

The TCP/IP Guide

A TCP/IP Reference You Can Understand!

NOTE: Using software to mass-download the site **degrades the server and is prohibited.**
If you want to read The TCP/IP Guide offline, [please consider licensing it](#). Thank you.



[The TCP/IP Guide](#)

- 9 [TCP/IP Lower-Layer \(Interface, Internet and Transport\) Protocols \(OSI Layers 2, 3 and 4\)](#)
- 9 [TCP/IP Internet Layer \(OSI Network Layer\) Protocols](#)
- 9 [Internet Protocol \(IP/IPv4, IPng/IPv6\) and IP-Related Protocols \(IP NAT, IPSec, Mobile IP\)](#)
- 9 [Internet Protocol Version 4 \(IP, IPv4\)](#)
- 9 [IP Datagram Encapsulation and Formatting](#)



[IP Datagram Encapsulation and Formatting](#)



Pages

1



[IP Datagram General Format](#)



IP Datagram Encapsulation

In [the chapter describing the OSI Reference Model](#), I looked at several ways that protocols at various layers in a networking protocol stack interact with each other. One of the most important concepts in inter-protocol operation is that of *encapsulation*. Most data originates within the higher layers of the OSI model. The protocols at these layers pass the data down to lower layers for transmission, usually in the form of discrete messages. Upon receipt, each lower-level protocol takes the entire contents of the message received and encapsulates it into its own message format, adding a header and possibly a footer that contain important control information. [Encapsulation is explained in general terms in a separate topic](#).

A good analogy for how encapsulation works is a comparison to sending a letter enclosed in an envelope. You might write a letter and put it in a white envelope with a name and address, but if you gave it to a courier for overnight delivery, they would take that envelope and put it in a larger delivery envelope. ([I actually have written a complete description of this sort of analogy](#), if you are interested.)

Due to the prominence of TCP/IP, the Internet Protocol is one of the most important places where data encapsulation occurs on a modern network. Data is passed to IP typically from one of the two main transport layer protocols: TCP or UDP. This data is already in the form of a TCP or UDP message with TCP or UDP headers. This is then encapsulated into the body of an IP message, usually called an *IP datagram* or *IP packet*. Encapsulation and formatting of an IP datagram is also sometimes called *packaging*—again, the implied comparison to an envelope is obvious. The process is shown in [Figure 85](#).

