UNIT - 4

4.1 Domain Name System (DNS)

One of the most important components of the application layer is the Domain Name System (DNS) server. DNS is a distributed hierarchical and global directory that translates machine or domain names to numerical IP addresses. DNS can run over either UDP or TCP. Some of the information-processing functions a DNS server handles are:

1. Finding the address of a particular host
2. Delegating a subtree of server names to another server
3. Denoting the start of the subtree that contains cache and configuration parameters, and giving corresponding addresses
4. Naming a host that processes incoming mail for the designated target
5. Finding the host type and the operating system information
6. Finding an alias for the real name of a host
7. Mapping IP addresses to host names

4.1.1 Domain Name Space

Any entity in the TCP/IP environment is identified by an IP address, which thereby identifies the connection of the corresponding host to the Internet. An IP address can also be assigned a domain name. Unique domain names assigned to hosts must be selected from a name space and are generally organized in a hierarchical fashion.

Domain names are defined in a tree-based structure with the root at the top, as shown in Figure 5.2. A tree is structured with a maximum of 128 levels, starting at level 0 (root). Each level consists of nodes. A node on a tree is identified by a label, with a string of up to 63 characters, except for the root label, which has empty string.

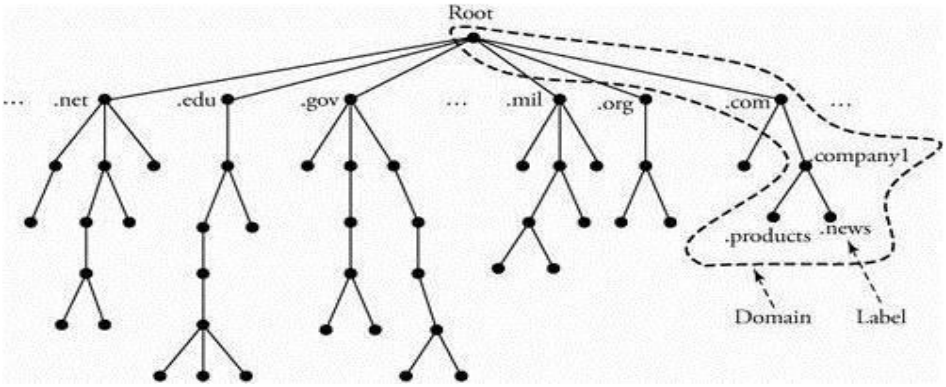


Figure 5.2. Hierarchy of domain name space, labels, and domain names

The last label of a domain name expresses the type of organization; other parts of the domain name indicate the hierarchy of the departments within the organization. Thus, an organization can add any suffix or prefix to its name to define its host or resources. A domain name is a sequence of labels separated by dots and is read from the

node up to the root. For example, moving from right to left, we can parse as follows: domain name news.company1.com, a commercial organization (.com) and the "news" section of "company1" (news.company1). Domain names can also be partial. For example, company1.com is a partial domain name.

Domain-Name Servers

The domain name space is divided into sub domains, and each domain or sub domain is assigned a domain name server. This way, we can form a hierarchy of servers, as shown in Figure 5.3, just as we did for the hierarchy of domain names. A domain name server has a database consisting of all the information for every node under that domain.

4.1.2 Name/Address Mapping

DNS operates based on the client/server application. Any client host can send an IP address to a domain name server to be mapped to a domain name. Each host that needs to map an address to a name or vice versa should access the closest DNS server with its request.

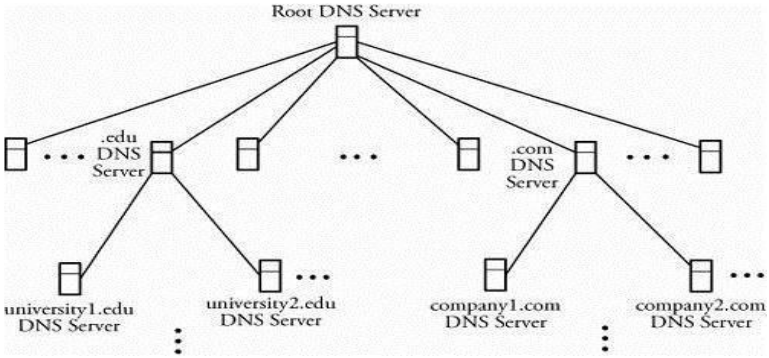


Figure 5.3. Hierarchy of DNS domain name servers

Mapping can be of either recursive or iterative. In recursive mapping (Figure 5.4), the client host makes the request to its corresponding DNS server. The DNS server is responsible for finding the answer recursively. The requesting client host asks for the answer through its local DNS server, news.company1.com Finally, .com server sends the query to the local DNS server of the requested place, as dns.company2.com, and finds the answer. The answer to a query in this method is routed back to the origin, as shown in the figure. The local DNS server of the requested place is called the authoritative server and adds information to the mapping, called time to live (TTL).

