[Docs] [txt|pdf] [Errata]

INFORMATIONAL
Errata Exist

Network Working Group
Request for Comments: 1180

T. Socolofsky
C. Kale
Spider Systems Limited

January 1991

A TCP/IP Tutorial

Status of this Memo

This RFC is a tutorial on the TCP/IP protocol suite, focusing particularly on the steps in forwarding an IP datagram from source host to destination host through a router. It does not specify an Internet standard. Distribution of this memo is unlimited.

Table of Contents

1. Introduction

This tutorial contains only one view of the salient points of TCP/IP, and therefore it is the "bare bones" of TCP/IP technology. It omits the history of development and funding, the business case for its use, and its future as compared to ISO OSI. Indeed, a great deal of technical information is also omitted. What remains is a minimum of information that must be understood by the professional working in a TCP/IP environment. These professionals include the systems administrator, the systems programmer, and the network manager.

This tutorial uses examples from the UNIX TCP/IP environment, however the main points apply across all implementations of TCP/IP.

Note that the purpose of this memo is explanation, not definition. If any question arises about the correct specification of a protocol, please refer to the actual standards defining RFC.

The next section is an overview of TCP/IP, followed by detailed descriptions of individual components.

2. TCP/IP Overview

The generic term "TCP/IP" usually means anything and everything related to the specific protocols of TCP and IP. It can include other protocols, applications, and even the network medium. A sample of these protocols are: UDP, ARP, and ICMP. A sample of these applications are: TELNET, FTP, and rcp. A more accurate term is "internet technology". A network that uses internet technology is called an "internet".

2.1 Basic Structure

To understand this technology you must first understand the following

```
                     ----------------------------
                     |    network applications  |
                     |                          |
                     |...  \ | /  ..  \ | /  ...|
                     |     -----      -----     |
                     |     |TCP|      |UDP|     |
                     |     -----      -----     |
                     |         \      /         |
                     |         --------         |
                     |         |  IP  |         |
                     |  -----  -*------         |
                     |  |ARP|   |               |
                     |  -----   |               |
                     |      \   |               |
                     |      ------              |
                     |      |ENET|              |
                     |      ---@--              |
                     ----------|-----------------
                               |
         ----------------------o---------
             Ethernet Cable
```

Figure 1. Basic TCP/IP Network Node

This is the logical structure of the layered protocols inside a computer on an internet. Each computer that can communicate using internet technology has such a logical structure. It is this logical structure that determines the behavior of the computer on the internet. The boxes represent processing of the data as it passes through the computer, and the lines connecting boxes show the path of

data. The horizontal line at the bottom represents the Ethernet cable; the "o" is the transceiver. The "*" is the IP address and the "@" is the Ethernet address. Understanding this logical structure is essential to understanding internet technology; it is referred to throughout this tutorial.

2.2 Terminology

The name of a unit of data that flows through an internet is dependent upon where it exists in the protocol stack. In summary: if it is on an Ethernet it is called an Ethernet frame; if it is between the Ethernet driver and the IP module it is called a IP packet; if it is between the IP module and the UDP module it is called a UDP datagram; if it is between the IP module and the TCP module it is called a TCP segment (more generally, a transport message); and if it is in a network application it is called a application message.

These definitions are imperfect. Actual definitions vary from one publication to the next. More specific definitions can be found in RFC 1122, section 1.3.3.

A driver is software that communicates directly with the network interface hardware. A module is software that communicates with a driver, with network applications, or with another module.

The terms driver, module, Ethernet frame, IP packet, UDP datagram, TCP message, and application message are used where appropriate throughout this tutorial.

2.3 Flow of Data

Let's follow the data as it flows down through the protocol stack shown in Figure 1. For an application that uses TCP (Transmission Control Protocol), data passes between the application and the TCP module. For applications that use UDP (User Datagram Protocol), data passes between the application and the UDP module. FTP (File Transfer Protocol) is a typical application that uses TCP. Its protocol stack in this example is FTP/TCP/IP/ENET. SNMP (Simple Network Management Protocol) is an application that uses UDP. Its protocol stack in this example is SNMP/UDP/IP/ENET.

They are also 1-to-n de-multiplexers. As de-multiplexers they switch one input to many outputs according to the type field in the protocol header.

```
         1   2 3 ...   n                   1   2 3 ...   n
          \  |      /      |                \  | |      /       ^
           \ | |   /       |                 \ | |     /        |
         -------------   flow              ----------------   flow
         |multiplexer|    of               |de-multiplexer|    of
         -------------   data              ----------------   data
              |            |                      |              |
              1            v                      1              |
```

Figure 2. n-to-1 multiplexer and 1-to-n de-multiplexer

If an Ethernet frame comes up into the Ethernet driver off the network, the packet can be passed upwards to either the ARP (Address Resolution Protocol) module or to the IP (Internet Protocol) module. The value of the type field in the Ethernet frame determines whether the Ethernet frame is passed to the ARP or the IP module.

