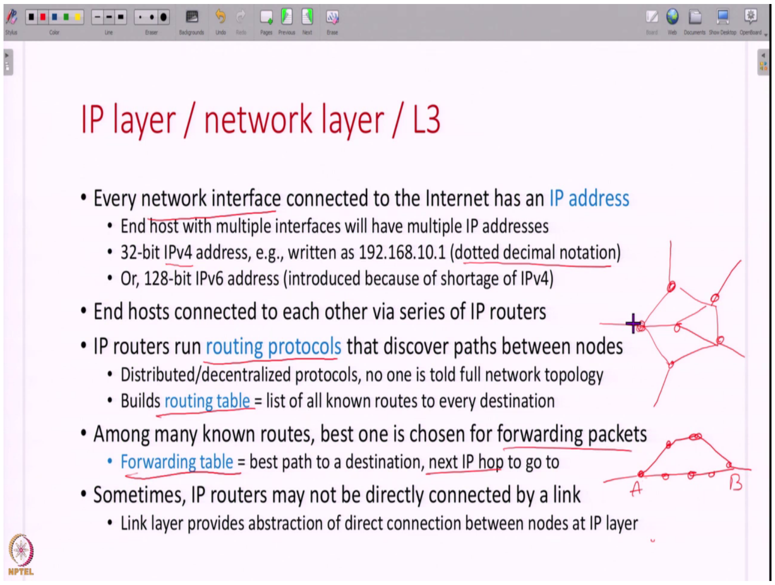
Design Engineering of Computer Systems
Professor. Mythili Vutukuru
Computer Science and Engineering
Indian Institute of Technology, Bombay
Lecture 32
Internet Routing and Forwarding

Hello, everyone. Welcome to the twenty second lecture in the course Design and Engineering of Computer Systems. So, this week we are discussing how the Internet works and in this lecture we are going to understand how Internet routing and forwarding works. So, let us get started.

So, in the previous lecture, we have seen the concept of the various layers that are there in the network stack. Internet software is built using the principle of layering. And in that the network layer is sort of in the middle. It is also called the IP layer or layer 3. And it deals with forwarding of network packets from one IP address to the other. We have seen this in the previous lecture.

(Refer Slide Time: 01:00)



IP layer / network layer / L3

- Every network interface connected to the Internet has an IP address
  - End host with multiple interfaces will have multiple IP addresses
  - 32-bit IPv4 address, e.g., written as 192.168.10.1 (dotted decimal notation)
  - Or, 128-bit IPv6 address (introduced because of shortage of IPv4)
- End hosts connected to each other via series of IP routers
- IP routers run routing protocols that discover paths between nodes
  - Distributed/decentralized protocols, no one is told full network topology
  - Builds routing table = list of all known routes to every destination
- Among many known routes, best one is chosen for forwarding packets
  - Forwarding table = best path to a destination, next IP hop to go to
- Sometimes, IP routers may not be directly connected by a link
  - Link layer provides abstraction of direct connection between nodes at IP layer

So, now let us understand what is an IP address. An IP address is a unique identification that is given to every network interface that is connected to the Internet. Earlier we said that every end host has an IP address, it is actually every network interface. That is if you have an Ethernet card and a wifi card on your laptop, you will get two IP addresses. The Ethernet card that interface will have its IP address. The wireless interface also will have its own IP address. So, every interface that is connected to the Internet has an IP address.

And these IP addresses are 32 bit, either 32 bit IPv4 addresses which are written this way in that dotted decimal notation or you can have 128 bit IPv6 addresses. So, IPv6 has come about primarily because of a shortage of IPv4 addresses. We are running out of IPv4 addresses. And end hosts are connected to each other using a series of IP routers and these routers run routing protocols.

So, these are distributed, decentralized protocols. That is, each router runs this protocol on its own and they exchange information with each other. And as a result of this communication, as a result of this routing protocol, all the routers discover all the paths between the various hosts of the Internet. And all this information about all the routes that are available is stored in the routing table.

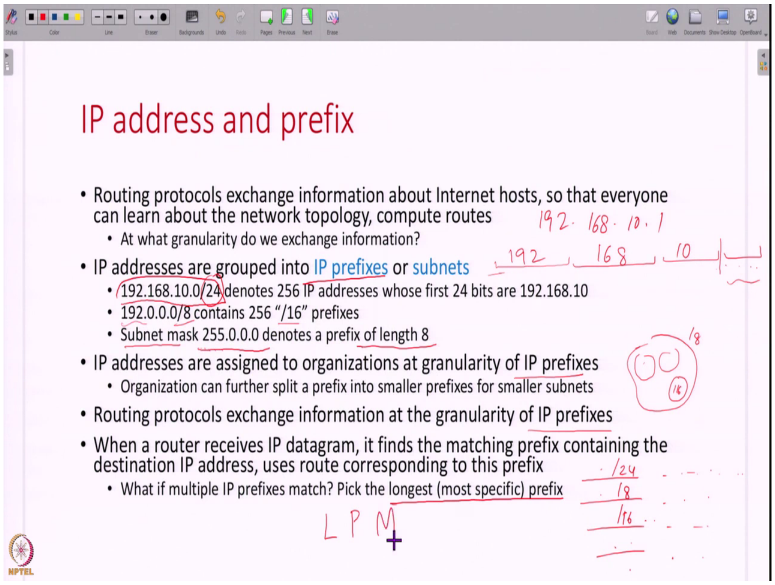
And from among these multiple routes available, one of the routes is chosen, the best route is chosen for forwarding packets and you store the best route is what is called the forwarding table. So, the forwarding table has for every destination which is the best path to send the packet to. For example, if you have a long path like this and a short path like this between two hosts A and B, then these routers along the shortest path will send the packet like this and not send the packet like that.