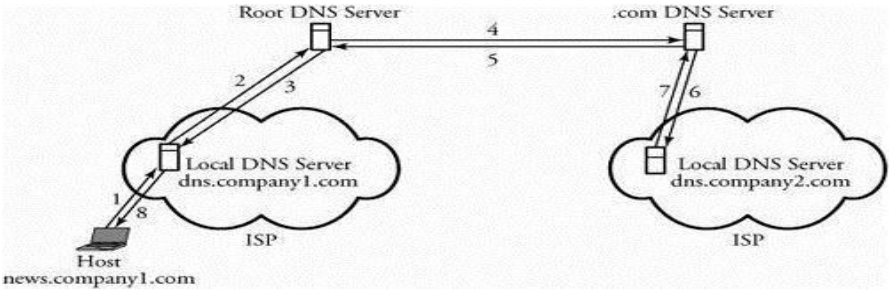


Figure 5.4. Recursive mapping

In the iterative approach, the mapping function is as shown in Figure 5.5. In this case, if it does not have the name to provide, the server returns to the client host.

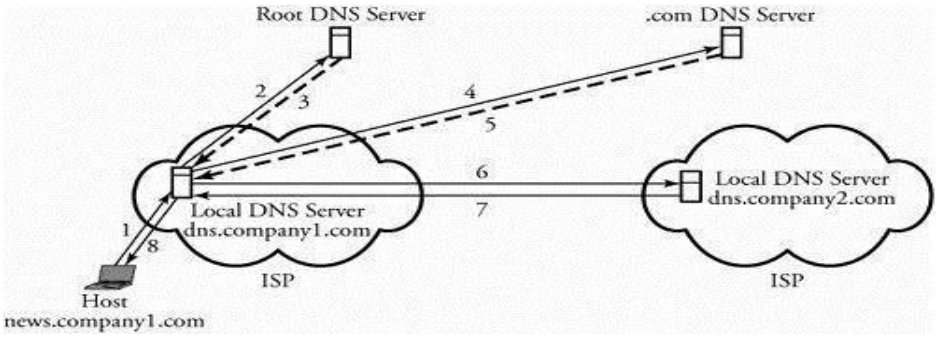


Figure 5.5. Iterative mapping

In Figure 5.5, the news.company1.com host sends the query to its own local DNS server, dns.company1.comthus trying the root DNS server firstand then tries .com server, finally ending up with the local DNS server of the requested place: dns.company2.com.

4.1.3 DNS Message Format

DNS communication is made possible through query and reply messages. Both message types have the 12-byte header format shown in Figure 5.6.

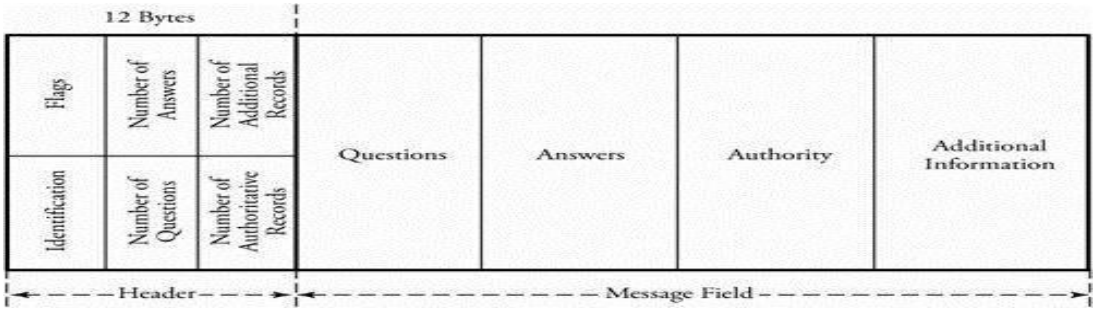


Figure 5.6. DNS message format

The header has six fields as follows. A client uses the identification field to match the reply with the query. This field may appear with a different number each time a client transmits a query. The server copies this number in its reply. The flags field contains subfields that represent the type of the message, such as the type of answer requested or requested DNS recursive or iterative mapping. The number of questions field indicates how many queries are in the question portion of the message. The number of answers shows how many answers are in the answer field. For the query message, this field contains all zeros. The number of authoritative records field consists of the number of authoritative records in the authority portion of a reply message. Similarly, this field is filled by zeros for a query message. Finally, the number of additional records field records are in the additional information portion of a reply message and is similarly filled by zeros in a query message.

