POLYTECH
CLERMONT-FERRAND

Application note

Principle of TCP/IP stack

Explanation of TCP/IP stack used in this application

Author : CHASSAING Jonathan GE5

2013-2014

Summary

Figures:

Introduction:

TCP / IP is a suite of protocols. The acronym TCP / IP stands for "Transmission Control Protocol / Internet Protocol". It comes from the names of the two major protocols of protocol suite, the TCP and IP protocols).

Originally designed for military purpose, the TCP/IP protocol is now the base of the internet. It uses addresses to identify every computer on the network, allowing the safe delivery of data embedded in IP packets. From its military legacy it has retain the following features:

- Splitting messages into packets;
- The use of an address system;
- The routing of network traffic (routing);
- Error checking of data transmission.

History of TCP/IP

The origin of TCP / IP back to the ARPANET. ARPANET is a telecommunication network designed by ARPA (Advanced Research Projects Agency), the research agency of the U.S. Department of Defense. Besides the ability to connect heterogeneous networks, this network had to withstand a possible nuclear war, unlike the telephone network usually used for telecommunications but considered too vulnerable. It was then agreed ARPANET use the technology of packet switching (datagram), a promising emerging technology. It is in this objective and technical choices that TCP and IP protocols were invented in 1974. ARPA then signed several contracts with manufacturers (mainly BBN) and the University of Berkeley Unix developed to enforce this standard, which was done.

How TCP/IP stack works:

TCP / IP is a layered model

In order to implement the TCP / IP model to any machine, regardless of the operating system, the TCP / IP protocols was divided into several modules, each performing a specific task. In addition, these modules perform these tasks one after the other in a specific order, so we stratified system is why we speak of model layers.

The term layer is used to refer to the fact that the data over the network through several protocol layers. The data (information packets) that run on the network are processed sequentially by each layer, just add a piece of information (called a header) and are then transmitted to the next layer.

The TCP / IP model is very close to the OSI model (model with 7 layers), which was developed by the International Organization of Standards (ISO, International Organization for Standardization) in order to standardize communications between computers.

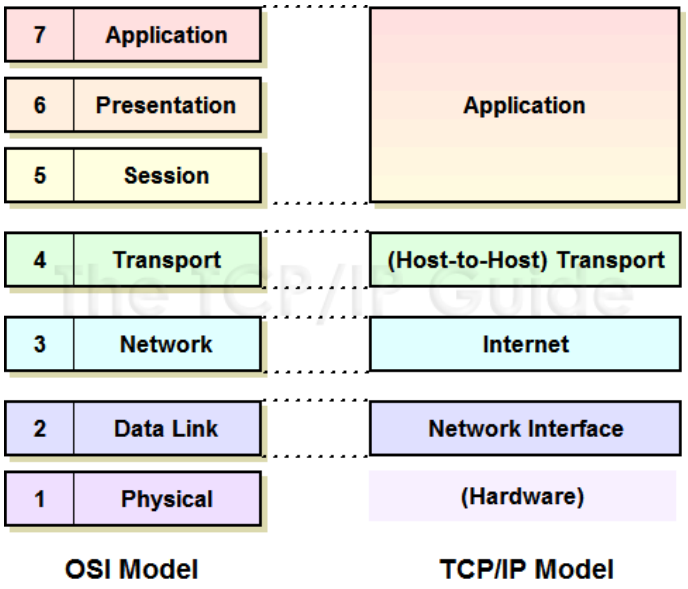

Figure 1 : OSI model Vs TCP/IP model

The host network layer

This layer is quite "strange". Indeed, it seems to "consolidate" the physical and data link layers of the OSI model. In fact, this layer has not really been specified, the only constraint of this layer is to allow a host to send IP packets over the network. The implementation of this layer is left free. In more concrete terms, this implementation is typical of the technology used on the LAN. For example, many local networks use Ethernet, Ethernet is an implementation of the host - network layer.

The internet layer

This layer is the keystone of the architecture. This layer performs the interconnection of networks (heterogeneous) without remote connection. Its role is to allow the injection of packets into any network and routing these packets independently of each other to destination. Since no connection is established in advance, packets may arrive out of order, control of release order is possibly the task of the upper layers.

Due to the imminent role of this layer in the packet routing, the critical point of this layer is routing. It is in this sense that we can afford to compare this layer with the network layer of the OSI model.

The internet layer has an official implementation: IP (Internet Protocol).

Note that the name of the layer ("internet") is written with a lowercase i, for the simple reason that internet word is used here in the broad sense (literally, "network interconnection"), even if the Internet (with a capital I) using this layer.