If an IP packet comes up into IP, the unit of data is passed upwards to either TCP or UDP, as determined by the value of the protocol field in the IP header.

If the UDP datagram comes up into UDP, the application message is passed upwards to the network application based on the value of the port field in the UDP header. If the TCP message comes up into TCP, the application message is passed upwards to the network application based on the value of the port field in the TCP header.

The downwards multiplexing is simple to perform because from each starting point there is only the one downward path; each protocol module adds its header information so the packet can be de-multiplexed at the destination computer.

Data passing out from the applications through either TCP or UDP converges on the IP module and is sent downwards through the lower network interface driver.

Although internet technology supports many different network media, Ethernet is used for all examples in this tutorial because it is the most common physical network used under IP. The computer in Figure 1 has a single Ethernet connection. The 6-byte Ethernet address is unique for each interface on an Ethernet and is located at the lower interface of the Ethernet driver.

The computer also has a 4-byte IP address. This address is located at the lower interface to the IP module. The IP address must be unique for an internet.

A running computer always knows its own IP address and Ethernet address.

2.4 Two Network Interfaces

If a computer is connected to 2 separate Ethernets it is as in Figure 3.

```
                  ----------------------------
                  |    network applications  |
                  |     -----      -----     |
                  |     |TCP|      |UDP|     |
                  |     -----      -----     |
                  |         \      /         |
                  |         --------         |
                  |         |  IP  |         |
                  |  -----  -*----*-  -----  |
                  |  |ARP|   |    |   |ARP|  |
                  |  -----   |    |   -----  |
                  |      \   |    |   /      |
                  |      ------  ------      |
                  |      |ENET|  |ENET|      |
                  |      ---@--  ---@--      |
                  ----------|-------|---------
                            |       |
                            |    ---o----------------------------
                            |             Ethernet Cable 2
             ---------------o----------
               Ethernet Cable 1
```

Figure 3. TCP/IP Network Node on 2 Ethernets

Please note that this computer has 2 Ethernet addresses and 2 IP addresses.

It is seen from this structure that for computers with more than one physical network interface, the IP module is both a n-to-m multiplexer and an m-to-n de-multiplexer.

```
         1   2 3 ...   n                   1   2 3 ...   n
          \  | |      /    |                \  | |      /       ^
           \ | |     /     |                 \ | |     /        |
         -------------   flow              ----------------   flow
         |multiplexer|    of               |de-multiplexer|    of
         -------------   data              ----------------   data
           / | |     \     |                 / | |     \        |
          /  | |      \    v                /  | |      \       |
         1   2 3 ...   m                   1   2 3 ...   m
```

Figure 4. n-to-m multiplexer and m-to-n de-multiplexer

It performs this multiplexing in either direction to accommodate incoming and outgoing data. An IP module with more than 1 network interface is more complex than our original example in that it can forward data onto the next network. Data can arrive on any network interface and be sent out on any other.

```
                           TCP      UDP
                             \      /
                              \    /
                          --------------
                          |     IP     |
                          |            |
                          |    ---     |
                          |   /   \    |
                          |  /     v   |
                          --------------
                           /         \
                          /           \
                       data           data
                      comes in         goes out
                     here               here
```

Figure 5. Example of IP Forwarding a IP Packet

called "forwarding" an IP packet. A computer that has been dedicated to the task of forwarding IP packets is called an "IP-router".

As you can see from the figure, the forwarded IP packet never touches the TCP and UDP modules on the IP-router. Some IP-router implementations do not have a TCP or UDP module.

2.5 IP Creates a Single Logical Network

The IP module is central to the success of internet technology. Each module or driver adds its header to the message as the message passes

down through the protocol stack. Each module or driver strips the corresponding header from the message as the message climbs the protocol stack up towards the application. The IP header contains the IP address, which builds a single logical network from multiple physical networks. This interconnection of physical networks is the source of the name: internet. A set of interconnected physical networks that limit the range of an IP packet is called an "internet".

2.6 Physical Network Independence

IP hides the underlying network hardware from the network applications. If you invent a new physical network, you can put it into service by implementing a new driver that connects to the internet underneath IP. Thus, the network applications remain intact and are not vulnerable to changes in hardware technology.

2.7 Interoperability

If two computers on an internet can communicate, they are said to "interoperate"; if an implementation of internet technology is good, it is said to have "interoperability". Users of general-purpose computers benefit from the installation of an internet because of the interoperability in computers on the market. Generally, when you buy a computer, it will interoperate. If the computer does not have interoperability, and interoperability can not be added, it occupies a rare and special niche in the market.

2.8 After the Overview

With the background set, we will answer the following questions:

When sending out an IP packet, how is the destination Ethernet address determined?

How does IP know which of multiple lower network interfaces to use when sending out an IP packet?

How does a client on one computer reach the server on another?

Why do both TCP and UDP exist, instead of just one or the other?

What network applications are available?

These will be explained, in turn, after an Ethernet refresher.

3. Ethernet

This section is a short review of Ethernet technology.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.