4.2 Remote Login Protocols

A client/server model can create a mechanism that allows a user to establish a session on the remote machine and then run its applications. This application is known as remote login. This can be done by a client/server application program for the desired service. Two remote login protocols are TELNET and SSH.

4.2.1 TELNET Protocol

TELNET (terminal network) is a TCP/IP standard for establishing a connection to a remote system. TELNET allows a user to log in to a remote machine across the Internet by first making a TCP connection and then pass the detail of the application from the user to the remote machine.

Logging to Remote Servers

With TELNET, an application program on the user's machine becomes the client. The user's keyboard and its monitor also attach directly to the remote server. The remote- logging operation is based on timesharing, whereby an authorized user has a login name and a password. TELNET has the following properties.

- Client programs are built to use the standard client/server interfaces without knowing the details of server programs.
- A client and a server can negotiate data format options.
- Once a connection is established through TELNET, both ends of the connection are treated symmetrically.

When a user logs in to a remote server, the client's terminal driver accepts the keystrokes and interprets them as characters by its operating system. Characters are typically transformed to a universal character set called network virtual terminal (NVT), which uses 7-bit USASCII representation for data. The client then establishes a TCP connection to the server. Texts in the NVT format are transmitted using a TCP session and are delivered to the operating system of the remote server. The server converts the characters back from NVT to the local client machine's format.

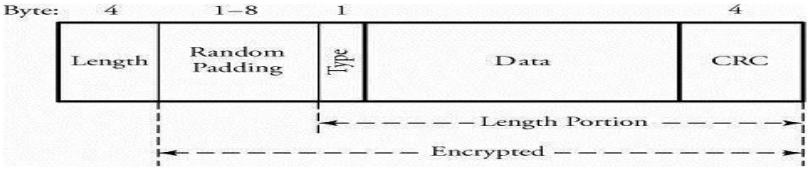
4.2.2 Secure Shell (SSH) Protocol

Secure Shell (SSH), another remote login protocol, is based on UNIX programs. SSH uses TCP for communications but is more powerful and flexible than TELNET and allows the user to more easily execute a single command on a remote client. SSH has the following advantages over TELNET.

- SSH provides a secure communication by encrypting and authenticating messages.
- SSH provides several additional data transfers over the same connection by multiplexing multiple channels that are used for remote login.

SSH security is implemented by using public-key encryption between the client and remote servers. When a user establishes a connection to a remote server, the data being transmitted remains confidential even if an intruder obtains a copy of the packets sent over an SSH connection. SSH also implements an authentication process on messages so that a server can find out and verify the host attempting to form a connection. Normally, SSH requires users to enter a private password.

The advantage of port forwarding is that application data can be passed between two sites the client and the second server without requiring a second client and server the first server as a client and the second server. Figure 5.7 shows the format of an SSH packet.



- Length indicates the size of the packet, not including the length field or the variable-length random padding field that follows it.
- Padding causes an intrusion to be more difficult.
- Type identifies the type of message.
- CRC, or cyclic redundancy check, is an error-detection field.

4.3 Electronic Mail (E-mail)

4.3.1 Simple Mail Transfer Protocol (SMTP) and E-mail

The Simple Mail Transfer Protocol (SMTP) plays a major role in transferring Internet electronic mail. This protocol transfers electronic mail (e-mail) from the mail server of a source to the mail servers of destinations. SMTP is older than the Hypertext Transfer Protocol (HTTP), the Web communication protocol, and imposes certain restrictions, such as limits on the size of e-mail content.

In Figure 5.8, user 1 is in a residential area, has an Internet service provider (ISP), and is sending an e-mail to user 2, working in an organization. Suppose that the mail servers are isp.com and organization.com, respectively. Thus, user 1 and user 2 have e-mail addresses of user1@isp.com and user2@organization.com, respectively. The procedure for an e-mail exchange between user 1 and user 2 is as follows.

