion is a rather small number by today's scenario ok. So, this cannot satisfy your support, the kind of internet growth that has been projected right, this is one. Secondly, there are many application, you see there are numerous streaming applications today, streaming audio, streaming video, streaming news and so on and so forth, they need some sort of a real time response. Like something is happening, you are watching it live, let us say. Now you will not like, if there is a 10 second pause in between and then the video again starts to play; that is not acceptable in terms of the quality of service right.

So, there are such applications which are increasing day by day where you need real time response. Real time response means, the delay that we encounter must be within some tolerable limit. Now what is the tolerable limit, it depends on the application of course, ok. So, for this kind of real time response one requirement is that you should not change route very frequently, because we mentioned that IP uses some kind of dynamic routing and packets may follow different parts at different times. Now if we consider the packets that

correspond to a video stream, the packets are following one path, suddenly the packet starts following another path which is the longest path, the delay will increase ok.

So, that kind of variable delay may not be desirable for such real time applications. Secondly, there are some scenarios that are coming up where you need more complex addressing and routing capabilities; like you recall in IP version 4, you have a 2 level routing; you have a network, you have a host.

Of course, within the host you, forcibly you use the subnet and introduced a third hierarchy, but that is just within the host address space available to you, but IP version 6 allows more number of levels of addressing which becomes more flexible and of course, it will get more complex.

(Refer Slide Time: 05:50)

**Main Features of IPv6**

- Something is common with IPv4:
  - IPv6 is connectionless – each datagram contains destination address and is routed independently.
  - Header contains the maximum number of hops a datagram can make before being discarded.
  - Some of the other general characteristics are also retained.

So, main features of IPv6, few of the features are borrowed from IPv4, whatever was there, same thing is carried forward. First such thing is connectionless, IP version 4 was based on datagrams, IP version 6 is also based on datagrams, which is a connectionless protocol.

So, each datagram are routed independently and each datagram will contain destination address, so that the intermediate routers can take proper and informed routing decisions ok. And just like IP, if you see in IP, there was a time to live field, it told you what is the maximum number of hops a datagram can take before it will be discarded, at every hop the TTL field was decremented by 1, so whenever it reaches 0, the packets gets discarded.

Similarly, here also the header will contain a similar field maximum number of hops, data datagram is permitted to take, because if you see you have some estimate regarding the maximum number of hops you require to reach a destination.

If you find that even within that number of hops you are not able to reach, it means there is some problem in the network, may be your packet is following a circular path in a loop, it is got stuck somewhere, so in that case you discard the packet and some of the other general characteristic like fragmentation and so on, those are also retained fine.

(Refer Slide Time: 07:34)

- New features of IPv6:
  - Address size: 128-bit addresses are used.
    - ❖  $2^{128}$  total addresses.
    - ❖  $6 \times 10^{23}$  unique addresses per square meter of the earth's surface.
  - Header format:
    - ❖ IPv6 uses a series of fixed-length headers to handle optional information.
    - ❖ A datagram consists of a base header followed by zero or more extension headers.

Now, let us talk about the new features which are interesting. The most important feature is that the number of bits in the source and destination IP address is increased, earlier it was 32, now it is 128, which means I can use up to  $2^{128}$  that many unique addresses. Now how large is this;  $2^{128}$ . This is something you can just, you can actually calculate and find out this is a decimal number which would be having may be 45 digits; 45 digit in decimal number which is huge, but how huge is that?

Just imagine, you think of the surface of the earth, just assume that I want to assign IP addresses to devices on the surface of the earth,  $2^{128}$  is such a large number that on every square meter on the earth surface, you can assign about  $6 \times 10^{23}$  unique addresses.