So, usually routers forward traffic on the shortest possible or the best possible path and that information of for each destination which is the best path and where should I send the packet to in order to go on the best path, what is my next hop, that information is stored in the forwarding table. So, the forwarding table is some kind of a summary of the routing table.

And note that not all IP routers, so I am just drawing a line like this between routers, but does not mean they are always connected by like one wire running from one to the other. So, the link between two IP routers can also be a very long link. There could be other elements on the path. So, all of that, the IP layer does not care about that is taken care of by the link layer.

So, the link layer deals with how to provide this connectivity between two endpoints, between two IP routers or an end host in an IP router that is taken care of by the link layer. We will come to that later. For now just assume that there are multiple hosts and routers each with their own IP address which are all connected to each other.

(Refer Slide Time: 04:16)



IP address and prefix

- Routing protocols exchange information about Internet hosts, so that everyone can learn about the network topology, compute routes
  - At what granularity do we exchange information?
- IP addresses are grouped into IP prefixes or subnets
  - 192.168.10.0/24 denotes 256 IP addresses whose first 24 bits are 192.168.10
  - 192.0.0.0/8 contains 256 "/16" prefixes
  - Subnet mask 255.0.0.0 denotes a prefix of length 8
- IP addresses are assigned to organizations at granularity of IP prefixes
  - Organization can further split a prefix into smaller prefixes for smaller subnets
- Routing protocols exchange information at the granularity of IP prefixes
- When a router receives IP datagram, it finds the matching prefix containing the destination IP address, uses route corresponding to this prefix
  - What if multiple IP prefixes match? Pick the longest (most specific) prefix

So, let us understand next the concept of what is an IP prefix. So, now we have seen that routing protocols, they exchange information about hosts. Now, at what granularity do you exchange information? There are 2 to the 32 IP addresses. And if everybody has to talk and exchange information about all of these 2 to the 32 hosts that is a lot of information on the Internet. So we do not want to tell everybody about every host. Instead what we do is we exchange information at the granularity of groups of posts or subnets. That is IP addresses are grouped into subnets.

How, by using a common prefix. They are grouped hierarchically. So, if you have an IP address, a 32 bit IP address and the first 24 bits are common, the first 24 bits are 192, 168, 10 and the last 8 bits can be anything, then how many such IP addresses do you have. You will have 256 such IP addresses. The last 8 bits can be any of the two to the eight possibilities. Then these 256 IP addresses are combinedly referred to as 192.168.10/24. So, the slash 24 is saying that the first 24 bits are common. The last 8 bits can be anything. So, this is called an IP prefix.

An IP prefix is a group of Internet addresses which have a common prefix. Similarly, if you look at this prefix, this is the first 8 bits are common. And of course, if the first 8 bits are common then inside this you will have 256 /16 prefixes and each slash 16 will have 256 /24 and each /24 will have 256 IP addresses. You can group these prefixes into bigger and bigger prefixes. And a prefix is also denoted by its subnet mask. The subnet mask specifies that the first 8 bits 255 are used all one, the first 8 bits are used, then remaining bits are not used that denotes a subnet mask.

You can specify either the prefix length. You can say this is a prefix of length 24, prefix of length eight or you can specify the subnet mask. These are two ways of describing how big this grouping of IP addresses is. And IP addresses are assigned at the granularity of IP prefixes. If an organization, so each end host want go somewhere and get an IP address for itself, instead an organization will be given an IP prefix, a group of IP addresses will be given.

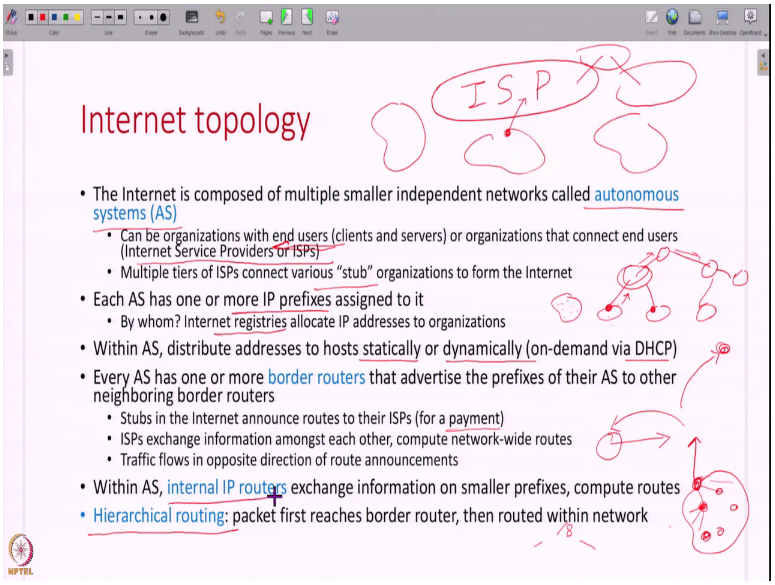
As you know if an organization is given a slash 8, it can split it into multiple slash 16 prefixes that can, or into slash 24 prefixes whatever. Within an organization you can always split a prefix like how the slash 8 contains multiple slash 16 which contain multiple slash 24s. You can always split a prefix into smaller prefixes, but they are managed at this granularity not at the granularity of individual IP addresses.