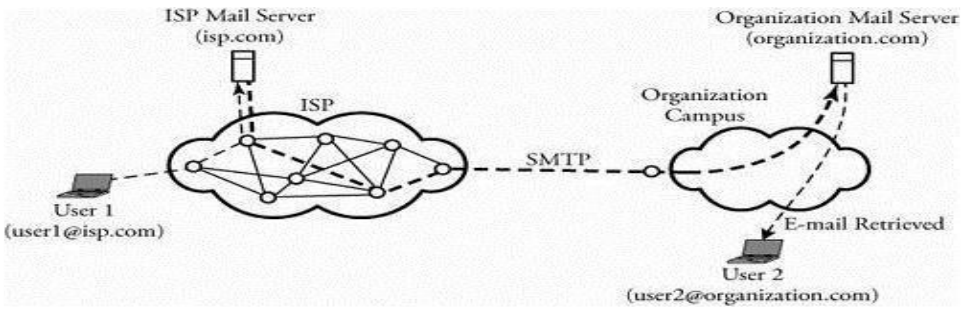


Figure 5.8. Two users exchanging e-mail through SMTP

Begin SMTP between Two Users

1. User 1 provides user 2's e-mail address (user2@organization.com) and composes its message.
2. User 1 sends the message to its mail server (isp.com).
3. Server isp.com places the message in its queue.
4. SMTP on user 1's mail server notices the message in the queue and opens a TCP connection with the organization mail server (organization.com).
5. Initial SMTP handshaking takes place between the two servers.
6. The message is sent to organization.com's mail server, using the established TCP connection.
7. User 2's mail server receives the message and then puts it in user 2's mailbox, ready to be retrieved by user 2.

4.4 File Transfer and FTP

File transfer is another computer networking application. It is always essential that files and information geographically distributed over different locations be shared among the members of a working group. In a certain application, files are typically saved in a server. A user then uses a file transfer protocol to access the server and transfer the desired file. Two file transfer protocols are FTP and SCP.

4.4.1. File Transfer Protocol (FTP)

File Transfer Protocol (FTP) is part of the TCP/IP suite and is very similar to TELNET. Both FTP and TELNET are built on the client/server paradigm, and both allow a user to establish a remote connection. However, TELNET provides a broader access to a user, whereas FTP allows access only to certain files. The essence of this protocol is as follows.

Begin File Transfer Protocol

1. A user requests a connection to a remote server.
2. The user waits for an acknowledgment.
3. Once connected, the user must enter a user ID, followed by a password.
4. The connection is established over a TCP session.
5. The desired file is transferred.
6. The user closes the FTP connection.

FTP can also run through a Web browser.

4.4.2. Secure Copy Protocol (SCP)

The Secure Copy Protocol (SCP) is similar to TELNET but is secure. Incorporated in the SCP structure are a number of encryption and authentication features that are similar to those in SSH. Also similar is the exchange of commands between local and remote hosts. SCP commands automatically prompt the user for the password

information when it is time to access a remote machine. SCP cannot handle file transfer between machines of significantly different architectures.

4.5 World Wide Web (WWW) and HTTP

Application-layer software is the intelligence built for end servers. The World Wide Web (WWW), or simply Web, is a global network of servers linked by a common protocol allowing access to all connected hypertext resources. When a client host requests an object, a Web server responds by sending the requested object through browsing tools. A browser is a user agent displaying the requested Web page. The Hyper Text Transfer Protocol (HTTP) transfers that page at the application layer. HTTP uses TCP rather than UDP, since reliability of delivery is important for Web pages with text. The TCP connection-establishment delay in HTTP is one of the main contributing delay factors associated with downloading Web documents.

HTTP is based on the client/server idea, having a client and a server program, both of which can be executed on different end systems. The communication is carried out through an exchange of HTTP messages. This protocol specifies the structure of these messages. For example, HTTP defines how a pair of client/server hosts should exchange messages. In this context, a Web page consists of files, such as Hypertext Mark-up Language (HTML) file or an image that can be addressed by a single uniform resource locator (URL). A URL is a global address of an HTML document and has two parts. The first part indicates what protocol is used, and the second part determines the IP address of the associated resource.

4.5.1. Web Caching (Proxy Server)

An HTTP request from a user is first directed to the network proxy server, or Web cache. Once configured by the network, a browser's request for an object is directed to the Web cache, which must contain updated copies of all objects in its defined proximity. The main reason for Web caching is to reduce the response time for a user request. This benefit is much more obvious when the bandwidth to a requested server is limited because of traffic at certain hours of the day.

Normally, each organization or ISP should have its own cache providing a high-speed link to its users. Consequently, it is to users' advantages that this rapid method of finding objects be available. This method of Internet access also reduces traffic on an organization's access link to the Internet. The details of Web caching are as follows:

Begin Web Caching Algorithm

1. The source browser makes a TCP connection to the Web cache.
2. The user browser transmits its HTTP request to the Web cache.
3. If it has a copy of the requested object, the Web cache forwards the object to the user browser.

Otherwise the Web cache establishes a TCP connection to the requested server and asks for the object. Once it receives the requested object, the Web cache stores a copy of it and forwards another copy to the requesting user browser over the existing TCP connection.