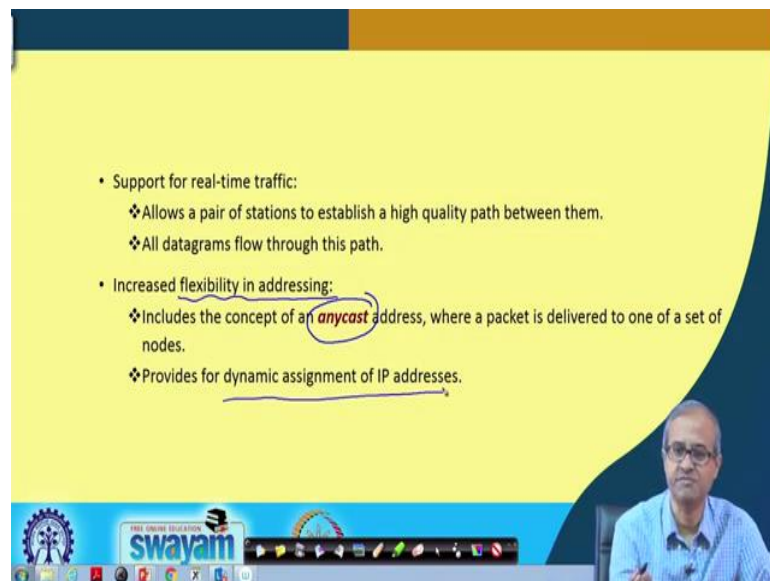
So, this number is indeed huge and it is expected that this will never be too small in the foreseeable future right. So, this 128 bit is good enough, quite large. And another

difference is that in IP version 4, there was a header and the header was fixed. So, whatever packet information the packet carried, it used to be put in the same header, but in IP version 6 what it does, it uses a series of fixed length headers.

Depending on what you need, you add optional headers like, if you do not need fragmentation, you do not use fragment header, if you need then you add a fragment header. So, there would be multiple headers, which will be connected in a linked list as part of the IP version 6 header, this is the basic concept.

So, a datagram in IP version 6 will consist of a base header that is the basic header which will contain for example, the source and destination addresses and followed by optional extension headers. There may not be any, but there can be several.

(Refer Slide Time: 10:35)



- Support for real-time traffic:
  - ❖ Allows a pair of stations to establish a high quality path between them.
  - ❖ All datagrams flow through this path.
- Increased flexibility in addressing:
  - ❖ Includes the concept of an **anycast** address, where a packet is delivered to one of a set of nodes.
  - ❖ Provides for dynamic assignment of IP addresses.

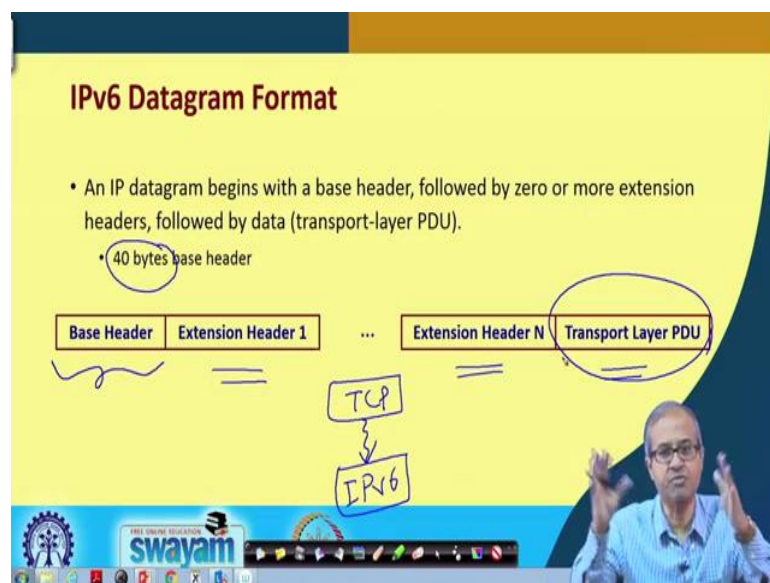
There are some additional features also; like as I had said for real time traffic, there are some additional features which have been included in IP version 6. Like it is possible for two ends, two hosts to agree on a high quality path, suppose I am viewing streaming video and I tell that well I want a guaranteed bandwidth of 512 kilobytes per second. So, they will agree along with the intermediate routers to provide you with a dedicated path. A path through which you will get dedicated 512 kbps of bandwidth, so that you can receive this streaming media without any disruption right and all datagrams will follow this path.