Similarly, routing protocols also exchange information at the granularity of IP prefixes. So every router will say, okay, these are all the prefixes that I am managing. I know how to get to these prefixes. This is the path to this prefix. All of that information is not stored at the granularity of individual destination but at the granularity of IP prefixes. And when a router receives an IP datagram, even your forwarding table everything will have for this /24 prefix, for this /8 prefix, for this /16 prefix, for various prefixes this is the path. That is how your routing tables, forwarding tables everything will be stored.

And when a router gets an IP packet? It will find out for this IP address which prefix does it belong to and accordingly for that prefix what should I do, where should I forward the packet to. Now, what if multiple prefixes match? For example, if you have an IP address 192.168.10.1 this IP address matches the /24, it matches this /24, it also matches the 192/8 it matches that prefix also. What if it matches multiple prefixes? Then the logic is that you will pick the longest prefix that is the most specific prefix, because that is the convention on the Internet. The most, the longer your prefixes that is more preferable.

So, in this way routing tables, forwarding tables store information about Internet hosts this routing information or forwarding information at the granularity of IP prefixes of various lengths. And this matching algorithm where if an IP address matches multiple prefixes you pick the longest one this is also called the longest prefix match or LPM algorithm.

(Refer Slide Time: 09:44)



Internet topology

- The Internet is composed of multiple smaller independent networks called autonomous systems (AS)
  - Can be organizations with end users (clients and servers) or organizations that connect end users (Internet Service Providers or ISPs)
  - Multiple tiers of ISPs connect various "stub" organizations to form the Internet
- Each AS has one or more IP prefixes assigned to it
  - By whom? Internet registries allocate IP addresses to organizations
- Within AS, distribute addresses to hosts statically or dynamically (on-demand via DHCP)
- Every AS has one or more border routers that advertise the prefixes of their AS to other neighboring border routers
  - Stubs in the Internet announce routes to their ISPs (for a payment)
  - ISPs exchange information amongst each other, compute network-wide routes
  - Traffic flows in opposite direction of route announcements
- Within AS, internal IP routers exchange information on smaller prefixes, compute routes
- Hierarchical routing: packet first reaches border router, then routed within network

So, next let us understand how the Internet structure or the Internet topology looks like. So the Internet is not one big giant network but it is composed of multiple smaller independent networks which are called autonomous systems or ASs. So, this autonomous systems can be either end user organization like organizations that have clients, servers, your college, your company or somebody providing a web service, NPTEL, all of these are end user organizations.

The Internet has these end user organizations. It also has what are called Internet service providers who connect all of these end user organizations. So, you have an end user organization here, here, all of these are connected by an ISP, or an Internet service provider. And then you have another ISP, then you have another ISP, these ISPs are all connected to each other. So you have multiple tiers, multiple layers of these ISPs, which together connect all these end user or stub organizations and your Internet looks like this.

There are various end user organization, then there are various ISPs, which are all connected to each other, there is a hierarchy and this is how the Internet is structured. You have stubs and you have ISPs. And every network here, whether it is an ISP or an end user organization, each network will get a few IP prefixes for itself, from where there are what are called registries in the Internet, which are sort of neutral third parties, which allocate IP addresses to these various organizations.

So, if you are a company or an organization, you will go to a registry, get an IP prefix, and then you will allocate that IP prefix to all the computers, to all the hosts inside your organization. And how are these IP addresses distributed within an organization, you know an organization got a slash 24, you have 256 IP addresses, how do you give it to, you can either statically give it to a host. You can say every computer, this is your IP address. But a lot of, a computer is not using its IP address. Therefore, you can also do a dynamic allocation.

So, there is a protocol called DHCP. There is a DHCP server. And if a computer wants an IP address temporarily, it can go to the DHCP server and get an IP address. Use it for a short period of time. And when the computer is not using this IP address, somebody else can use this IP address. In this way IP addresses can be statically assigned or dynamically shared between all the hosts in an organization. But of course you only have to assign addresses from the prefix given to you. You cannot randomly pick any IP address that you want.

And then every autonomous system has some routers at the border, which are called border routers. What these border routers do is they will announce these IP prefixes to other autonomous systems. For example, if you have some IP prefix, you will tell your ISP that, hey, these are all my IP addresses, please tell everybody. And this ISP will tell everybody that, hey, these, this IP addresses are here. This autonomous system owns these IP addresses. You will tell that information to everybody. That way everybody on the Internet will know who you are.