Figure 5.9 shows three Internet service providers (ISPs). A user in ISP domain 3 is browsing to find and watch an object named http://www.filmmaker.com in ISP domain 1. The request for this object is directed to the Web cache, shown by dashed lines. In this example, the Web cache has no record of the requested object and therefore is establishing another TCP connection to update its record.

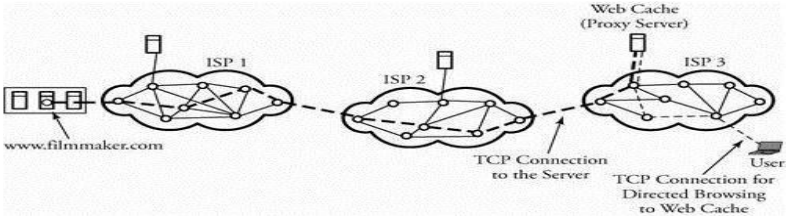


Figure 5.9. A user's browser requesting an object through the Web cache

4.6 Multiprotocol Label Switching (MPLS):

Multiprotocol label switching (MPLS) improves the overall performance and delay characteristics of the Internet. MPLS transmission is a special case of tunneling and is an efficient routing mechanism. Its connection-oriented forwarding mechanism, together with layer 2 label-based lookups, enables traffic engineering to implement peer-to-peer VPNs effectively.

MPLS adds some traditional layer 2 capabilities and services, such as traffic engineering, to the IP layer. The separation of the MPLS control and forwarding components has led to multilayer, multiprotocol interoperability between layer 2 and layer 3 protocols. MPLS uses a small label or stack of labels appended to packets and typically makes efficient routing decisions. Another benefit is flexibility in merging IP-based networks with fast-switching capabilities. This technology adds new capabilities to IP-based networks:

- Connection-oriented QoS support
- Traffic engineering
- VPN support
- Multiprotocol support

Traditional IP routing has several limitations, ranging from scalability issues to poor support for traffic engineering. The IP backbone also presents a poor integration with layer 2 existing in large service provider networks. For example, a VPN must use a service provider's IP network and build a private network and run its own traffic shielded from prying eyes. In this case, VPN membership may not be well engineered in ordinary IP networks and can therefore result in an inefficient establishment of tunnels.

MPLS network architectures also support other applications, such as IP multicast routing and QoS extensions. The power of MPLS lies in the number of applications made possible with simple label switching,

ranging from traffic engineering to peer-to-peer VPNs. One of the major advantages of MPLS is integration of the routing and switching layers. The development of the label- switched protocol running over all the existing layer 2 and layer 3 architectures is a major networking development.

4.6.1. MPLS Operation

MPLS is based on the assignment of labels to packets. Assigning labels to each packet makes a label-swapping scheme perform its routing process much more efficiently. An MPLS network consists of nodes called label switch routers (LSR). An LSR switches labeled packets according to particular switching tables. An LSR has two distinct functional components: a control component and a forwarding component. The control component uses routing protocols, such as OSPF and the border gateway protocol (BGP). The control component also facilitates the exchange of information with other LSRs to build and maintain the forwarding table.

A label is a header used by an LSR to forward packets. The header format depends on the network characteristics. LSRs read only labels and do not engage in the network-layer packet headers. One key to the scalability of MPLS is that labels have only local significance between two devices that communicate. When a packet arrives, the forwarding component uses the label of the packet as an index to search the forwarding table for a match. The forwarding component then directs the packet from the input interface to the output interface through the switching fabric.

MPSL Packet Format

MPLS uses label stacking to become capable of multilevel hierarchical routing. A label enables the network to perform faster by using smaller forwarding tables, a property that ensures a convenient scalability of the network. Figure 6.6 shows the MPLS header encapsulation for an IP packet. An MPLS label is a 32-bit field consisting of several fields as follows.

- Label value is a 20-bit field label and is significant only locally.
- Exp is a 3-bit field reserved for future experimental use.
- S is set to 1 for the oldest entry in the stack and to 0 for all other entries.
- Time to live is an 8-bit field used to encode a hop-count value to prevent packets from looping forever in the network.