So, you see this is a slide deviation from the connectionless approach that IP versions is also supports. For real time traffic, you have something like a connection oriented concept coming in ok. So, it is like a hybrid now, it is not a pure datagram approach, sometimes when you need to have quality of service guarantee for real time applications, you may also specify the path which the datagrams will follow ok.

And the next thing is that there is some new kinds of addressing which has come up. Like earlier there were addressing like host to host addressing, you are sending to a particular host or broadcast sent to everybody, but now there is a new addressing mode called anycast addressing which have been introduce.

Anycast means there are a set of computers, I am telling you send this message to anyone of these ok. I do not specify which one, but any one of these. So, if it reaches anyone of these, I am happy. This is refer to as anycast address and also this provides for dynamic assignments of IP address, it can change over time, this facilitate result there.

(Refer Slide Time: 13:02)

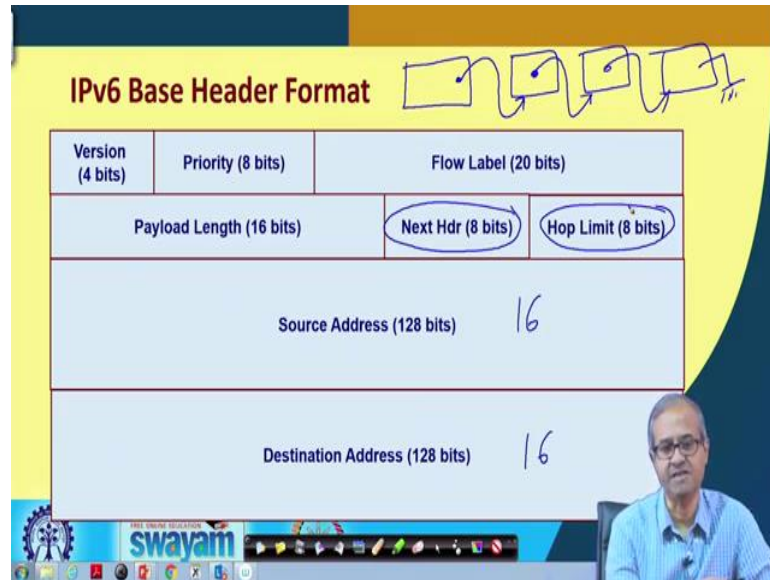


So, IP datagram format as I had said, there can be multiple extension headers, the general picture is like this, it starts with the base header which is 40 bytes in size, but there can be multiple extension headers, the size of the extension headers can be different and at the end, you have the data, because typically let us say TCP is running on top and below you have this IP version 4/6 running, IPv6.



So, TCP will be sending you some message for transmission, so that PDU, Protocol Data Unit, transport layer protocol data unit that is the message will be appended with the IP header, whatever there are and they will be send as IP packet.

(Refer Slide Time: 14:06)



Now, the base header, 40 bytes, will be having a fixed structure like this, which is some of the fields or somewhat similar to, I mean IP version 4, but because of the change, size of the address as you see, source address and destination address are 128 bits long. Well 128 bits means how many bytes? 16 bytes, you have 16 bytes here, you have 16 bytes here, version is 4 bits.

This is the version 6, so this will contain the number 6, 0 1 1 0. Priority you can specify a priority in the packet now, which in a version 4 is not possible. Higher priority packet will move faster, router will give higher priority to those. Flow level, this is again concerned with flow control, some additional fields or information are there.

I am not going into the detail of all of them. Then comes the payload length which is the size of the total packet. Next header, I said that there will be a base header and there can be a number of extension headers ok. Now there will be a field here which will be pointing to the next header, this will be pointing, so this will be like a link list.

So, this next header field will contain the information that whether there is a, there is an extension header following this or not. If there is then it will contain what type of extension



header and if it does not contain any more, then it will contain some delimiter indicating that there is no more extension headers and this hop limit, I told you about just like time to leave that field is also there. This is the basic header in IP version 6.