And whenever somebody wants to send some traffic to you, then they know your location. They know that you are located here and they can send the traffic. So, you advertise your route this way and traffic flows in the opposite direction. You tell somebody, hey, look, this is my IP address, and then that somebody can send back traffic to you using this routing information that you have provided.

So, the purpose of ISPs is basically they propagate this information, and users have these IP prefixes and these ISPs will propagate that information about the end user so that everybody knows about all of these end users. And these ISPs they charge a payment for it. If you have certain IP addresses, they will announce these IP addresses to the rest of the Internet on your behalf, and for that, you will pay your ISP. That is what it means to get service, to get Internet service from an ISP. What does it mean?

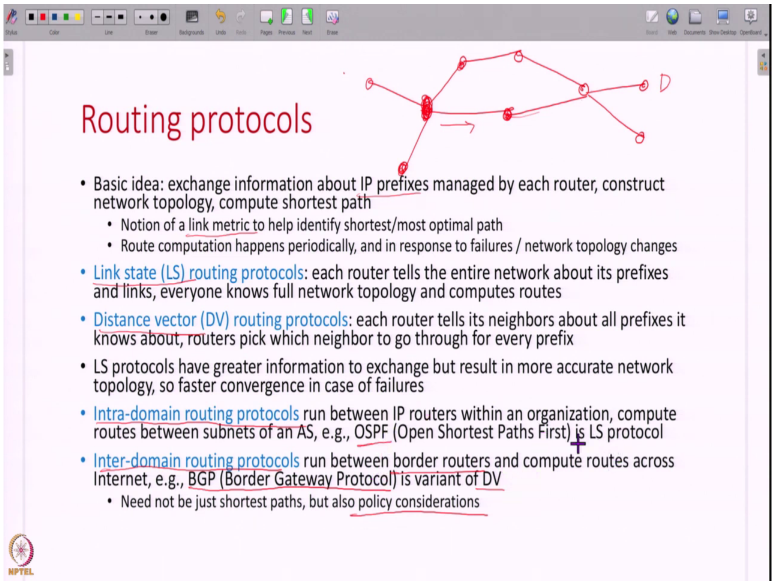
The ISP will come, put a wire into your network and whatever are your IP addresses it will tell everybody else so that everybody knows about you and can send traffic to your IP address through your ISP. And within an autonomous system also you will have internal router. So, an autonomous system can be big and inside that also you will have routers.

So, you will have some border routers which are talking to the outside world and you will also have some internal routers which are also exchanging information because you have split your network into smaller subnets, your prefix into smaller subnets, assigned it to different parts of your network and then these routers are also talking to each other and exchanging information about smaller prefixes. So, in some sense, routing happens in a hierarchical manner.

So, there is this big prefix that you have advertised through your ISP. And ISP will tell everybody else. They know where this big prefixes and all traffic will come to the border router. Then the border router will now send that traffic inside. Then there are smaller routers. You announced a /8, traffic to the /8 prefixes coming here. Then each of the /16s, this border router will distribute. Then each of these routers will then distribute to the various /24 prefixes.

In this way routing and forwarding happens hierarchically on the Internet. You will traffic to a /8 is distributed to the /16s, to the /24s and so on. And this is important for the scalability of the Internet. This hierarchical routing, if everybody is learning about every host on the Internet it gets messy. So, instead you will, the border routers will only announce the bigger prefix to the ISPs and internally whatever routing decisions have to made will be made by internal IP routers.

(Refer Slide Time: 16:03)



Routing protocols

- Basic idea: exchange information about IP prefixes managed by each router, construct network topology, compute shortest path
  - Notion of a link metric to help identify shortest/most optimal path
  - Route computation happens periodically, and in response to failures / network topology changes
- Link state (LS) routing protocols: each router tells the entire network about its prefixes and links, everyone knows full network topology and computes routes
- Distance vector (DV) routing protocols: each router tells its neighbors about all prefixes it knows about, routers pick which neighbor to go through for every prefix
- LS protocols have greater information to exchange but result in more accurate network topology, so faster convergence in case of failures
- Intra-domain routing protocols run between IP routers within an organization, compute routes between subnets of an AS, e.g., OSPF (Open Shortest Paths First) is LS protocol
- Inter-domain routing protocols run between border routers and compute routes across Internet, e.g., BGP (Border Gateway Protocol) is variant of DV
  - Need not be just shortest paths, but also policy considerations

And then all of these routers they basically exchange information about IP prefixes so that you know where each IP prefix is and you can construct this network topology and you can compute the shortest path. So, you have routers exchange information about IP prefixes and also some notion of a link metric to identify this link is expensive to use, this link is shorter, this link is longer, you will have some kind of a link metrics so that you can identify the shortest path.

And all these routers are exchanging information and periodically they will compute the best route to every destination or whenever some failure happens, something happens, your network topology changes, all of these routers are dynamically, constantly exchanging information and re-computing these best routes.