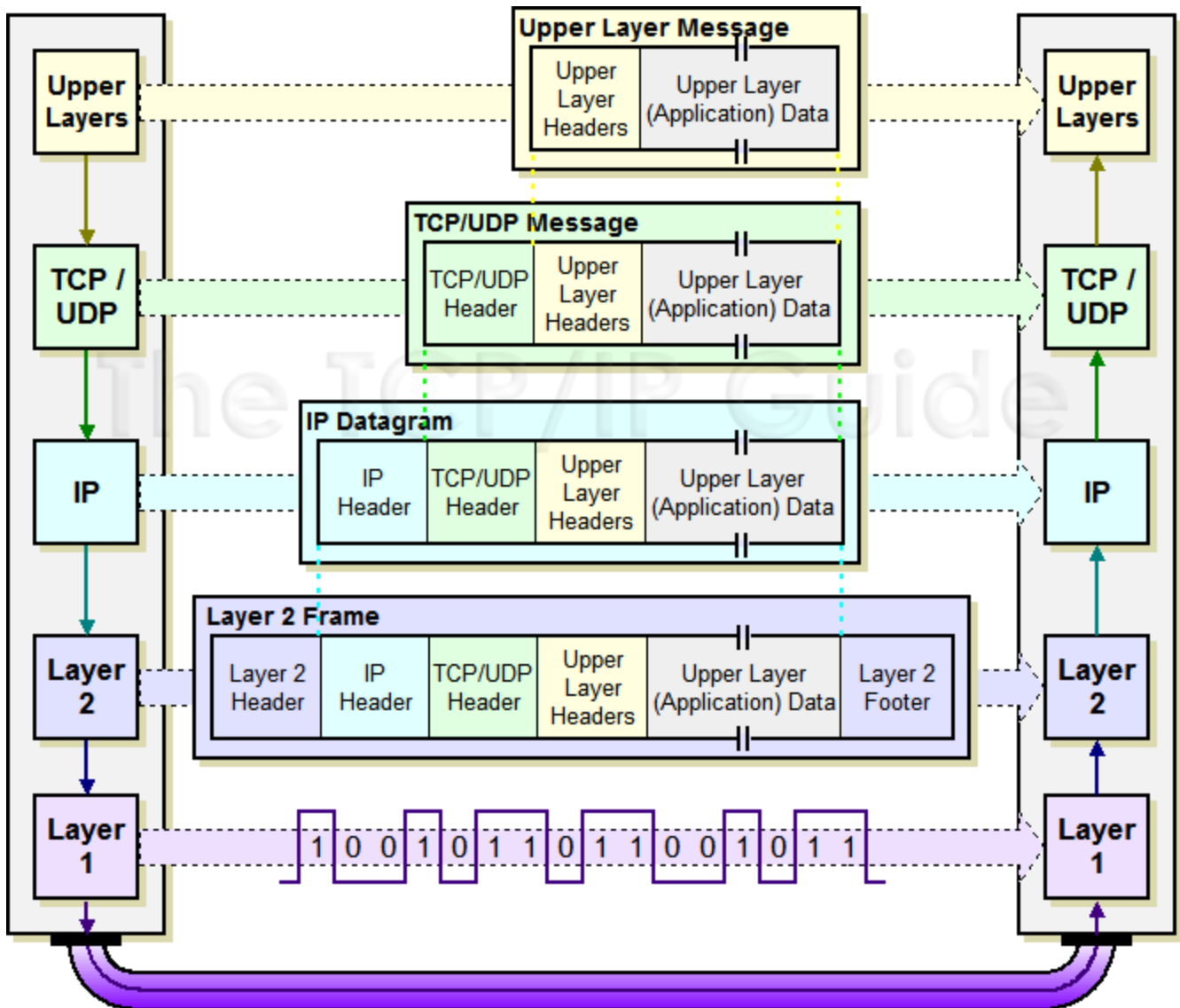


Figure 85: IP Datagram Encapsulation

This is an adaptation of [Figure 15](#), the very similar drawing for the OSI Reference Model as a whole, showing specifically how data encapsulation is accomplished in TCP/IP. As you can see, an upper layer message is packaged into a TCP or UDP message. This then becomes the payload of an IP datagram, which is shown here simply with one header (things can get a bit more complex than this.) The IP datagram is then passed down to layer 2 where it is in turn encapsulated into some sort of LAN, WAN or WLAN frame, then converted to bits and transmitted at the physical layer.

If the message to be transmitted is too large for the size of the underlying network, it may first be [fragmented](#). This is analogous to splitting up a large delivery into multiple smaller envelopes or boxes. In this case, each IP datagram carries only part of the higher-layer message. The receiving device must [reassemble the message](#) from the IP datagrams. So, a datagram doesn't always carry a full higher-layer message; it may hold only part of one.

The IP datagram is somewhat similar in concept to a frame used in Ethernet or another data link layer. The important difference, of course, is that IP datagrams are designed to facilitate transmission across an **internetwork**, while data link layer frames are used only for direct delivery within a physical network. The fields included in the IP header are used to manage internetwork datagram delivery. This

includes key information for delivery such as the address of the destination device, identification of the type of frame, and control bits. [The header follows a specific format described in the following topic.](#)

After data is encapsulated into an IP datagram, it is passed down to the data link layer for transmission across the current “hop” of the internetwork. There, it is of course further encapsulated, IP header and all, into a data link layer frame such as an Ethernet frame. An IP datagram may be encapsulated into many such data link layer frames [as it is routed across the internetwork](#); on each hop the IP datagram is removed from the data link layer frame and then repackaged into a new one for the next hop. The IP datagram, however, is not changed (except for some control fields) until it reaches its final destination.



[IP Datagram Encapsulation and Formatting](#)



Pages

1



[IP Datagram General Format](#)



If you find The TCP/IP Guide useful, please consider making a small Paypal donation to help the site, using one of the buttons below. You can also donate a custom amount using the far right button (not less than \$1 please, or PayPal gets most/all of your money!) In lieu of a larger donation, you may wish to [consider purchasing a download license of The TCP/IP Guide](#). Thanks for your support!

Donate \$2



Donate \$5



Donate \$10



Donate \$20



Donate \$30



Donate: \$



[Home](#) - [Table Of Contents](#) - [Contact Us](#)

[The TCP/IP Guide](#) (<http://www.TCPIPGuide.com>)

[Version 3.0](#) - Version Date: September 20, 2005

© Copyright 2001-2005 Charles M. Kozierok. All Rights Reserved.

[Not responsible for any loss resulting from the use of this site.](#)