(Refer Slide Time: 16:09)

**The Fields**

- **Version** (4 bits): contains the value 6. *0110*
- **Priority** (8 bits): specifies routing priority class.
- **Flow Label** (20 bits): used with applications that require performance guarantee.
- **Payload Length** (16 bits): total length of the extension headers and the transport-level PDU.
- **Next Header** (8 bits): identifies the type of information that immediately follows the current header (IP extension, TCP or UDP).

Now talking about the fields I have already mentioned, let me go through this quickly. This version I told you about, this is a 4 bit field, IP version 6, it will contain the value 6 which in binaries 0110. Priority; specifies the routing priority class, higher priority packets will be handled faster by the routers.

Flow level; this is particularly used for real time application, where there are applications that require performance guarantee. So, flow label contain relevant information, this will contain some specific id or code which will allow the packets to follow this particular path for example, right. Payload length is the total length of the extension header plus the transport level protocol data unit; that means, the data you are sending.

Next header; as I said, it is the pointer to the next extension header, it identifies the type of information that immediately follows the current header, it can be an IP extension or it can be the actual PDU, TCP or UDP ; that means, the data.

(Refer Slide Time: 17:33)

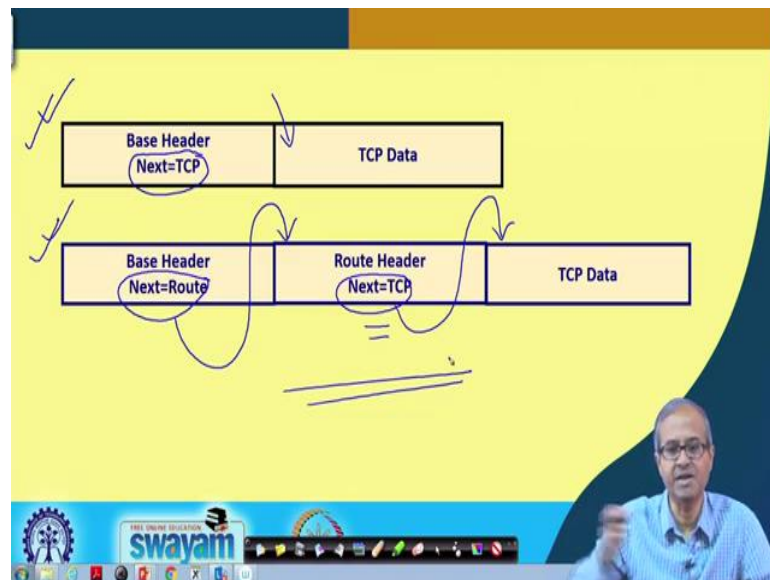
• **Hop Limit:** decremented by 1 at each hop; discarded when it reaches 0.

• **Source/destination addresses:** 16 octets (128 bits) each.

The slide features a yellow background with a blue and orange header. At the bottom, there is a Swayam logo and a video feed of a man in a blue shirt.

Hop limit; similar to IP version 4, it is decremented by 1 at each hop and the addresses are 128 bits or 16 bytes or octets ok.

(Refer Slide Time: 17:47)



So, pictorially it will look something like this. This is one example, suppose this is a IPv6 packet which does not contain any extension header. So, in the base header, the relevant fields are there and the next header field contains TCP, which means it will know that the, that whatever follows the base header is the actual TCP PDU which is carrying the data to be transmitted.

Take this next example where there one extension header. So, base header will contain an information *next = route*, it tells, it has an extension header which gives you some information about route, maybe it is a source routing or specifying the route to be followed. And in the next header, this extension header, there will also be a next field which will tell TCP; that means, this will be the actual TCP data. So, here there can be multiple such header, so this next pointer will be following this headers one by one ok.

(Refer Slide Time: 19:06)

**IPv6 Extension Headers**

- **Routing Header**
  - Provides source routing.
- **Hop-by-hop Options Header**
  - Defines special options that are processed at each hop.
- **Fragment Header**
  - For fragmentation and reassembly.
- **Authentication Header**
  - For packet integrity & authentication.

