IT430 Assignment No 01 Solution Spring 2013 (25-04-2013)

www.vchowk.com

Internet layers- TCP/IP stack
Internet communication mo\del (TCP/IP Stack) proposes a 4 layer architecture. Following are the names of internet layers that map with the OSI model's layer:
Application layer is equivalent to OSI model's Application Presentation and Session layer
Host-to-Host Layer is equivalent to OSI model's Transport layer
Internet layer is equivalent to OSI model's Network layer
Network Access layer is equivalent to OSI model's data link layer
What are the protocols that will be responsible in delivery of e-mail from the source to destination
There are three protocols commonly used for emails. Simple Mail Transfer Protocol (SMTP) is use fro sending mail messages between servers. IN other words it is used for message uploads. Post Office Protocol 3 (POP3) or internet Message Access Protocol (IMAP) can be used to retrieve messages. They should also be configured with SMTP. POP is use to download email to client machine from the server side and the message is deleted from the email server after download. On other hand in case of IMAP ther message in not deleted in the email server and thus can reopened from another location/machine.

In assignment you have to explain the functions performed at each layer and the the protocols envolved on every layer of TCP/IP model when e-mail is sent from source to destination.

SMTP is a connection oriented, text based protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically a Transmission Control Protocol (TCP) connection. An SMTP session consists of commands originated by an SMTP client (the initiating agent, sender, or transmitter) and corresponding responses from the SMTP server (the listening agent, or receiver) so that the session is opened, and session parameters are exchanged. A session may include zero or more SMTP transactions. An SMTP transaction consists of three command/reply sequences (see example below.) They are:

1. MAIL command, to establish the return address, a.k.a. Return-Path, mfrom, or envelope sender. This is the address for bounce messages.
2. RCPT command, to establish a recipient of this message. This command can be issued multiple times, one for each recipient. These addresses are also part of the envelope.
3. DATA to send the message text. This is the content of the message, as opposed to its envelope. It consists of a message header and a message body separated by an empty line. DATA is actually a group of commands, and the server replies twice: once to the DATA command proper, to acknowledge that it is ready to receive the text, and the second time after the end-of-data sequence, to either accept or reject the entire message.

Besides the intermediate reply for DATA, each server's reply can be either positive

(2xx reply codes) or negative. Negative replies can be permanent (5xx codes) or transient (4xx codes). A reject is a permanent failure by an SMTP server; in this case the SMTP client should send a bounce message. Adrop is a positive response followed by message discard rather than delivery.
The initiating host, the SMTP client, can be either an end-user's email client, functionally identified as (MUA), or a relay server's mail transfer agent (MTA), that is an SMTP server acting as an SMTP client, in the relevant session, in order to relay mail. Fully capable SMTP servers maintain queues of messages for retrying message transmissions that resulted in transient failures.
A MUA knows the outgoing mail SMTP server from its configuration. An SMTP server acting as client, i.e. relaying, typically determines which SMTP server to connect to by looking up the MX (Mail eXchange) DNS resource record for each recipient's domain name. Conformant MTAs (not all) fall back to a simple A record in case no MX record can be found. Relaying servers can also be configured to use a smart box.
An SMTP server acting as client initiates a TCP connection to the server on the "well known port" designated for SMTP: port 25. MUAs should use port 587 to connect to an MSA. The main difference between an MTA and an MSA is that SMPTAuthentication is mandatory for the latter only.

Protocol Stacks and Packets

So your computer is connected to the Internet and has a unique address. How does it 'talk' to other computers connected to the Internet? An example should serve here: Let's say your IP address is 1.2.3.4 and you want to send a message to the computer 5.6.7.8. The message you want to send is "Hello computer 5.6.7.8!". Obviously, the message must be transmitted over whatever kind of wire connects your computer to the Internet. Let's say you've dialed into your ISP from home and the message must be transmitted over the phone line. Therefore the message must be translated from alphabetic text into electronic signals, transmitted over the Internet, then translated back into alphabetic text. How is this accomplished? Through the use of a protocol stack. Every computer needs one to communicate on the Internet and it is usually built into the computer's operating system (i.e. Windows, Unix, etc.). The protocol stack used on the Internet is referred to as the TCP/IP protocol stack because of the two major communication protocols used. The TCP/IP stack looks like this:

Protocol Layer	Comments
Application Protocols Layer	Protocols specific to applications such as WWW, e-mail, FTP, etc.
Transmission Control Protocol Layer	TCP directs packets to a specific application on a computer using a port number.
Internet Protocol Layer	IP directs packets to a specific computer using an IP address.
Hardware Layer	Converts binary packet data to network signals and back. (E.g. ethernet network card, modem for phone lines, etc.)

If we were to follow the path that the message "Hello computer 5.6.7.8!" took from our computer to the computer with IP address 5.6.7.8, it would happen something like this:

Diagram 2

1. The message would start at the top of the protocol stack on your computer and work it's way downward.
2. If the message to be sent is long, each stack layer that the message passes through may break the message up into smaller chunks of data. This is because data sent over the Internet (and most computer networks) are sent in manageable chunks. On the Internet, these chunks of data are known as packets.
3. The packets would go through the Application Layer and continue to the TCP layer. Each packet is assigned a port number. Ports will be explained later, but suffice to say that many programs may be using the TCP/IP stack and sending messages. We need to know which program on the destination computer needs to receive the message because it will be listening on a specific port.
4. After going through the TCP layer, the packets proceed to the IP layer. This is where each packet receives it's destination address, 5.6.7.8.
5. Now that our message packets have a port number and an IP address, they are ready to be sent over the Internet. The hardware layer takes care of turning our packets containing the alphabetic text of our message into electronic signals and transmitting them over the phone line.
6. On the other end of the phone line your ISP has a direct connection to the Internet. The ISPsrouter examines the destination address in each packet and determines where to send it. Often, the packet's next stop is another router. More on routers and Internet infrastructure later.
7. Eventually, the packets reach computer 5.6.7.8. Here, the packets start at the bottom of the destination computer's TCP/IP stack and work upwards.
8. As the packets go upwards through the stack, all routing data that the sending computer's stack added (such as IP address and port number) is stripped from the packets.
9. When the data reaches the top of the stack, the packets have been re-assembled into their original form, "Hello computer 5.6.7.8!"