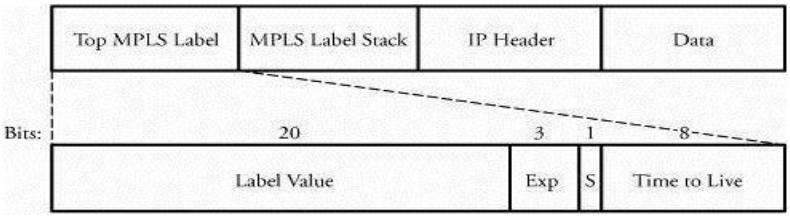


Figure 6.6. MPLS header encapsulation for an IP packet

4.6.2. Routing in MPLS Domains

Figure 6.7 shows the label-switching paradigm in an MPLS network. An ingress LSR is an edge device that performs the initial packet processing and classification and applies the first label. An ingress LSR creates a new label. A core LSR swaps the incoming label with a corresponding next-hop label found from a forwarding table. At the other end of the network, another edge router, the egress LSR, is an outbound edge router and pops the label from the packet. It should be noted that multiple labels may be attached to a packet, forming a stack of labels. Label stacking enables multilevel hierarchical routing.

For example, BGP labels are used for higher-level hierarchical packet forwarding from one BGP speaker to the other, whereas Interior Gateway Protocol (IGP) labels are used for packet forwarding within an autonomous system. Only the label at the top of the stack determines the forwarding decision.

Once an IP packet enters an MPLS domain, the ingress LSR processes its header information and maps that packet to a forward equivalence class (FEC). At this point, a label switch path (LSP) through the network must be defined, and the QoS parameters along that path must be established. The QoS parameters define how many resources are to be used for the path and what queueing and discarding policy are to be used. For these functions, two protocols are used to exchange necessary information among routers: An intradomain routing protocol, such as OSPF, is used to exchange routing information, and the Label Distribution Protocol (LDP) assigns labels.

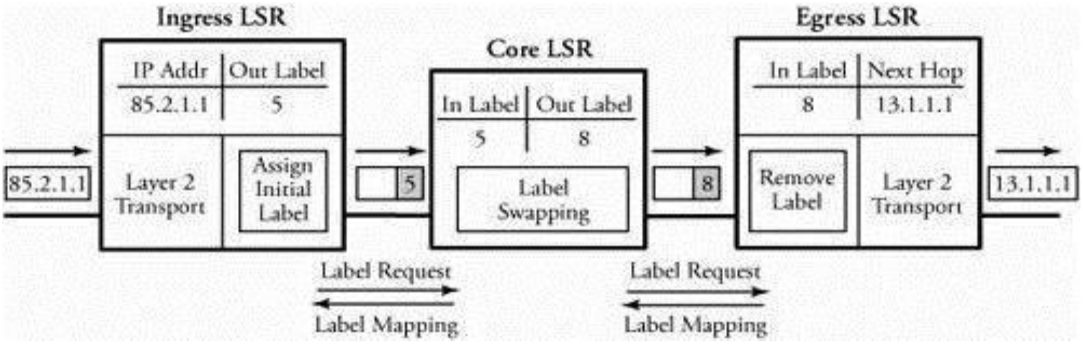


Figure 6.5. Multiple layer 2 switching example in MPLS

At the end of the process, the router appends an appropriate label for FEC purposes and forwards the packet through. Packet forwarding at the core LSR is based on a label-swapping mechanism. Once it receives a labeled packet, the core LSR reads the label as an index to search in the incoming label map table for the corresponding next-hop label. The label in the MPLS header is swapped with the out-label and sent on the next hop.

This method of packet forwarding simplifies the routing process by replacing the longest-prefix match of IP routing with simple short-label exact-match forwarding. The real benefit of this method is that instead of processing IP headers for forwarding packets, routers process a short label. Once a packet arrives at the egress LSR, its MPLS header is decapsulated, and the stripped packet is routed to its destination.

In summary, an MPLS domain has three label manipulation instructions: An ingress LSR creates a new label and pushes it to the label stack of a packet, a core LSR swaps the incoming label with a corresponding next-hop label found from the forwarding table, and an egress LSR (outbound edge router) pops a label from the label stack. Only the label at the top of the stack determines the forwarding decision. The egress LSR strips the label, reads the IP packet header, and forwards the packet to its final destination.

4.6.3. Tunneling and Use of FEC

In an MPLS operation, any traffic is grouped into FECs. FEC implies that a group of IP packets are forwarded in the same manner for example, over the same path or with the same forwarding treatment. A packet can be mapped to a particular FEC, based on the following criteria:

- Source and/or destination IP address or IP network addresses
- TCP/UDP port numbers
- Class of service
- Applications

As mentioned earlier, labels have only local significance. This fact removes a considerable amount of the network-management burden. An MPLS packet may carry as many labels as required by a network sender. The process of labeled packets can always be performed based on the top label. The feature of label stack allows the aggregation of LSPs into a single LSP for a portion of the route, creating an MPLS tunnel. Figure 6.8 shows an IP packet moving through an MPLS domain. When the labeled packet reaches the ingress LSR, each incoming IP packet is analyzed and classified into different FECs. This traffic- classification scheme provides the capability to partition the traffic for service differentiation.