**All Extension headers are chained in a linked list.**

- Through **Next Hop** field.

swayam

Now, the important kinds of extension headers which are there, they are summarized here, there can be a routing header as your seeing here. Routing here, routing header permits source routing. No I am repeating here, what source routing actually means. Now in the conventional packet routing what happens?

There are routers, a packet arrives at a router, the router looks at the destination address, looks at its own routing table and makes a decisions where to forward this packet. So, it finds out which is the next router I have to send this packet, so that it can reach the destination correctly or if it is destined to a host in my same network, I can directly send it to the network, to that host, directly send it to that host.

Source routing means, the person who is generating the packet, the host directly tells that which sequence of routers must be followed for my packets, because I know that this is a good path, the speed is high, so my packate will go faster. So, if I know that I can specify

that entire information as a routing header and this is what is meant by source routing. The source is specifying the route which path to follow right.

Second comes the hop by hop options header. Well you can specify the second depends on some applications that every hop, means actually whenever you move from one host to the other, you can check for certain conditions, depending what on that you can decide whether to forward the packet, not forward or do something else.

Then for fragmentation you have fragment header, this will contain information just like an IP version 4 which helps in fragmentation and reassembly; some flags, fragment offset and so on. And to ensure packet integrity, there is also an authentication header, where you can implement some kind of hash function, we will be talking about hash function later in this course. This hash functions are used for authentication purposes. Well authentication means suppose I am a router, I have received the packet from some other router x.

Now, there can be two scenarios; one the router x is actually sending the packet, the packet I am getting is actually coming from x or it may so happen that some kind of a malicious entity, it can be a hacker or some other node which is trying to send some you can say illegal packets to my network, is sending a packet in such a way that this source address is the same as x, but it was not send by x, this is what is meant by authentication.

I must be sure that whatever I am getting is actually coming from the person or the node which it is claiming as a resource that is authentication right. And here as I told all the extension headers are changed in a linked list using the next header field which is present in the base header as well as all the extension headers.

(Refer Slide Time: 23:13)

**A Point About Fragmentation**

- IPv6 fragmentation is similar to that in IPv4.
- Required information contained in a separate fragment extension header.
  - Presence of the fragment header identifies the datagram as a fragment.
  - Base header copied into all the fragments.

The diagram illustrates the fragmentation process. A single box labeled 'BH' (Base Header) is shown at the top. Two arrows point downwards from this box to two separate boxes. Each of these boxes contains 'BH FR' (Base Header and Fragment Header) and a checkmark, indicating that the base header is copied into each fragment and a separate fragment header is added.

Now, about fragmentation just as I mentioned, this IP version 6 fragmentation is similar to IP version 4, but unlike IP version 4 where the additional information or fragmentation was put inside the same header, here we are using a separate extension header. So, when a fragment get, there is at when IP version 6 packet, let us say there was a packet which had a base header, let us say BH, when it gets fragmented into two pieces, the scenario will be something like this.

This base header will be copied to both the fragments and there will be a fragment header containing relevant information, these are extension header, this will be added to the two fragments and the data one part will come here, the next will come here. This is how fragmented packets will look like.

(Refer Slide Time: 24:35)

**IPv6 Addressing**

- Addresses do not have defined classes.
  - A prefix length associated with each address (flexibility).
- Three types of addresses:
  - ✓ **Unicast:** corresponds to a single computer.
  - ✓ **Multicast:** Refers to a set of computers, possibly at different locations. Packet delivered to every member of the set.
  - ✓ **Anycast:** Refers to a set of computers with the same address prefix. Packet delivered to exactly one of the computers in the set.
    - ✦ Required to support replication of services.

Now, talking about IP version 6 addressing as I told you earlier that it introduces a new kind of addressing called anycast. So, here the different addressing types are summarized. Here there is no concept of classes just like IP version 4, there was class A, B, C, D, E we are defining, but here there is no concept of address class, classless. You specify a prefix length very similar to CIDR, Classless Internet Domain Routing that how many bits you will be using for the network and how many bits for the host that you specify here and that introduces a lot of flexibility.