So, there are roughly two philosophies of how you design routing protocols which are called link state routing as well as distance vector routing. So, what is link state routing? Suppose you have a topology like this of multiple routers and then what is link state routing is every router will tell everybody in the network information about all its links. So, this router will say I have so and so all of these prefixes I am managing and I am connected to this guy.

This router will say, these are all my links. This router will say these are all my links. In this way every router will tell the entire network about all of its prefixes and links its information, its story, it will tell everybody in the network. And once everybody knows about everybody else, then everybody can construct this full network topology. And once each router has his full network topology, okay, if I want to go from this point to this point, this is my shortest path, everybody can compute that and accordingly forward packets.

The other type of routing protocols is what are called distance vector routing protocols. That is routers do not tell everybody their information. Instead they will only gossip with their neighbors. So, this router will say, oh, I know how to get to those IP prefixes. Then this router will say, okay, if you know, then I will use you. So, for example, if you have a some destination D over here, this router will say I know how to get to D. This router will also say I know how to get to D and I have a three hop path. This router will say, oh, I have a two hop path to get to D. Then this router will decide.

After listening to the gossip from both these guys, it will decide, okay, maybe I will go like this in order to send a packet to D. In this way, nobody has a full knowledge of the entire network topology. Everybody is only talking to their neighbors. But from what their neighbors tell them they are going to pick which neighbor to use to go to any destination. So, in general, I mean, these are two different ways of designing routing protocols.

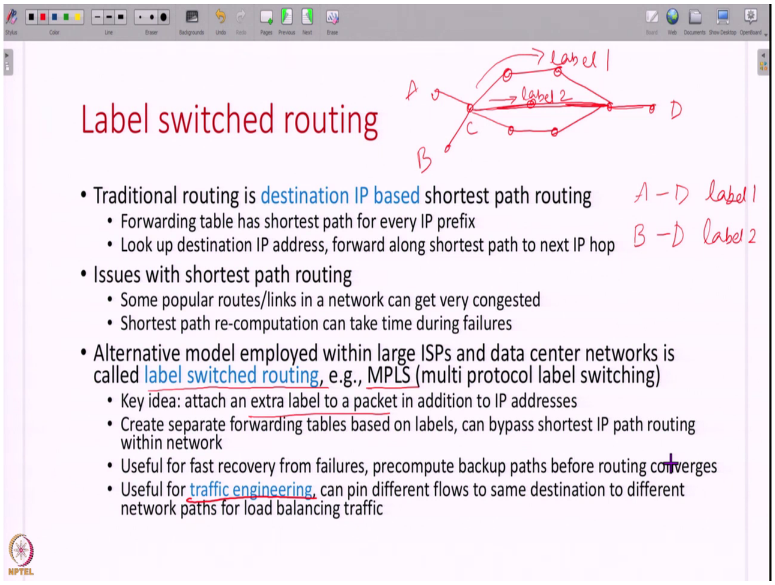
And in general link state protocols are somewhat better because they give you a more accurate picture of the network topology. And therefore in case any failure happens then link state protocols can recover from that failure faster. But in the real world, you have implementations of both these protocols in use. And in a networking course you will study these protocols in more detail. We would not have time to cover them here.

And the other classification of routing protocols is within a domain the routing protocols that are used are called intra-domain routing protocols. And these can be different from what routing protocols are used by border routers, which are inter-domain routing protocols, because you have different considerations.

Inside a network you just want to pick shortest paths. But outside, on the Internet you might also have some policy considerations, this guy paid me more money, this guy did not pay me any money at all. Therefore, I will pick this network. This network is cheaper to use. This network is more expensive to use. You might have other considerations not just the shortest path.

Therefore, you have an inter-domain routing protocol that is very popular today called the BGP, border gateway protocol, that is like a distance vector protocol but slightly different which can help you express all of these constraints also. And it is, that is why it is used for inter-domain routing, whereas within a network you will use a simple protocol like OSPF is a very popular protocol today which is a simple link state shortest path protocol. So, whatever is the routing protocol in use, there are different routing protocols within a network and outside a network and the network operator has to suitably pick a routing protocol based on his or her goals.

(Refer Slide Time: 21:05)



Label switched routing

- Traditional routing is destination IP based shortest path routing
  - Forwarding table has shortest path for every IP prefix
  - Look up destination IP address, forward along shortest path to next IP hop
- Issues with shortest path routing
  - Some popular routes/links in a network can get very congested
  - Shortest path re-computation can take time during failures
- Alternative model employed within large ISPs and data center networks is called label switched routing, e.g., MPLS (multi protocol label switching)
  - Key idea: attach an extra label to a packet in addition to IP addresses
  - Create separate forwarding tables based on labels, can bypass shortest IP path routing within network
  - Useful for fast recovery from failures, precompute backup paths before routing converges
  - Useful for traffic engineering, can pin different flows to same destination to different network paths for load balancing traffic

Now, the other concept I would like to introduce is what is called labels switched routing. That is traditional networks they use destination based routing. Given a destination I will take the shortest path to that destination. But sometimes taking the shortest path may not always be the good idea. What if you know some failure has occurred, the shortest path is congested. Suppose you have a network like this and there is one long path to the destination and there is one very multiple long paths to the destination and one direct path to the destination, and your destination is say here, and you have traffic coming from A, B through some router C going to this destination D.