The transport layer

Its role is the same as the transport layer OSI model: allow peer entities to conduct a conversation.

Officially, this layer has two implementations: TCP (Transmission Control Protocol) and UDP (User Datagram Protocol). TCP is a reliable connection-oriented protocol, which allows error-free routing of packets from a machine internet to another machine on the same internet. Its role is to fragment the message to be transmitted so as to pass it on the internet layer. Conversely, on the destination host, TCP puts in order the fragments transmitted over the internet layer to reconstruct the original message. TCP also handles flow control connection.

UDP, however, is simpler than TCP protocol is unreliable and connectionless. Its use presupposes that you do not need or flow control, or conservation order packet delivery. For example, it is used when the application layer is responsible for the reordering of messages. We remember that in the OSI model layers are in charge of checking the order of message delivery. This is an advantage of the TCP / IP model of the OSI model, but we will return later. Another use of UDP, the transmission of voice. Indeed, reversing two phonemes does not hinder the understanding of the final message. More generally, occurs when the UDP packet delivery time is predominant.

The Application Layer

Unlike the OSI model, the next higher layer to the transport layer, simply because the presentation and session layers appeared unnecessary. It was indeed overview with the use the network software use very rarely these 2 layers, and finally, the OSI model stripped of these two layers is very similar to TCP / IP model.

This layer contains all the high-level protocols, such as Telnet, TFTP (Trivial File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), HTTP (HyperText Transfer Protocol). The important point for this layer is the choice of transport protocol to use. For example, TFTP (mostly used on local area networks) use UDP because it is assumed that the physical connections are sufficiently reliable and sufficiently short transmission time so that there is no reversal of packets to the arrival. This choice makes TFTP faster than FTP protocol that uses TCP. Conversely, SMTP uses TCP as for e-mail delivery, we want all messages reach their entirety and without errors.

Using a TCP/IP stack

Headers

From the basic data to be sent, the TCP / IP stack adds all the layers needed to send this data over the network. These layers can be summarized as headers present upstream of the data, as shown in the following figure.