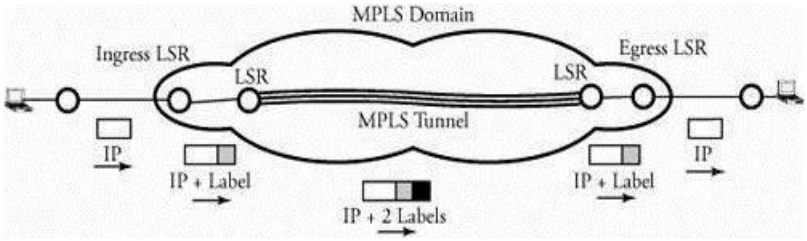


Figure 6.8. An IP packet labeled in an MPLS domain and tunneled to reach the other end of the domain

Route selection can be done either hop by hop or by explicit routing. With hop- byhop routing, each LSR can independently choose the next hop for each FEC. Hop-byhop routing does not support traffic engineering, owing to limited available resources. Explicit routing can provide all the benefits of traffic engineering. With explicit routing, a single LSR determines the LSP for a given FEC. For explicit routing, LSRs in the LSP are identified, whereas in an explicit routing, only some of the LSRs in an LSP are specified.

With the introduction of constraint-based routing, FEC can segregate the traffic into different levels of QoS, each with different service constraints, to support a variety of services, such as latency-based voice traffic and

security-based VPN. At the beginning of the tunnel, an LSR assigns the same label to packets from a number of LSPs by pushing the label onto each packet's stack. At the other side of the tunnel, another LSR pops the top element from the label stack, revealing the inner label.

4.7 Virtual Private Networks:

A virtual private network (VPN) is a data network having connections that make use of public networking facilities. The (VPN) part of public network is set up "virtually" by a private-sector entity to provide public networking services to small entities. With the globalization of businesses, many companies have facilities across the world and use VPNs to maintain fast, secure, and reliable communications across their branches.

VPNs are deployed with privacy through the use of a tunneling protocol and security procedures. Figure 16.1 shows two organizations, 1 and 3, connected through their corresponding routers, forming a tunnel in the public network, such as the Internet. Such a structure gives both private organizations the same capabilities they have on their own networks but at much lower cost. They can do this by using the shared public infrastructure. Creating a VPN benefits an organization benefits by providing

- Extended geographical communication
- Reduced operational cost
- Enhanced organizational management
- Enhanced network management with simplified local area networks
- Improved productivity and globalization

But since each user has no control over wires and routers, one of the issues with the Internet is still its lack of security, especially when a tunnel is exposed to the public. Thus, VPNs remain susceptible to security issues when they try to connect between two private networks using a public resource. The challenge in making a practical VPN, therefore, is finding the best security for it. Before discussing VPN security, we focus on types of VPNs. There are two types of VPNs each determined by its method of tunneling, remote-access and site-to- site. We will explain these two approaches in the next two sections.

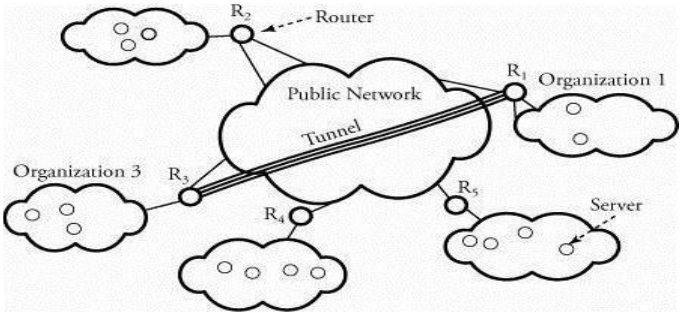


Figure 6.6. Two organizations connected through a tunnel using public Facilities

4.7.1. Remote-Access VPN

Remote-access VPN is a user-to-LAN connection that an organization uses to connect its users to a private network from various remote locations. Large remote-access VPNs are normally outsourced to an Internet service provider to set up a network-access server. Other users, working off campus, can then reach the network-access server and use the VPN software to access the corporate network. Remote-access VPNs allow encrypted connections between an organization's private network and remote users through a third-party service provider. Tunneling in a remote-access VPN uses mainly the Point-to-Point Protocol (PPP). PPP is the carrier for other Internet protocols when communicating over the network between a host computer and a remote point.