So, if C will, for every packet it will always pick only the shortest path, look at the IP address, pick the shortest path, then this link will get very congested to go to D. This path will get very congested. When there are other paths that are free, you might have as well split your traffic across all of these paths. But if you are just doing shortest path, destination IP based shortest path routing, you will always pick this path no matter how congested it is.

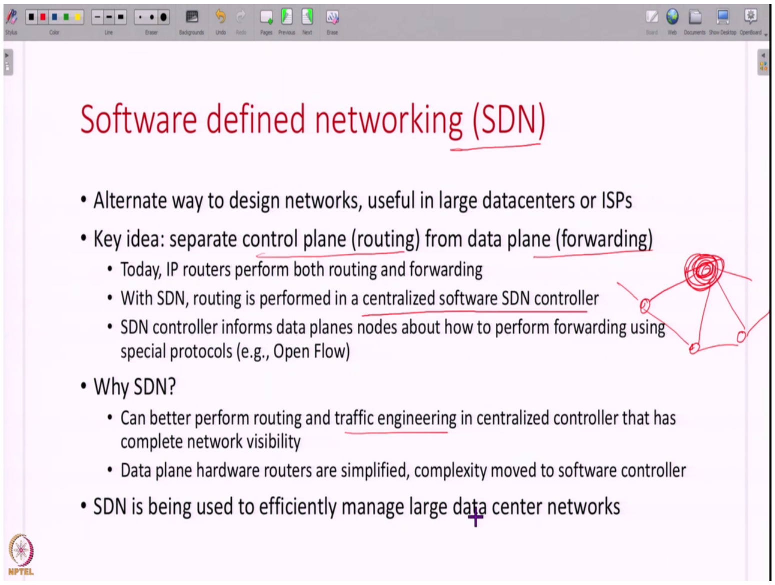
So, an alternate way is to do what is called a label switched routing. That is on every flow, for example, packets coming from A on this A to D path, you assign some label to packets, on this B to D path you assign another label to packets, and you say that traffic with label 1 goes like this and traffic with label 2 goes like this.

So, labels switched routing, so there are many protocols for this, MPLS is the most popular one, with labels switched routing, of course, is a complicated topic, but the high level idea is that you can attach an extra label to a packet and use this label in order to do your forwarding. You create separate forwarding tables based on this label so that you do not have to always do the shortest IP, destination IP based forwarding.

You can assign these labels and then based on your load, if this link is congested, you move your label to here, you move label 1 here. You can shuffle around your labels. You can pin different labels to different paths. And ensure that your network is not congested and it is more optimally used.

So, all of this is broadly called traffic engineering. That is you can pin different flows to the same destination to different paths for better load balancing of traffic. And this label switched routing also helps you recover from failures faster, because you are no longer waiting for the IP based routing to converge. So this is again not for every network, but for large data center networks this can be used in order to efficiently use your network.

(Refer Slide Time: 23:57)



Software defined networking (SDN)

- Alternate way to design networks, useful in large datacenters or ISPs
- Key idea: separate control plane (routing) from data plane (forwarding)
  - Today, IP routers perform both routing and forwarding
  - With SDN, routing is performed in a centralized software SDN controller
  - SDN controller informs data planes nodes about how to perform forwarding using special protocols (e.g., Open Flow)
- Why SDN?
  - Can better perform routing and traffic engineering in centralized controller that has complete network visibility
  - Data plane hardware routers are simplified, complexity moved to software controller
- SDN is being used to efficiently manage large data center networks

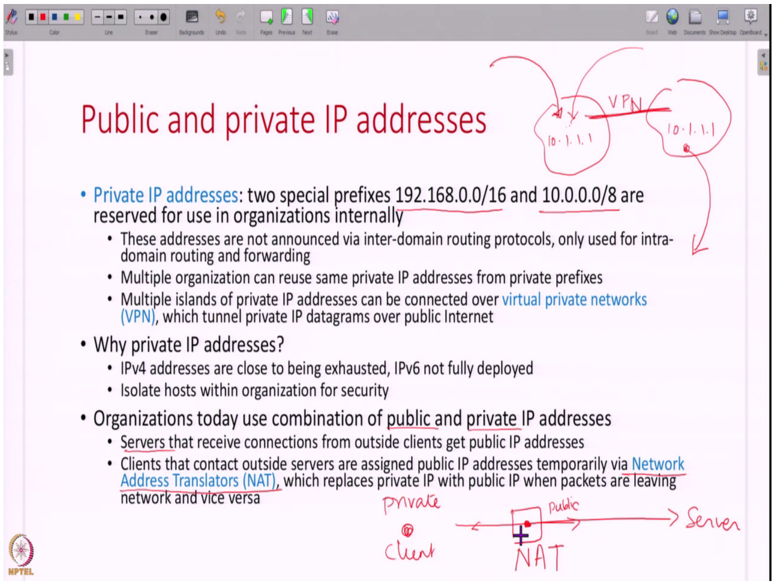
Then the other concept that is being used in data center networks today is what is called software defined networking or SDN. So until now what we have seen is every router does both the control plane that is the routing protocol as well as the forwarding. And this control plane is distributed across different routers.