Broadly three classes of addresses are supported; one is a unicast address which means it is a directed address of a particular node in the Internet; that means, you are sending a packet to a particular computer; that is a unicast address. Multicast or broadcast, multicast means you are sending the packet to all members of a given set, you can say that I am sending a packet to a subnetwork, it should go to all the computers inside that subnetwork, that is a multicast address right.

So, it must be delivered to every member of the set. Set is usually a network or a sub network, then you have this new address anycast, where it says the packet will be delivered to exactly one of the computers in the set. In multicast it was delivered to everybody, but here it is delivered to exactly one.

(Refer Slide Time: 26:28)

**Colon Hexadecimal Notation**

- An IPv6 address is 128 bits long.
  - Dotted decimal notation too long.
  - Use colon-hexadecimal notation. Each group of 16 bits written in hex, with a colon separating groups.
  - Example:  
`7BD6:3DC:FFF:FFF:0:2D:F321:FFFF`  
`7BD6:0:0:0:0:0:B6` → `7BD6::B6`
  - Sequence of zeros is written as two colons.

Handwritten notes: 03DC, 0000, 002D

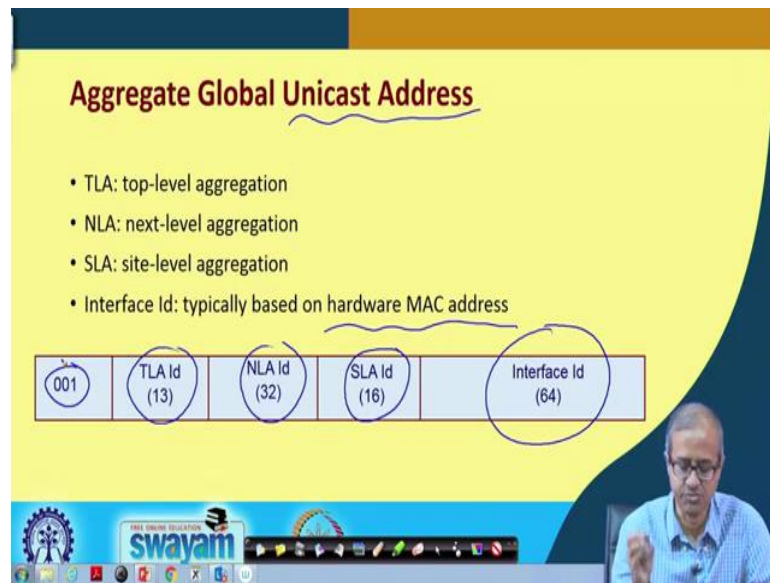
And now the way this 128 bit address has specified. For IP version 4, you use the dotted decimal notation you recall, but 128 bits is too long, if you use the dotted decimal notation after 8 bits, there would be 16 such numbers separated by dots. So, in IP version 6, they have come up with the some kind of a shortcut notation. There is one thing to observe that in this 128 bit address which is pretty large, there are many bits in between which are zeros and large number of these 128 bits are often zeroes.

So, we use something called colon, instead of dot, we use a colon-hexadecimal notation. Like you say this is an example, instead of decimal we are using hexadecimal, each digit represents 4 bits. So,  $\frac{128}{4}$  is 32, I need 32 hexadecimal digits. See 1, 2, 3, 4, 5, 6, 7, 8,  $8 \times 4$  is 32. So, each entity between two colons represent 4 hexadecimal digits. Like for example, here I mentioned 3DC, 3DC is not 4, I have mentioned only 3, it means there is a 0 before that. Similarly, I had mentioned only 0 here. Only 0 means this is all 0, 16 0's, 2D means there are two hexadecimal 0's before that, so like this you can specify.

So, you can write it, if there are; if there are a large number of 0's in between which is often the case as I told you, you can either write it like this or you can even use a shorter version, the beginning is non zero, the ending is non zero, in between everything is 0, you use two colons side by side, it will indicate that everything in between are all 0's ok. So, these are some ways to express the IP version 6 address in a compact fashion.