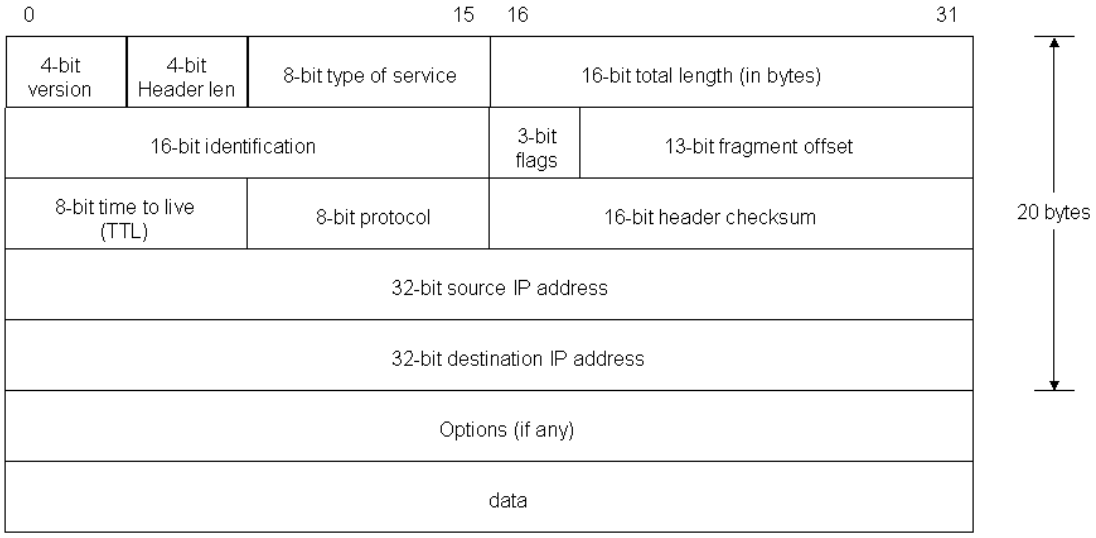

Figure 2 : IP header with data

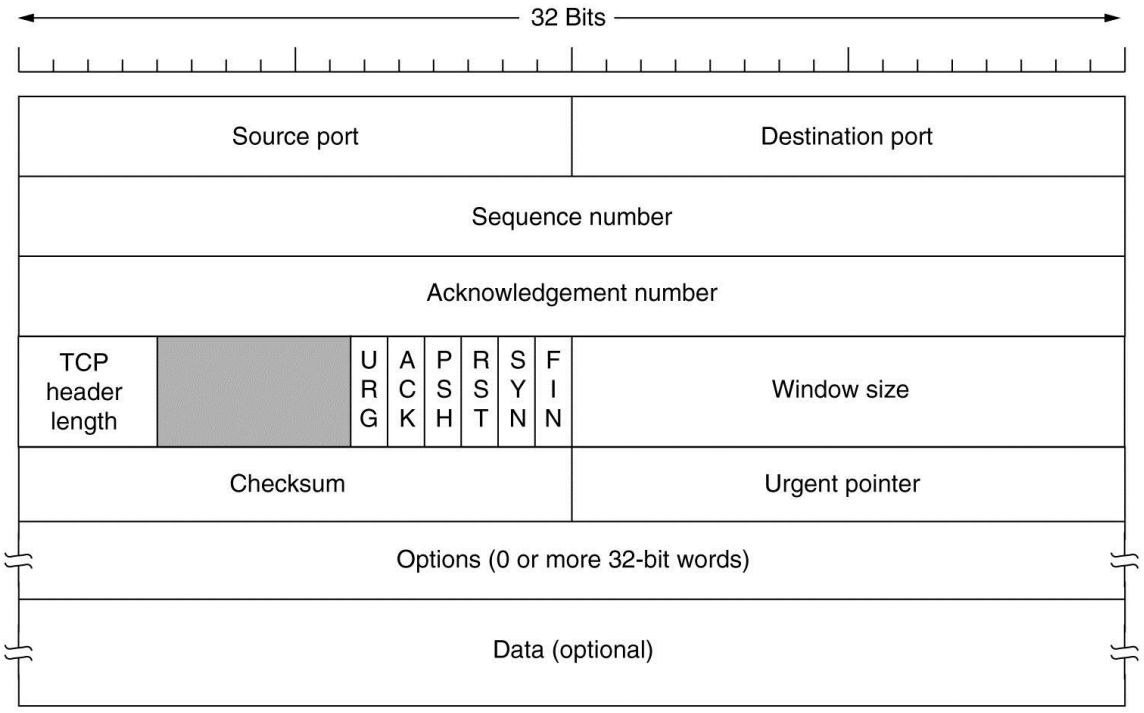

Figure 3 : TCP header (data is IP layer)

These headers are added by the TCP / IP stack to be able to properly orient the data on the network, as the addressee collects all the data, all without errors.

Here the construction of the frame, we will now see how the data travels over the Internet.

Sending a data

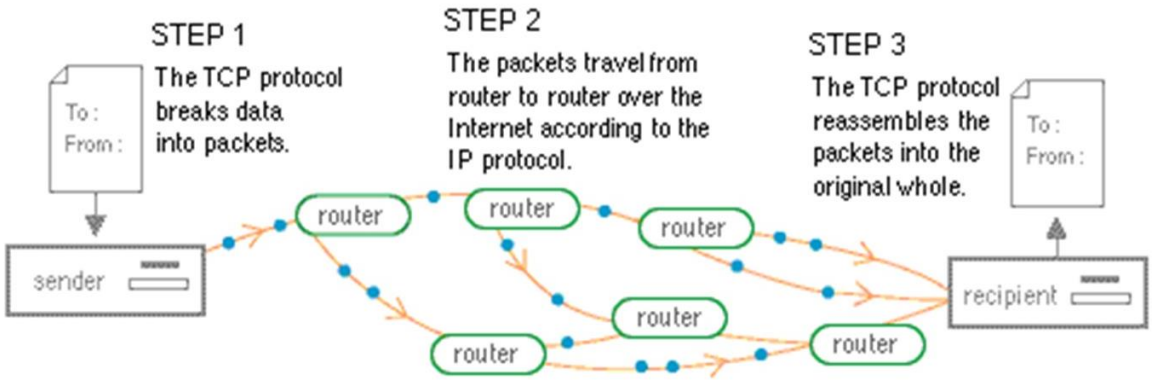

Figure 4 : How data travels over the Net

Conclusion:

TCP / IP stack to format the initial data so that it can be sent over the network correctly. It is a complex process that requires a very good familiarized with its operation to implement.

Fortunately, there are TCP / IP stacks that are proven, widely used by network engineers. Some of these TCP / IP stacks are free.

When developing a network application, so you better focus on the creation of the application upstream, rather than spending time to redo a TCP / IP stack.