But instead, what people are designing in data centers today is they are using what is called software defined networking where you separate out the control plane, the routing into a centralized controller, that is there is one centralized controller and there are multiple switches in the data plane that are forwarding traffic and this controller will tell all of these switches what to do.

Use this path. Do this. This is your shortest path. This path is better for load balancing. This path is better for traffic engineering. You can do better routing, better traffic engineering, recover from failures, better, all of that you can do better because you are using a centralized control plane and a distributed data plane.

And so of course it has its own problems. You cannot do one centralized control plane for the entire Internet. Therefore, this is mostly used in data center networks and small scale networks like this where there is a large amount of traffic and efficiency is important. So this is a new idea for how to design networks. That is very different from the traditional ideas that were used in the Internet.

(Refer Slide Time: 25:41)



Public and private IP addresses

- Private IP addresses: two special prefixes 192.168.0.0/16 and 10.0.0.0/8 are reserved for use in organizations internally
  - These addresses are not announced via inter-domain routing protocols, only used for intra-domain routing and forwarding
  - Multiple organization can reuse same private IP addresses from private prefixes
  - Multiple islands of private IP addresses can be connected over virtual private networks (VPN), which tunnel private IP datagrams over public Internet
- Why private IP addresses?
  - IPv4 addresses are close to being exhausted, IPv6 not fully deployed
  - Isolate hosts within organization for security
- Organizations today use combination of public and private IP addresses
  - Servers that receive connections from outside clients get public IP addresses
  - Clients that contact outside servers are assigned public IP addresses temporarily via Network Address Translators (NAT), which replaces private IP with public IP when packets are leaving network and vice versa

So, the one other concept that is important when we study real life computer systems is the concept of private IP addresses. So, there are, of course, there are 2 to the 32 different IP addresses, IPv4 addresses possible. Out of all of these IPv4 addresses two prefixes, two groups of IP addresses, this one and this one, are reserved for use within organizations internally. These are called private IP addresses. So, what does that mean?

These IP addresses are used only within an organization only for intra-domain routing, and are never announced in inter-domain routing. And therefore, multiple organizations can reuse the same IP addresses. In this organization also you can assign an address 10.1.1.1. In this organization also you can assign this address. So, these are like nicknames. These private IP addresses do not have to be globally unique. They are only limited to one organization and announced only within that organization.

So, why do we use private IP addresses, because IP addresses are getting exhausted. We may not have enough IP addresses for everybody. So, therefore, that is one reason. The other reason is that you want to isolate the hosts in an organization. If somebody has a private IP address, nobody from outside can talk to this host. Why, because these private IP addresses are not shared with anybody. It is like your secret, your secret nickname. Therefore, it also gives you a certain amount of security.

And if you have two different networks with private IP addresses, you can also connect them over using VPN software, virtual private network software. So, suppose you are logged in, your college has a private network and you want to log into your college from, college network from home, then you use a VPN. That is a way to connect these private networks to each other over the Internet.

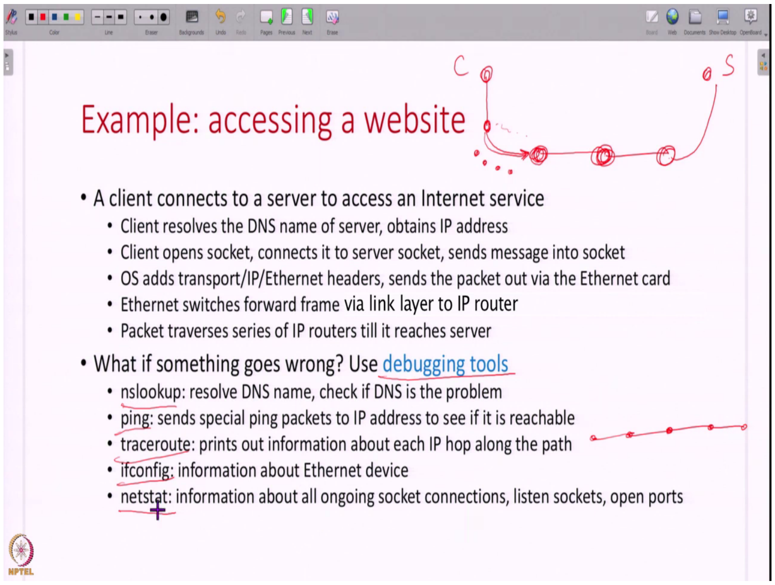
And so organizations today usually use a combination of both public and private IP addresses. Note that you cannot just live with private IP addresses. You need some public IP addresses, for example, for things like servers, which are connected to outside client. Somebody outside has to send you a request, send you a connection request. Then you need a public IP address that has to be announced via DNS and has to be known to the client.