(Refer Slide Time: 28:55)



Now, talking about the 128 bit address, you see this 128 bit address is typically separated using a number of hierarchies. The last 64 bits here, this specifies something similar to the host address path in IP version 4, it talks about some kind of unique addressing inside the network that you are addressing.

Now it can be, this interface Id can also be based on the hardware MAC address if you want to. Typically hardware MAC address are 48 bits in size and they are unique, you can use that same MAC address here if you want or if you want the network administrator can assign sequential numbers 1,2,3,4,5, like in a normal IP version 4 address numbering.

And there are several other hierarchical fields, site level aggregation, next level aggregation, top level aggregation. Now what this means it may depend on from I mean its one place to another, like at the top level it may indicate country for example, that which country the network belongs to, next level may indicate within that country which internet service provider is providing with the connection, the third level can provide some other hierarchical information; that means, within state level or district level some information.

So, you can have this kind of hierarchical address definitions which is very flexible, depending upon your need, you can define this fields, but if you do not want some of these fields can be left 0's and this unicast address always starts with 001 ok, this is what looks like. This is just an example I gave for unicast address.

(Refer Slide Time: 31:06)

**IPv4-Mapped IPv6 Addresses**

- Allow a host that supports both IPv4 and IPv6 to communicate with a host that supports only IPv4.
- IPv6 address is based on IPv4 address.
- 80 0's, followed by 16 1's, followed by a 32-bit IPv4 address.

Handwritten diagram: A box divided into three sections. The first section is labeled '80' and contains '0's'. The second section is labeled '16' and contains '1's'. The third section is labeled '32' and contains '32-bit'. Below this, an arrow points from an empty circle to a circle containing 'v4', with '32-bit' written next to it.

swayam

Now you see, most of the networks today are IP version 4 networks. Now we are saying that IP version 6 is required, we should migrate to IP version 6 whenever we can. Now there are some incompatibilities, the packet sizes, packet formats are all different.

So, you can have two kind of, this kind of scenarios for compatibility; one is called IP version 4 mapped IP version 6 addresses. This says within a host you install both IPv4 and IPv6 software. So, a host can support both IPv6 and IPv4 and that kind of a host is wanting to communicate with another host which supports only version 4.

In that case what is done and if the packet which is being produced is a IP version 6 packet with a IP version 6 address, then you generate some kind of an IP version 6 address which is formed like this; 80 0's followed by 16 1's, followed by a 32 bit IP version 4 address, because the person you are sending to, that is a IP version 4 network which means it will have 32 bit IP addresses.

So, this 128 bit address that will have to prepare, the last 32 bits will contain the actual IP address of the destination IP version 4, then there will be 16 1's; 16 1's and in the first part there will be 80 0's. This is the convention which is followed.

So, any address like this will mean that this is an IPv4 mapped IPv6 address, where the last 32 bit is actually representing an IP version 4 address.

(Refer Slide Time: 33:24)

**IPv4 Compatible IPv6 Addresses**

- Allows a host supporting IPv6 to talk IPv6 even if the local routers do not talk IPv6.
  - Tell endpoint software to create a tunnel by encapsulating the IPv6 packet in an IPv4 packet.
  - 80 0's, followed by 16 0's, followed by a 32-bit IP address.