Besides IPsec, other types of protocols associated with PPP are L2F, PPTP, and L2TP. The Layer 2 Forwarding (L2F) protocol uses the authentication scheme supported by PPP. The Point-to-Point Tunneling Protocol (PPTP) supports 40-bit and 128-bit encryption and uses the authentication scheme supported by PPP. The Layer 2 Tunneling Protocol (L2TP) combines features of both PPTP and L2F.

4.7.2. Site-to-Site VPN

By using effective security techniques, an organization can connect multiple fixed sites over a public network. Site-to-site VPNs can be classified as either intranets or extranets.

- Intranet VPNs connect an organization's remote-site LANs into a single private network.
- Extranet VPNs allow two organizations to work in a shared environment through a tunnel built to connect their LANs.

Figure 6.4 shows the three types VPNs discussed so far. Organization 1's main campus and branch campus are connected through an intranet VPN tunnel. The main campus can also be connected to organization 2 through an extranet VPN tunnel. The employees of organization 1 can also access their corporation through a remote-access VPN. Each remote-access member must communicate in a secure medium. The main benefit of using a VPN is scalability with a reasonable cost. However, the physical and virtual distances of two communicating organizations have a great impact on the overall cost of building a VPN.

In a site-to-site VPN, generic routing encapsulation (GRE) is normally the encapsulating protocol. GRE provides the framework for the encapsulation over an IP-based protocol. IPsec in tunnel mode is sometimes used as the encapsulating protocol. IPsec works well on both remote-access and site-to-site VPNs but must be supported at both tunnel interfaces. The Layer 2 Tunneling Protocol (L2TP) can be used in site-to-site VPNs. L2TP fully supports IPsec regulations and can be used as a tunneling protocol for remote-access VPNs.

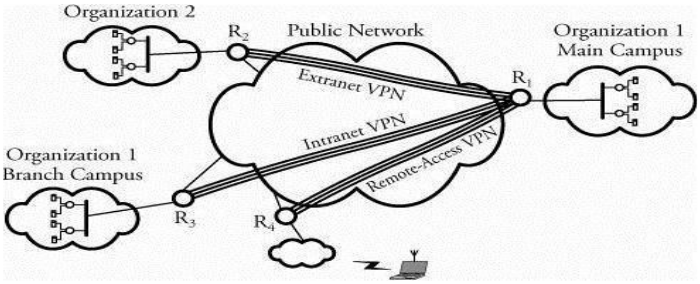


Figure 6.4. Three types of VPNs to and from a headquarter organization

4.7.3. Tunneling and Point-to-Point Protocol (PPP)

A tunnel is a connection that forms a virtual network on top of a physical network. In computer networking, a tunnel resembles a telephone line in a public switched telephone network. VPNs typically rely on tunneling to create a private network that reaches across a public network. Tunneling is a process of encapsulating packets and sending them over the public network. Employees who are located outside an organization's main building can use point-to-point connections to create tunnels through the Internet. Since tunneling connections normally run over the Internet, they need to be secure. A tunnel is a relatively inexpensive connection, since it uses the Internet as its primary form of communication. Besides Internet protocols, tunneling requires two other types of protocols:

1. Carrier protocols, through which information travels over the public network
2. Encapsulating protocols, through which data is wrapped, encapsulated, and secured One of the amazing implications of VPNs is that packets that use a protocol not supported on the Internet, such as NetBeui, can be placed inside an IP packet and sent safely over the Internet.

VPNs can put a packet that uses a nonroutable IP address inside a packet to extend a private network over the Internet.

Consider the two LANs of the two organizations shown in Figure 6.7. We want to connect these two LANs through the Internet by using tunnels. Assume that the two LANs, as organization 1 and organization 2, want to use their own customized networking protocols, denoted by x, using connectionless datagram IP services. The IP resources can be at the scale of the Internet. Therefore, x-type packets cannot run over the Internet directly. The IP gateway R1 listens for x- type packets on organization 1, encapsulates x-type packets in the transport-layer UDP datagrams, and transmits them over the Internet to R2. When R2 receives the encapsulates x packets, it decapsulates and feeds them into organization 2. This connectionin fact, a tunnel made through the Internet resembles a direct physical link between the two LANs.

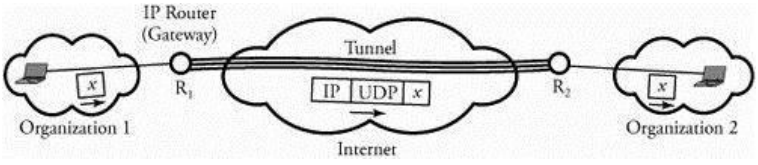


Figure 6.7. A customized protocol packet tunneling through the Internet