But if you are only talking to people within your organization, then you do not need a public IP address. You can simply use private IP addresses. It is like in your home all of you can address each other using nicknames. But when somebody outside has to talk to you, they have to use your publicly known name. That is the concept of public IP addresses and private IP addresses.

And if there is a client inside that has to talk to an external server then also you need a temporary public IP address. Clients also need a public IP address. Why, because you need the source address also put in your IP datagram and that has to be a public address. So, temporarily IP addresses that are public IP addresses that are needed are assigned via what are called NATs or network address translator.

That is if you have a client inside a network that has to talk to some other server outside and this client is using a private IP address, then this network address translator or the NAT in your network will temporarily for this connection give you a public IP address and rewrite the header in your packet and on the reverse direction it will remove the public address and give you a private IP datagram. In this way this network address translators will just, with a small number of public IP addresses they can manage large networks.

(Refer Slide Time: 29:25)



Example: accessing a website

- A client connects to a server to access an Internet service
  - Client resolves the DNS name of server, obtains IP address
  - Client opens socket, connects it to server socket, sends message into socket
  - OS adds transport/IP/Ethernet headers, sends the packet out via the Ethernet card
  - Ethernet switches forward frame via link layer to IP router
  - Packet traverses series of IP routers till it reaches server
- What if something goes wrong? Use debugging tools
  - nslookup: resolve DNS name, check if DNS is the problem
  - ping: sends special ping packets to IP address to see if it is reachable
  - traceroute: prints out information about each IP hop along the path
  - ifconfig: information about Ethernet device
  - netstat: information about all ongoing socket connections, listen sockets, open ports

So, now putting all of these concepts together, let us understand, you have a client and you have a server on the Internet, how do they talk to each other. So, this client will use DNS, get the IP address of the server, open a socket here, write the message to the server into the socket, then the OS will add all the headers, the transport layer processing all of that, send the packet out, the packet will go through a series of IP routers, and finally, reach the server.

And how do you get to your IP router? On the way you will have multiple Ethernet switches or if Ethernet is your link layer technology or some other wire or over a wireless network you will send the packet to this IP router. This IP router might use another link layer technology to go to this other IP router.

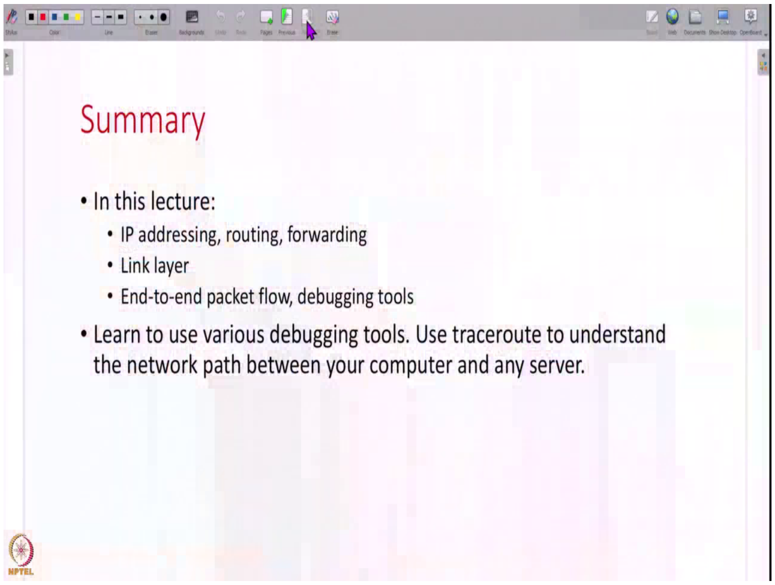
In this way across different, different links, you will jump from one IP hop to the other and finally reach your server. So, there are different layers here. You have the application layer, the transport layer, IP layer, link layer, and finally, the physical layer, all of these working together to establish this end to end communication. And of course, there are many steps here. Many things can go wrong. Your connection can break at any point, which is why we have a bunch of debugging tools available to us to help us debug the working of networks.

For example, there are tools that will help you resolve a DNS name to check if DNS is the problem. You can do something like ping. Ping is a special, packets are sent to any IP address to see is the IP path working. You can do traceroute, where you can actually find out, if you run the traceroute tool, it will tell you the locations of all these IP routers along your path.

You can find out information about your link layer device itself using commands like ifconfig. You can find out information about all the socket communications going on using tools like netstat. In this way, for each layer, there are separate debugging tools that will help you understand how that particular layer is working.

So, in this lecture, I have briefly covered how IP addressing, routing, forwarding works and how the link layer works. So, I did not go into a lot of details into any of these because covering them in a lot of detail requires a complete course. But I have given you enough to help you understand the end to end packet flow in a computer system. And if something goes wrong, how do you go about debugging this?

(Refer Slide Time: 31:51)



Summary

- In this lecture:
  - IP addressing, routing, forwarding
  - Link layer
  - End-to-end packet flow, debugging tools
- Learn to use various debugging tools. Use traceroute to understand the network path between your computer and any server.

So, please try to use these tools that we have studied on your own. This will help you understand these concepts better. Thank you all that is all I have for this lecture. And let us continue this discussion in the next lecture.