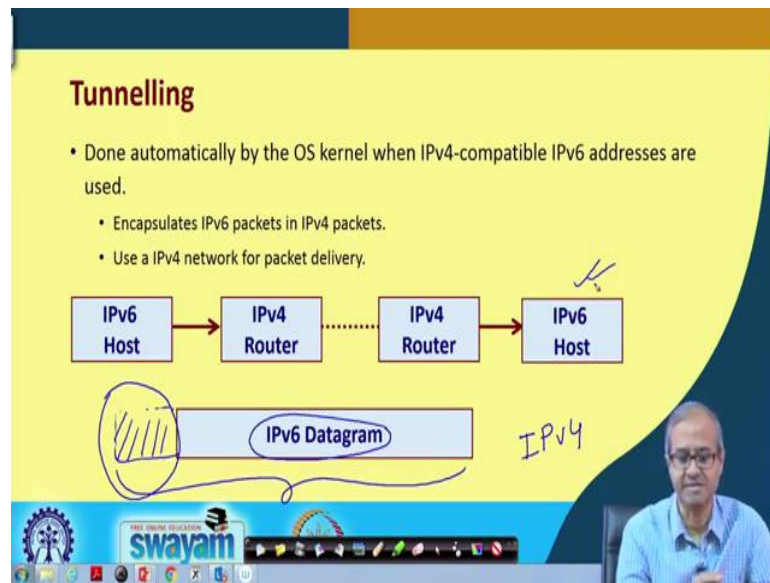
The diagram illustrates the tunneling process. It shows three nodes: a source IPv6 node, a central IPv4 node, and a destination IPv6 node. An arrow labeled '32' points from the source IPv6 node to the IPv4 node, representing the encapsulation of the IPv6 packet into an IPv4 header. Another arrow points from the IPv4 node to the destination IPv6 node, representing the delivery of the encapsulated packet. Below the IPv4 node, a box labeled 'V4' contains a smaller box labeled 'V6', representing the IPv6 packet encapsulated within the IPv4 header.

Similarly, you can have another option which is called IP version 4 compatible IP version 6 address. Here it is something like this, suppose there is a network running version 6, there is another network running version 6, but in between there maybe routers through which the packets are flowing, they may be version 4 compatible. So, how will this version 6 packets flow in this network?

So, here you use something called tunneling, this IPv6 packets which are coming, these are the IPv6 packets. You create an IP version 4 header on top of it and create an IP version 4 packet. Let the IP version 4 packet flow through this network, this we are call tunneling. That means, a version 6 packet is tunneling through the network encapsulated inside the version 4 packet, that is the idea.

And here the way we specify the address is that 80 0's followed by; that means, 96 0's followed by the 32 bit IP address; that means, when you are sending this packet to this intermediate router, it is a version 4 network which means it will be having a 32 bit address right. So, that address you specified like this 96 0's followed by the 32 bit address.

(Refer Slide Time: 35:14)



So, this is what tunnelling is all about, which was used in the previous case. So, here diagrammatically it is explained. So, you have an IP version 6 host which is generating an IP version 6 packet. Then you have to send it to IPv4 router, what to do? You append an IPv4 header before it.

So, the whole thing becomes an IPv4 packet. This IPv4 packet gets routed through an IPv4 network and it reaches the final destination. Now in the final destination this packet is stripped out and this IP version 6 datagram is brought out and it is delivered to the IPv6 host, this is what is meant by tunneling right.

(Refer Slide Time: 36:10)

**Transition from IPv4 to IPv6**

- Three alternate transition strategies:
  - a) **Dual stack:** Both IPv4 and IPv6 protocol stacks supported in the gateway.
  - b) **Tunneling:** An IPv6 datagram flows through an intermediate IPv4 network by encapsulating the whole IPv6 packet as payload.
  - c) **Header translation:** An IPv4 address is translated into a IPv6 address, and vice versa.

So, talking about transition to IPv4 to IPv6, due to the incompatibilities, there are challenges, but there are broadly three approaches you can think of; one is you implement both IPv4 and IPv6 protocols in the router or the gateway, so that if someone wants to use or send IPv4 packets it can be handled, v6 packets can also be handled something like that.

You can use tunneling just like, just like I mentioned, so IPv6 datagram can flow through intermediate IPv4 networks using tunneling, using encapsulation or you can do some header translation, and the IPv6 packet you can translate into an IPv4 packet and on the other side, you can again translate it back to IPv6 packet.

But the problem is that many of the features IPv6 supports which were not there in IPv4, will get lost once you translate it into a IPv4 packet ok. So, this has very limited use. So, with this we come to the end of this lecture where we talked about some of the salient features of the IP version 6 protocol.

Now in the next lecture we shall be looking at various examples, where we shall be showing you how routing tables are constructed for specific networks, how packets are handled by the routers, how packet forwarding takes place, subnetting, CIDR, all these things that we have studied earlier, we shall be illustrating through some examples.

Thank you.